

Technology
Arts Sciences
TH Köln

Faculty of Information, Media and Electrical Engineering

Master Computer Science and Systems #Engineering 2024

Module Manual

Version: 1.1.2025-08-29-11-03-34

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

<https://f07-studieninfo.web.th-koeln.de/mhb/current/en/MaTIN2024.html>

1. Program Description

Digitalization affects all areas of life: From toys and smartphones to cars and homes, there is hardware and software. Everyday processes such as shopping or pure communication are digitally permeated. A deep understanding of basic technology in all its breadth and depth, including theoretical models, communication and interaction possibilities as well as hardware and software architectures and intelligent, self-learning systems, is therefore becoming increasingly important.

Orientation of the degree program

The application-oriented Master's degree course in Computer Engineering is designed for a standard period of study of 3 semesters. The course builds consecutively on the Bachelor's degree in Computer Engineering. Computer Engineering. However, it is also suitable as a second course of study to a Bachelor's degree in Computer Science, Electrical Engineering or Media Technology. Students gain in-depth knowledge and understanding of advanced concepts, methods and technologies in computer science and IT-related communication technology. They will advance to the current state of the art. A high proportion of practical work and projects ensures that students can relate what they have learned in the individual subjects to each other and use it to solve challenging practical and theoretical problems. theoretischer Probleme einsetzen können.

Occupational fields

The career prospects and future opportunities for computer engineering graduates are very good. In this respect, students with a good degree lay a promising foundation for their career. Graduates of the course are particularly predestined for planning, development and management tasks. They have access to a wide range of fields of activity both in companies in the information and communication technology companies, in related fields such as the automotive industry and automation technology as well as with service providers such as banks and insurance companies. Graduates with a Master of Science degree can also go on to study for a doctorate or take up a position in the senior civil service of public institutions. Course of study The 3-semester course initially teaches in-depth specialist knowledge in theoretical computer science, computer engineering, mathematics and interdisciplinary skills and soft skills. Specialization is achieved through the individual combination of nine elective modules from various disciplines of Computer Engineering and other computer science and communication technology disciplines. These subjects provide an in-depth knowledge and understanding of the advanced concepts, methods and technologies of the chosen disciplines. In this way, students can shape the subject focus of their studies according to their personal interests. Active participation in a current research project in a current research project at the institute is firmly anchored in the course. Here, students practice the scientifically sound analysis and solution of novel problems. The degree program concludes with the completion of a Master's thesis: methods and techniques of scientific work are applied independently to a challenging task.

Alongside the course, professional further training is offered to become a Cisco Certified Network Associate Security (CCNA Security).

2. Graduate Profile

Graduates of the M. Sc. Computer Science and Systems Engineering degree program have in-depth scientific and methodological skills in the design, analysis and development of complex IT systems. In contrast to the Bachelor's degree program, the focus is on research, innovation, leadership and management skills as well as the cross-system application of IT technologies. You will be able to independently take responsibility for challenging projects in development, research or management and actively shape technological transformation processes.

The Master's degree program in Computer Science and Systems Engineering is aimed at graduates with a sound first degree in computer science and deepens their knowledge in the field of software, hardware, distributed systems, AI, signal processing and communication technology.

Compared to the Bachelor's degree course, which focuses on the broad teaching of technical fundamentals and applications, the Master's degree focuses on:

- Scientific work and research-related project development
- Individual specialization through elective modules
- Interdisciplinary systems thinking
- Leadership and innovation skills

Graduates of the Master's degree program develop an individual profile in the following areas:

- You are proficient in the development, analysis and evaluation of complex IT systems, taking into account technical, social, legal and ecological framework conditions.
- You will be qualified for management tasks in research and development projects, including the management of interdisciplinary and international teams.
- Through intensive project work and the research project, you will acquire the ability to work independently on challenging issues with scientific depth.
- They can independently develop technical innovations in areas such as distributed systems, AI, embedded systems, network technologies and multimedia communication and transfer them into industrial or scientific applications.
- You will be able to contribute to the advancement of knowledge in computer science through scientific work and qualify for a doctorate.
- In addition to technical skills, the course also promotes self-organization, ethical judgement and intercultural communication skills - key requirements in modern technology professions.
- There are a wide range of professional fields: from technology development in industry, IT strategy and consulting to research at universities, institutes and public institutions.

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.

Development and Design

Interdisciplinary development and testing of algorithms, circuits, software, devices, communication and media technology systems as well as complex computer, communication and embedded systems.

Research and innovation

Perform scientific research work and apply and expand scientific knowledge, from basic research to industrial research, with the qualification for doctoral studies.

Leadership and management

Assume technical leadership and project responsibility, including the coordination and management of working groups and internationally distributed teams, as well as the management of planning and manufacturing processes, project controlling and product management.

Quality assurance and testing

Carrying out quality controls and tests for products and processes, using measurement and testing technologies and coordinating certification processes.

4. Competencies

The modules of the degree program train students in different competencies, which are described below. The [Profile Module Matrix] (#Module Matrix) shows which competences are addressed by which modules.

Development and design of complex systems

Ability to design and implement large systems, taking into account electrical, software, mechanical and optical aspects, based on a thorough requirements analysis from a technical, economic, ecological and social point of view.

Testing and evaluation of complex systems

Plan, perform and analyze tests to verify and validate these systems, including consideration of user perspectives and technical-economic aspects.

Scientific work and research

Mastery and application of scientific methods, including the ability to research, evaluate and cite relevant literature, and to formulate and present results.

Project management and teamwork

Skills in organizing, managing and supervising projects and teams, even under uncertain conditions, and in making professional and organizational decisions.

Self-organization and self-taught skills

Identification of personal skills, efficient time management and the ability for self-directed learning.

Communication and intercultural competence

Ability to present and defend scientific and technical results convincingly in both German and English, including international and interdisciplinary contexts.

Technical and scientific fundamentals

Comprehensive and in-depth STEM subject knowledge and its application to real-world and theoretical problems.

Sustainability and social responsibility

Evaluate and develop sustainable and socially responsible technologies, including consideration of ethical values.

Analysis, simulation and abstraction

Ability to analyze complex systems, abstract key features and solve problems based on models.

Leadership and decision-making responsibility

Assume responsibility in technical management tasks, develop solution strategies for complex tasks.

Applying ethical values and principles in practice

Incorporating social and ethical considerations into technical decisions and design processes.

Integrative thinking and acting in interdisciplinary teams

Coordinating and integrating contributions from different disciplines to solve complex tasks.

Innovation and creativity

Developing new solutions and concepts to overcome technical challenges.

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	Prüfungslast	Examination Types with Weights
1	THI	Theoretische Informatik	PF	5	1	▪ final: oral examination [100%]
	VMT	Vertiefung Mathematik	WB	5	≤ 2	▪ selection dependend
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ selection dependend
	W	Allgemeiner Wahlbereich	WB	10	≤ 4	▪ selection dependend
	X	Fachübergreifende Kompetenzen und Soft Skills	WB	5	≤ 2	▪ selection dependend
2	FP	Forschungsprojekt	PF	10	1	▪ accompanying: term paper [100%]
	VTI	Vertiefung Technische Informatik	WB	10	≤ 4	▪ selection dependend
	W	Allgemeiner Wahlbereich	WB	10	≤ 4	▪ selection dependend
3	MAA	Masterarbeit	PF	27	1	▪ final: Thesis [100%]
	KOLL	Kolloquium zur Masterarbeit	PF	3	1	▪ final: Colloquium [100%]

5.2 Alternativer Studienverlaufsplan

Sem.	Abbr.	Module Name	Mandatory (PF) Elective Catalog (WB)	ECTS	Prüfungslast	Examination Types with Weights
1	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ selection dependend
	VMT	Vertiefung Mathematik	WB	5	≤ 2	▪ selection dependend
	THI	Theoretische Informatik	PF	5	1	▪ final: oral examination [100%]
2	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ selection dependend
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ selection dependend
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ selection dependend
3	X	Fachübergreifende Kompetenzen und Soft Skills	WB	5	≤ 2	▪ selection dependend
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ selection dependend
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ selection dependend
4	FP	Forschungsprojekt	PF	10	1	▪ accompanying: term paper [100%]
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ selection dependend
5	MAA	Masterarbeit	PF	27	1	▪ final: Thesis [100%]
	KOLL	Kolloquium zur Masterarbeit	PF	3	1	▪ final: Colloquium [100%]

6. Modules

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

6.1 ACC - Advanced Channel Coding

Module ID	ACC_MaTIN2024
Module Name	Advanced Channel Coding
Type of Module	Elective Modules
Recognized Course	ACC - Advanced Channel Coding
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

What? Designing and rating of systems for the reliable transmission of data over distorted channels and storage of data for data at rest and data in motion

How? By applying results from information theory and applying methods and algorithms for error correcting codes using existing simulations tools, self written programmes, and studying existing systems.

What for? To be able to design, select, use and apply actual and future digital communication systems for reliable data transmission, and to rate their performance.

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:

- Introduction
- Basic terms and definitions
- short history of channel coding
- System and channel models
- Review of binary error correcting block and convolutional Codes
- Generator and Parity check matrices,
- decoding principles, Trellis and Viterbi Algorithm
- Some principles on Information Theory
- Channel coding theorem
- Channel capacity and example calculations
- Cyclic Codes, Reed Solomon Codes
- Encoding and Decoding, Euclidean and Berlekamp-Massey -Algorithm for Decoding
- Basics on LDPC, Polar, and TURBO Codes
- iterative decoding, Sum Product Algorithm
- Recursive Convolutional Codes
- Performance comparison
- Basics on Space Time Coding
- Channel Model, Capacity improvement, Alamouti Scheme, STBC and STTC and their decoding

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

Existing simulation Tools like, e.g., the Matlab Communications Toolbox or AFF3CT (aff3ct.github.io) are used to:

- test theoretical results from lecture and tutorial
- implement algorithms for error control coding
- simulate BER and rate the performance, compare schemes
- adapt programs to solve equivalent problems
- become familiar with standard simulations tools
- train cooperation in small teams

Students learn to generate, check, present, and discuss performance results for FEC codes. They need to search for and to study scientific literature as background sources for their simulations. Teams of students get different code families to study. Results are presented to the whole group.

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Lab

Examination Types with Weights	▪ accompanying: oral contribution or exercise lab under exam conditions [ungraded] and ▪ final: (digital) written exam or oral examination [100%]
---------------------------------------	--

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

Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul HIM: Grundkenntnisse zur linearen Algebra, der Algebra in endlichen Zahlkörpern, der Stochastik und der digitalen Kommunikationstechnik aus den vorangegangenen Bachelorstudiengängen. Da das Fach im ersten Fachsemester des Masters gewählt werden kann, können keine belastbaren Kenntnisse aus dem Fach HIM verpflichtend vorausgesetzt werden, auch wenn sie hilfreich wären. ▪ basics in linear algebra ▪ basics in stochastics ▪ good programming skills
Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 2 Praktikumstermine und 1 Präsentation ▪ Participation in final examination only after successful participation in Lab
Recommended Literature	▪ R. E. Blahut. Algebraic Codes for Data Transmission. Cambridge University Press, Cambridge, 2003. ▪ S. Lin and D. J. Costello. Error Control Coding. ISBN 0-13-042672-5. Prentice-Hall, 2004 ▪ T. M. Cover and J. A. Thomas. Elements of Information Theory. Wiley, New Jersey, 2006 ▪ A. Neubauer. Kanalcodierung. Schlembach, Wilburgstetten, 2006. ▪ R. Roth. Introduction to Coding Theory. Cambridge, second edition, 2006 ▪ B. Sklar. Digital Communications. Prentice Hall PTR, Upper Saddle River, New Jersey, 2001
Use of the Module in Other Study Programs	▪ ACC in Master Communication Systems and Networks 2020 ▪ ACC in Master Communication Systems and Networks 2024 ▪ ACC in Master Technische Informatik 2020
Permanent Links to Organization	ILU course page
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.2 AMC - Advanced Multimedia Communications

Module ID	AMC_MaTIN2024
Module Name	Advanced Multimedia Communications
Type of Module	Elective Modules
Recognized Course	AMC - Advanced Multimedia Communications
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

What?

Understanding service requirements, driven by heterogeneous services, in All-IP networks, and how to design, implement and evaluate quality-of-service (QoS) and quality-of-experience (QoE) mechanisms. Competences to evaluate, analyze, design, implement and test multiservice IP networks with heterogeneous service requirements.

How?

Based on Bachelor-level competences on IP networking and services, students learn different application (service) requirements from filetransfer to streaming and how to separate and fulfill these requirements in IP networks. In a small team and organized as semester project, students develop their own multiservice networks, optionally based on existing systems, and learn how to design, implement and analyze their own multiservice network solution.

What for?

To be able to design, analyze, select, use and apply actual and future network technologies, based on All-IP networks concepts for enterprise networks, telecommunication networks and mobile networks.

Module Contents

Lecture / Exercises

Content for multimedia applications, encoding of multimedia data, integration of data, audio and video, multimedia traffic requirements, multimedia transport protocols, RTP and MPEG-TS, traffic modeling burst silence model, quality of service (QoS), multiservice networks, IntServ, RSVP, DiffServ, ToS and DSCP, Traffic Classification, Traffic Measurement, Traffic Shaping, Network Scheduling, Queueing (FIFO, RR, WRR, WFQ, CB-WFQ, PQ, LLQ), Congestion Avoidance (RED, WRED, CB-WRED), Quality-of-Experience (QoE), MOS Scale, Error Detection, Error Correction, FEC, Interleaving, Jitter Buffer.

Students evaluate technologies and network architectures of multiservice networks; they analyze requirements of Multimedia services and systems, design architectures for multiservice networks, implement multiservice networks, and analyze Multimedia communication protocols and their performance metrics.

Lab

Fundamental knowledge of multiservice networks or multimedia applications in All-IP networks including planning, implementation and evaluation of services. Protocol analysis for functional analysis, performance analysis and troubleshooting.

Students evaluate requirements of Multimedia services, and necessary methods for QoS in multiservice networks. They plan and implement IP Multimedia environments as team project, and test QoS performance measures. They are competent in functional analysis and troubleshooting by deep packet inspection (DPI) protocol analysis. They evaluate the performance of the Multimedia network or services in terms of timing, throughput, latency and delays, jitter, robustness in case of packet errors, and security aspects. Individual project proposals by students are welcome.

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

Examination Types with Weights	■ accompanying: exercise lab [ungraded] and ■ final: (digital) written exam or oral examination [100%]
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	■ Modul NGN: Fundamentals of Networks and Protocols (typically Bachelor Level, like prerequisites in NGN) Layered Communications and Protocol Stacks (ISO/OSI, IETF TCP/IP, IEEE), LAN, MAN, WAN, Fixed Line and Mobile Network Fundamentals, Data Link-Technologies (Ethernet, WiFi), IP-Networking (IPv4, IPv6), IP Routing Protocols (static Routes, RIP, OSPF, BGP), Transport Protocols (TCP (incl. Flow Control / Congestion Control), UDP) and Port Numbers, Application Protocols (HTTP, Request-Response Pattern, Publish-Subscribe Pattern). ■ Bachelor-level knowledge of protocols and layer models, Internet protocols (UDP, TCP, IP, HTTP, FTP), IP addressing (IPv4, IPv6), routing techniques (IP routing, functionality of a router, routing protocols, RIP, OSPF), transmission systems and layer 2 protocols, Ethernet. Understanding distributed systems and applications, sockets and client/server programming, request-response patterns, publishg-subscribe patterns.
Mandatory Prerequisites	■ Lab requires attendance in the amount of: 6 Meilensteintermine und Projektvorstellungen ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ J. Kurose, K. Ross: Computer Networking: A Top-Down Approach, Global Edition, Prentice Hall, 7th ed., 2016 ■ A. S. Tanenbaum, D. J. Wetherall: Computer Networks, Pearson , 5th ed., 2013 ■ W. Stallings: Foundations of Modern Networking, Pearson Education, 2016 ■ H. W. Barz, G. A. Bassett: Multimedia Networks, John Wiley & Sons, 2016 ■ T. Szigeti, C. Hattingh, R. Barton, B. Kenneth: End-to-End QoS Network Design: Quality of Service for Rich-Media & Cloud Networks (2nd Edition) End-to-End QoS Network Design: Quality of Service for Rich-Media & Cloud Networks, Cisco Press, 2nd Ed. 2013 ■ R. Steinmetz, K. Nahrstedt: „Multimedia Systems“, Springer 2004 ■ R. Steinmetz, „Multimedia-Technologie“, Springer 2000
Use of the Module in Other Study Programs	■ AMC in Master Communication Systems and Networks 2020 ■ AMC in Master Communication Systems and Networks 2024 ■ AMC in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.3 AMS - Special Aspects of Mobile Autonomous Systems

Module ID	AMS_MaTIN2024
Module Name	Special Aspects of Mobile Autonomous Systems
Type of Module	Elective Modules
Recognized Course	AMS - Special Aspects of Mobile Autonomous Systems
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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: Das Modul vermittelt Kompetenzen zur Entwicklung von mobilen autonomen Systemen, insbesondere im Themenbereich der räumlichen Interpretation und Kognition für die sichere Navigation von unbemannten Roboter- und Fahrzeugsystemen sowie intelligente Interaktion und Kollaboration unter Menschen und Robotern. Womit: Die Dozentin vermittelt Wissen und Basisfertigkeiten in einem Vorlesungsteil und betreut parallel dazu praktische Projekte, wobei die Studierenden mittels forschenden Lernens technische Ansätze studieren und erproben, Prototypen aufbauen und testen, Ergebnisse präsentieren, sowohl technische als auch ethische und soziale Aspekte diskutieren, und das Ganze schriftlich dokumentieren. Wozu: Kompetenzen in der Entwicklung von mobilen autonomen Systemen sind essentiell für technische Informatiker*innen und Nachwuchs in verwandten Ingenieurberufen. Derartige Kompetenzen sind unentbehrlich für die Forschung, Entwicklung sowie technische Innovation. Das projektbasierte und forschende Lernen im Team hilft den Studierenden außerdem, sich mit relevanten ethischen und sozialen Aspekten zu beschäftigen, welche im Zusammenhang mit autonomen Systemen stehen.	
Module Contents	
Lecture	
Mobile autonomous systems Cognitive and behaviour-based robotics Environmental modelling and spatial cognition Interaction and navigation	
Project	
Teamwork: Development of autonomous systems with cognitive capabilities and intelligent behaviours. Such cognitive capabilities include e.g.: Recognize objects with sensors, estimate their locations or movements, make 3D representations and interpretations or build a map of the environment etc. Intelligent behaviours can be demonstrated among others by such actions: Move and navigate autonomously without collision in unknown environments, fetch or transport objects for a special application, nature interactions and collaborations among human and robots.	
Teaching and Learning Methods	▪ Lecture ▪ Project
Examination Types with Weights	▪ accompanying: project work [ungraded] and ▪ final: oral examination [100%]
Workload	150 Hours
Contact Hours	34 Hours $\pm$ 3 SWS
Self-Study	116 Hours

Recommended Prerequisites	Capability of software and project development Knowledge of signal processing and mathematics
Mandatory Prerequisites	Project requires attendance in the amount of: 1 Präsentation
Recommended Literature	▪ Siegwart et.al.: Introduction to autonomous mobile robots, MIT Press, 2010
Use of the Module in Other Study Programs	AMS in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.4 ARP - Alternative Rechnerarchitekturen und Programmiersprachen

Module ID	ARP_MaTIN2024
Module Name	Alternative Rechnerarchitekturen und Programmiersprachen
Type of Module	Elective Modules
Recognized Course	ARP - Alternative Computer Architectures and Programming Languages
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. René Wörzberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Georg Hartung/Professor Fakultät IME im Ruhestand
Learning Outcome(s)	
Die Studierenden lernen kennen, wenden an und analysieren verschiedene wichtige Konzepte von Rechnerarchitekturen und Programmiersprachen. Dazu wenden sie für jedes ausgewählte Konzept nach einer kurzen Vorstellung es auf ein selbstgewähltes Beispiel an, wozu sie sich weiteres Wissen über das Konzept erwerben müssen, und analysieren die Vor- und Nachteile des Konzepts in einem Bericht. Damit erlangen sie einen größeren Überblick über verfügbare Architekturen und Programmiersprachen für ihre spätere Tätigkeit als IT-Spezialist, Manager oder in der Forschung.	
Module Contents	
Lecture / Exercises	
Knowledge of the respective modeling method, programming procedure or architecture and its programming ("Topics"); practice of initial Topic skills in exercises.	
Project	
Application of the topic to a self-selected task, analysis of the features of the topic on a concrete example, synthesis with own experiences, teamwork (processing in a small group)	
Teaching and Learning Methods	▪ Lecture / Exercises ▪ Project
Examination Types with Weights	▪ accompanying: project work [50%] and ▪ final: oral examination or (digital) written exam [50%]
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	- Experience in imperative programming languages, esp. C - Basic knowledge and experience in operating systems, esp. Linux - Basic knowledge and experience in software engineering - Basic knowledge of computer design and operation, including operation of important digital components - Basic knowledge of formal languages and automata theory
Mandatory Prerequisites	Project requires attendance in the amount of: 4 Termine

Recommended Literature

- Jensen, K., Kristensen, L.M.: Coloured Petri Nets
- Nilsson, U.; Maluszynski, J.: Logic, Programming and Prolog
- T. Eiter, G. Ianni, T. Krennwallner: 'Answer Set Programming: A Primer' in: Reasoning WEB Semantic Technologies for Information Systems
- Steve Klabnik and Carol Nichols: The Rust Programming Language
- William Gropp et al.: Using Advanced MPI / Modern Features of the Message Passing Interface, MIT Press
- Gerassimos Barlas Multicore and GPU Programming - An Integrated Approach Morgan Kaufmann Publ., Inc.

Use of the Module in Other Study Programs

ARP in Master Technische Informatik 2020

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.5 AVT - Audio- und Videotechnologien

Module ID	AVT_MaTIN2024
Module Name	Audio- und Videotechnologien
Type of Module	Elective Modules
Recognized Course	AVT - Audio and Video Technologies
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr.-Ing. Klaus Ruelberg/Professor Fakultät IME
Lecturer(s)	Prof. Dr.-Ing. Klaus Ruelberg/Professor Fakultät IME

Learning Outcome(s)

Was:

Audio- und Videotechnologien kommen in vielfältiger Weise in der Medienindustrie zum Einsatz. Die Mediendistributionskette, die im Rahmen der LV als exemplarische Anwendung herangezogen und analysiert wird, umfasst verschiedene Technologien wie Datenkompression, Audio- und Videosignalverarbeitung, Fehlerschutzmechanismen, digitale Modulationsverfahren.

Womit:

Studierende durchdringen eigenständig ausgewählte Themengebiete der Audio- und Videotechnologien, bereiten diese auf und halten einen Fachvortrag.

In einem in die LV integrierter Übungsblock entwickeln die Studierende eigenständig algorithmische Lösungskonzepte und setzen diese programmtechnisch um.

Wozu:

Die Studierenden können aktuelle Verfahren zur Audio- und Videocodierung entwickeln und in Hard- und Software implementieren. Sie können Mediendistributionsketten planen, beurteilen und umsetzen sowie fachliche Führungs- und Projektverantwortung übernehmen.

Module Contents

Lecture / Exercises

Audio and video source coding

Channel models, channel coding (FEC & digital modulation principles)

Broadcast transmission systems (DVB - Digital Video Broadcasting)

implement audio and video coding methods in hard and software

develop algorithms and methods for audio and videocoding

develop and implement digital broadcasting systems

Exercises / Lab

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

Examination Types with Weights	■ accompanying: oral contribution [ungraded] and ■ final: oral examination or (digital) written exam [100%]
---------------------------------------	--

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

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

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

Recommended Prerequisites	none
Mandatory Prerequisites	Exercises / Lab requires attendance in the amount of: 1 Termin
Recommended Literature	■ Proakis, J. Salehi, M. (2007) Digital Communications. McGraw-Hill. ISBN 978-0072957167 ■ Reimers, U. (2001) Digital Video Broadcasting. Springer Verlag. ISBN 978-3-662-04562-6
Use of the Module in Other Study Programs	■ AVT in Master Medientechnologie 2020 ■ AVT in Master Medientechnologie 2024 ■ AVT in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.6 CI - Computational Intelligence

Module ID	CI_MaTIN2024
Module Name	Computational Intelligence
Type of Module	Elective Modules
Recognized Course	CI - Computational Intelligence
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	Prof. Dr. Rainer Bartz/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden erarbeiten sich grundlegende Kenntnisse zur Theorie und Anwendung von Methoden der Computational Intelligence.

Die Studierenden kennen die gängigen Typen von Optimierungsaufgaben und können konkrete Aufgaben einordnen.

Die Studierenden kennen das Prinzip des Simplex-Algorithmus und können eine Problemstellung in die für ihn geeignete Standardform überführen und eine Lösung erarbeiten. Sie können lineare Probleme mit einem Simplex-Algorithmus lösen.

Die Studierenden können neuronale Netze einordnen und ihre Anwendbarkeit auf Problemstellungen bewerten. Sie können Lernverfahren klassifizieren und ihre Arbeitsweise beschreiben. Sie können nichtlineare Probleme der Modellbildung und Klassifizierung mit einem neuronalen Netz lösen.

Sie kennen die Methodik der Fuzzy Logik und können eine Problemstellung darauf abbilden und das resultierende Systemverhalten begründen. Sie können unscharf definierte Aufgaben mit Hilfe von Fuzzy Logik lösen.

Die Studierenden kennen die Arbeitsweise evolutionärer Algorithmen und können ihre Varianten einordnen. Sie können reale Problemstellungen in geeignete Repräsentationen umsetzen. Sie können Selektionsverfahren bewerten und geeignete Selektionsalgorithmen entwerfen. Sie können schwierige Probleme mit Heuristiken der evolutionären Algorithmen lösen.

Die Studierenden können mit üblichen Werkzeugen der Computational Intelligence umgehen.

Die Studierenden können Aufgaben in einem kleinen Team lösen.

Die Studierenden können Systemparameter variieren, Messreihen durchführen und Ergebnisse darstellen, bewerten und diskutieren. Sie können das Verhalten eines Systems bewerten und durch geeignete Modifikationen verbessern.

Die Studierenden können internationale wissenschaftliche Literatur analysieren, einordnen, in ihren Kontext stellen und präsentieren.

Module Contents**Lecture / Exercises**

Optimization strategies

- classification of problems
- gradient algorithms
- simplex algorithm
- multiobjective optimization and Pareto approach

Artificial neural networks

- artificial neurons
- neural network structures
- training algorithms

Fuzzy logic

- fuzzification
- inference
- defuzzification

Evolutionary algorithms

- genome representations
- selection mechanisms
- recombination operators
- mutation operators

The students acquire fundamental knowledge on theory and applications of computational intelligence

The students know about typical classes of optimization tasks and how to map a specific problem to those classes

They know the simplex algorithm and can transform problems into the standard form to find the solutions

The students can classify artificial neural networks and determine their applicability for specific tasks

They can vary the parameters of neural networks and rate their impact on the results

They can classify training algorithms and understand the backpropagation algorithm

They know about the fuzzy logic approach, can apply it to specific problems and justify the resulting system behavior

The students know how evolutionary algorithms work and can distinguish the variants

They can transform a problem specification into a representation appropriate for an evolutionary algorithm

They can rate selection strategies and define suitable algorithms

The students can solve linear problems with the use of the simplex algorithm

They can apply artificial neural networks to solve problems of modeling and classification

They can define fuzzy logic systems to solve imprecise and vague tasks

They can solve difficult problems heuristically using evolutionary algorithms

Lab

Application of artificial neural networks to a classification task

Variation and multiobjective optimization of neural network parameters

Fuzzy-based closed loop control of a system with two inputs

The students are familiar with tools supporting computational intelligence

The students can vary system parameters, perform test series, and evaluate, present and discuss the results

The students are able to understand, present, analyze and discuss scientific publications

The students are able to solve problems in small teams

They can tackle optimization tasks in a structured and systematic way

They can rate the behavior of a system with regard to objectives and study and improve the behavior through parameter variations

They are able to cope with international scientific publications, understanding, presenting and discussing them in their context

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
Examination Types with Weights	▪ accompanying: exercise lab and ▪ final: (digital) written exam or oral examination or project work or lab report
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	vector functions, gradient
Mandatory Prerequisites	Lab requires attendance in the amount of: 2 Termine
Recommended Literature	▪ Domschke W., Drexl A.: Einführung in Operations Research; Springer ▪ Zell, A.: Simulation Neuronaler Netze; Oldenbourg ▪ Nauck, D. et al.: Neuronale Netze und Fuzzy-Systeme; Vieweg ▪ Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing; Springer ▪ Gerdes, I. et al.: Evolutionäre Algorithmen; Vieweg ▪ Grosse et al.: Taschenbuch der praktischen Regelungstechnik, Fachbuchverlag Leipzig
Use of the Module in Other Study Programs	CI in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.7 DLO - Deep Learning und Objekterkennung

Module ID	DLO_MaTIN2024
Module Name	Deep Learning und Objekterkennung
Type of Module	Elective Modules
Recognized Course	DLO - image processing master
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
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)	
Die Teilnehmer*innen können selbständig entscheiden, in welchen Situationen sich der Einsatz von Verfahren aus dem Bereich Deep Learning anbietet. Sie können eine entsprechende Lösung entwerfen, iterativ verbessern und praktisch umsetzen. Mögliche Probleme auf dem Weg dahin (z.B. beim Erstellen eines Datensatzes oder beim Training) können sie qualifiziert analysieren und passende Ideen zur Bewältigung entwickeln. Da sie einen guten Überblick über die langjährigen Entwicklungen in Forschung und Technik haben, können sie qualifiziert auf aktuelle Herausforderungen und offene Fragen im Zusammenhang mit Deep Learning schauen. Die Studierenden werden so in die Lage versetzt, sich sowohl im weiteren Studienverlauf als auch im Berufsleben kompetent mit Ansätzen zu beschäftigen, die auf Deep Learning beruhen.	
Module Contents	
Lecture	
Deep learning algorithms and their application for object recognition in images.	
Lab	
- training of a neural network - evaluation of the performance of a neural network	
Teaching and Learning Methods	- Lecture - Lab
Examination Types with Weights	- accompanying: exercise lab [ungraded] and - final: (digital) written exam or oral examination or single/multiple choice [100%]
Workload	150 Hours
Contact Hours	34 Hours $\triangleq$ 3 SWS
Self-Study	116 Hours
Recommended Prerequisites	The students should have some basic knowledge about image processing and pattern recognition
Mandatory Prerequisites	Lab requires attendance in the amount of: 4 Termine
Recommended Literature	I. Goodfellow, Y. Bengio und A. Courville. Deep Learning. MIT Press, 2016 C. C. Aggarwal. Neural Networks and Deep Learning: A Textbook. Springer, 2018 C. Bishop und H. Bishop. Deep Learning: Foundations and Concepts. Springer, 2024 D. V. Godoy. Deep Learning with PyTorch Step-by-Step: A Beginner's Guide. Fundamentals. 2022 D. V. Godoy. Deep Learning with PyTorch Step-by-Step: A Beginner's Guide. Computer Vision. 2022

Use of the Module in Other Study Programs	▪ DLO in Master Elektrotechnik 2020 ▪ DLO in Master Elektrotechnik und Informationstechnik 2024 ▪ DLO in Master Medientechnologie 2020 ▪ DLO in Master Medientechnologie 2024 ▪ DLO in Master Technische Informatik 2020
--	--

Specifics and Notes

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

6.8 DMC - Digital Motion Control

Module ID	DMC_MaTIN2024
Module Name	Digital Motion Control
Type of Module	Elective Modules
Recognized Course	DMC - Digital Motion Control
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Jens Onno Krah/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Jens Onno Krah/Professor Fakultät IME
Learning Outcome(s)	
- Servomotoren kennenlernen und betreiben - Servoumrichter kennenlernen und verwenden - Digitale Regelalgorithmen nutzen - Prozessidentifikation und Parameterestimation - Auslegung von Antriebssystemen	
Module Contents	
Lecture / Exercises	
- Structure of servo motors - Structure of servo inverters - Digital control algorithms - Process identification - Design of drive systems	
Lab	
- Direct Digital Control - Quasi-continuous control - Predictor / Observer - Parameterization of a control system - Evaluation of Bode diagrams - Demonstrate action competence - Commissioning of a servocontroller - Minimization of following errors	
Teaching and Learning Methods	- Lecture / Exercises - Lab
Examination Types with Weights	- accompanying: lab report [ungraded] and - final: (digital) written exam or oral examination [100%]
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	RT, DSS

Mandatory Prerequisites	Lab requires attendance in the amount of: 3 Termine
Recommended Literature	▪ Krah, Jens Onno, Vorlesungsskript MC ▪ Krah, Jens Onno: Vorlesungsskript RT (Download) ▪ Handbuch ServoStar 300: www.danahermotion.net ▪ Schultz, G.: Regelungstechnik, Oldenbourg Verlag, München-Wien ▪ Lutz, Wendt: Taschenbuch der Regelungstechnik, Verlag Harri Deutsch
Use of the Module in Other Study Programs	▪ DMC in Master Elektrotechnik 2020 ▪ DMC in Master Elektrotechnik und Informationstechnik 2024 ▪ DMC in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.9 DSP - Digital Signal Processing

Module ID	DSP_MaTIN2024
Module Name	Digital Signal Processing
Type of Module	Elective Modules
Recognized Course	DSP - Digital Signal Processing
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

Design, analyse and implement DSP systems in soft and hardware considering computational complexity and hardware resource limitation, by a thorough understanding of the theoretical concepts, especially frequency domain analysis, and practical implementation of DSP systems in software using Python and on microprocessors, to be able to design, select, use and apply actual and future DSP systems for various signal processing application in commercial products.

Module Contents**Lecture / Exercises**

Signals, Systems and Digital Signal Processing

Basic Elements of DSP Systems

Classification of Signals

Continuous-Time and Discrete-Time Signals

Deterministic and Random Signals

Even and Odd Signals

Periodic and Aperiodic Signals

Energy and Power of Signals

Some Fundamental Signals

Discrete-Time Linear Time-Invariant Systems

Difference Equations

Discrete-Time Convolution

Unit-Pulse and Impulse Response

Basic Systems Properties: Causality, Stability, Memory

Ideal Sampling and Reconstruction

Ideal Sampling and the Sampling Theorem

Aliasing

Fourier-Transform of Discrete-Time Signals

Eigenfunctions of Discrete-Time LTI Systems

Frequency response of Discrete-Time LTI Systems

The Fourier-Transform of Discrete-Time Signals

Ideal Continuous-Time Filters

Discrete Fourier-Transform

Sampling the DTFT

The DFT and the Inverse DFT

The Fast Fourier Transform

Radix-2 FFT Algorithms

Linear Convolution Using the FFT

Overlap-And-Add

Random Signals

Review of Probability and Random Variables

Ensemble Averages

Correlation Functions

Stationary and Ergodic Processes

Power Spectral Density

Transmission of Random Signals over LTI Systems

Advanced Sampling Techniques

Quantization and Encoding

Sampling of Bandpass Signals

Sampling of Random Signals

Sample Rate Conversion

Sample Rate Reduction by an Integer Factor

Sample Rate Increase by an Integer Factor

Sample Rate Conversion by a Rational Factor

Oversampling and Noise Shaping

Students understand the fundamentals of discrete-time signals and systems

Students can analyse the frequency content of a given signal using the appropriate Fourier-Transform and methods for spectrum estimation

Analysis of discrete-time LTI Systems

Students can calculate the output signal via convolution

Students can determine the frequency response of a given system

Students can characterize a given system in the frequency domain and in the z-domain

Implementation of discrete-time LTI systems
Students can implement the convolution sum in software
Students can implement different structures for IIR systems in software
Students can use the FFT to implement an FIR system

Lab

Review of Probability and Random Variables
Moments, Averages and Distribution Functions
Random Signals
Ensemble Averages
Correlation Functions
Stationary and Ergodic Processes
Power Spectral Density
Transmission of Random Signals over LTI Systems
Combating noise
Remove or suppress high-frequency noise from low-pass signals
Ability to trade-off different methods for digital coding of speech and audio signals

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights

- accompanying: (intermediate) certificate [ungraded] and
- final: oral examination [100%]

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

No formal requirements, but students will be expected to be familiar with:
Basic Knowledge of Signals and Systems: Continuous-Time LTI-Systems and Convolution, Fourier-Transform
Basic Knowledge of Probability and Random Variables

Mandatory Prerequisites

- Lab requires attendance in the amount of: 8 Termine
- Participation in final examination only after successful participation in Lab

Recommended Literature

- John G. Proakis and Dimitris K. Manolakis. Digital Signal Processing (4th Edition). Prentice Hall, 2006.
- Alan V. Oppenheim, Ronald W. Schaffer. Discrete-Time Signal Processing (3rd Edition). Prentice Hall, 2007.
- Vinay Ingle and John Proakis. Digital Signal Processing using MATLAB. Cengage Learning Engineering, 2011.

Use of the Module in Other Study Programs

- DSP in Master Communication Systems and Networks 2020
- DSP in Master Communication Systems and Networks 2024
- DSP in Master Technische Informatik 2020

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.10 ESD - Embedded Systems Design

Module ID	ESD_MaTIN2024
Module Name	Embedded Systems Design
Type of Module	Elective Modules
Recognized Course	ESD - Embedded Systems Design
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Markus Cremer/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Markus Cremer/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden können die Machbarkeit der Entwicklung einer Produktidee im Bereich der Embedded Systems in Bezug auf praktische Realisierbarkeit, Aufwand, Zeit und Kosten und mit vorausschauendem Blick auf den gesamten Entwicklungsprozess sicher beurteilen. Hierzu setzen sie, ausgehend von einer eigenen Produktidee, Methoden und Hilfsmittel (z.B. Software-Tools, Konzepte, Best-Practices, v.a. auch Hardwareentwicklung) eines typischen industriellen Entwicklungsprozesses für Embedded Systems eigenständig praktisch um. Später sind die Studierenden in der Lage, diesen gesamten Entwicklungsprozess in der Industrie oder in Forschungsprojekten autonom zu bewerten und umzusetzen.

Module Contents

Lecture / Exercises

The exact content will not be determined until the summer semester 2025.

Project

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

Examination Types with Weights	▪ accompanying: project work [100%]
---------------------------------------	---

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

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

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

Recommended Prerequisites	Basic knowledge of electrical engineering (simple analog and digital circuits) Basic knowledge of embedded systems (basics of microcontrollers incl. implementation of firmware)
----------------------------------	---

Mandatory Prerequisites

Recommended Literature	▪ Murti, K. (2022). Design Principles for Embedded Systems. Springer Singapore. https://doi.org/10.1007/978-981-16-3293-8 ▪ Schmidt, R., Hauschild, D., & Kluge, I. (2024). Elektronik Design: Theorie und Praxis. Elektronik Design: Theorie Und Praxis. https://doi.org/10.1007/978-3-662-68676-8 ▪ Ünsalan, C., Gürhan, H. D., & Yücel, M. E. (2022). Embedded system design with ARM Cortex-M microcontrollers: Applications with C, C++ and MicroPython. Embedded System Design with ARM Cortex-M Microcontrollers: Applications with C, C++ and MicroPython, 1–569. https://doi.org/10.1007/978-3-030-88439-0 ▪ Morshed, B. I. (2021). Embedded systems - A hardware-software co-design approach: Unleash the power of arduino! In Embedded Systems - A Hardware-Software Co-Design Approach: Unleash the Power of Arduino! Springer International Publishing. https://doi.org/10.1007/978-3-030-66808-2 ▪ Marwedel, P. (2021). Embedded System Design. https://doi.org/10.1007/978-3-030-60910-8 ▪ Lienig, J., & Scheible, J. (2020). Fundamentals of Layout Design for Electronic Circuits. Fundamentals of Layout Design for Electronic Circuits. https://doi.org/10.1007/978-3-030-39284-0
Use of the Module in Other Study Programs	▪ ESD in Master Communication Systems and Networks 2024 ▪ ESD in Master Elektrotechnik und Informationstechnik 2024 ▪ ESD in Master Medientechnologie 2024
Permanent Links to Organization	<u>ILU course</u>
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.11 ETH - Ethik

Module ID	ETH_MaTIN2024
Module Name	Ethik
Type of Module	Elective Modules
Recognized Course	ETH - Ethics
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	NN/Lehrbeauftragter

Learning Outcome(s)

Die Studierenden sind in der Lage, ein digitales Produkt gemäß aktueller Anforderungen an Sicherheit und Privatheit zu gestalten. Sie können indem sie

- die technischen, rechtlichen und ethischen Grundlagen von Datenschutz, Datensicherheit und Privacy verstehen,
 - Basistechnologien zu Authentifizierung und Autorisierung kennen und anwenden können,
 - Zielkonflikte zwischen funktionalen Anforderungen und ELSI (ethical / legal / social implications) erkennen und mit Hilfe von Analysemethoden bewerten, sowie
- daraus Schlussfolgerungen für die Hard- und Softwarearchitektur ziehen, indem etwa Sicherheits- / Privatsphärenaspekte in das Produkt integriert werden,
- so dass sie marktfähige und ethisch-rechtliche einwandfreie digitale Produkte erstellen können.

An ausgewählten Problemen zur Fragestellung "Ethische Fragestellungen im Coding", die in der wissenschaftlichen Literatur beschrieben sind, führen die Studierenden Untersuchungen durch, z.B. durch eigene weitere Literaturrecherche, Interviews mit Akteuren im C&C-Umfeld usw. Sie erarbeiten sich damit die Fähigkeit, ethische Problemstellungen in ihrer beruflichen Praxis zu erkennen und sie diskutieren zu können. Dazu betreiben sie

- Literaturrecherche,
- Diskussion digitaler Produkte unter ethischen Aspekten,
- Interviews mit Akteuren im technisch-informatischen-Umfeld usw.

Sie erarbeiten sich damit die Fähigkeit, eine auf ethischen Prinzipien beruhende Folgenabschätzung in ihrer beruflichen Praxis vorzunehmen und damit als "Global Citizen" verantwortungsvoll handeln zu können.

Module Contents

Lecture / Exercises

- technical, regulatory and ethical foundations of data protection, security and privacy
- analytical methods for solving conflicting interests between system requirements and ELSi (ethical / legal / social implications)
- ethics' basic concepts

Seminar

- application of ethics on concrete scenarios

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

Examination Types with Weights	▪ accompanying: technical discussion and ▪ final: oral contribution
---------------------------------------	--

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

Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	none
Mandatory Prerequisites	
Recommended Literature	
Use of the Module in Other Study Programs	ETH in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.12 FP - Forschungsprojekt

Module ID	FP_MaTIN2024
Module Name	Forschungsprojekt
Type of Module	Mandatory Module
Recognized Course	FP - Forschungsprojekt
ECTS credits	10
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers

Learning Outcome(s)

Studierende untersuchen und lösen eine wissenschaftliche Problemstellung, indem sie

- selbständig den aktuellen Stand der Wissenschaft auf einem Fachgebiet durch Literaturrecherche erarbeiten,
- ein eigenes Projekt in Abstimmung mit Kollegen planen, durchführen und kontrollieren,
- das gegebene Problem selbständig (oder im Team) mit wissenschaftlichen Methoden untersuchen und lösen,
- im Studium erworbenes Fachwissen auf Problemstellung anwenden und hierbei vertiefen,
- eigene Lösung mit alternativen Lösungsmöglichkeiten vergleichen,
- erstellte Lösung in Gesamtzusammenhang einordnen und aus fachlicher und gesellschaftlicher Sicht kritisch bewerten und
- den Stand der Wissenschaft, die fachlichen Grundlagen, die gewählte Lösung und ihre Bewertung gegenüber den weiteren möglichen Lösungsalternativen klar und nachvollziehbar in schriftlicher auf Englisch Form darstellen sowie
- mündlich auf Englisch präsentieren und verteidigen.

um wissenschaftliche Methoden in folgenden Modulen, insbesondere der Masterarbeit, und späteren Berufsleben anwenden zu können.

Module Contents

Research Project

Teaching and Learning Methods	Research Project
Examination Types with Weights	▪ accompanying: term paper [100%]
Workload	300 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	288 Hours
Recommended Prerequisites	
Mandatory Prerequisites	
Recommended Literature	
Use of the Module in Other Study Programs	FP in Master Technische Informatik 2020
Specifics and Notes	

Last Update

19.7.2025, 14:32:16

6.13 HIM - Advanced Mathematics

Module ID	HIM_MaTIN2024
Module Name	Advanced Mathematics
Type of Module	Elective Modules
Recognized Course	HIM - Advanced Mathematics
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every 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 grundlegende Konzepte und Methoden der Mathematik, die in den Ingenieurwissenschaften benötigt werden (K. 8). Die Abstraktion und mathematischen Formalisierung von Problemen soll erlernt und angewendet werden (K. 2). Die Studierenden lernen die Anwendung mathematischer Methoden (K. 16). Es soll die Anwendung statistischer Verfahren und die Begründung wissenschaftlicher Aussagen erlernt werden (K. 17).

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.

Wozu: Fortgeschrittene Mathematik-Kenntnisse (beispielweise in Vektoranalysis, Statistik und Optimierung) werden in mehreren Modulen des Studiengangs benötigt. Mathematische Methoden sind essentiell für Ingenieure, die wissenschaftlich arbeiten und wissenschaftliche Erkenntnisse anwenden und erweitern (HF2).

Module Contents**Lecture / Exercises**

A combination of:

- Vector Analysis
- Probability Theory, Statistics and Multivariate Statistics
- Stochastic processes
- Optimization

Vector Analysis

- Vector Spaces
- Scalar and Vector Functions
- Differential Operators
- Line Integrals
- Double Integrals
- Triple Integrals
- Change of Variables
- Surface Integrals
- Divergence Theorem
- Theorem of Stokes
- Maxwell Equations

Probability and Statistics

- Descriptive Statistics
- Two-dimensional Data
- Simple Linear Regression
- Probability Spaces
- Random Variables
- Expectation, Variance, Moments
- Jointly Distributed Random Variables
- Independent Random Variables
- Covariance
- Binomial Random Variable
- Poisson Random Variable
- Uniform Random Variable
- Normal Random Variable
- Chi-Square Distribution
- t-Distribution
- Central Limit Theorem
- Distributions of Sampling Statistics
- Confidence Intervals
- Hypothesis Testing
- t-Test, f-Test, Chi-Square Test
- Overview of various Tests

Multivariate Statistics

- Analysis of multidimensional data
- Multivariate Random Variables
- Matrix decompositions, Singular Value Decomposition (SVD)
- Factor analysis, Principal Component Analysis (PCA)
- Multiple Linear Regression

Stochastic Processes

- Discrete and continuous time processes
- Random walk
- Markov chain
- Poisson process
- Queuing theory

Optimization

- Linear Programming
- Unconstrained Optimization: Gradient method, Newton's method, Trust Region method
- Constrained Optimization: Karush–Kuhn–Tucker (KKT) conditions, Lagrange multipliers, Penalty and Barrier functions
- Special optimization problems: Mixed Integer Nonlinear Programming, Nonlinear Stochastic Optimization

-

Teaching and Learning Lecture / Exercises**Methods**

Examination Types with Weights ■ final: (digital) written exam or oral examination [100%]

Workload 150 Hours

Contact Hours 34 Hours $\pm$ 3 SWS

Self-Study 116 Hours

Recommended Prerequisites Differential and integral calculus and linear algebra (Bachelor-level mathematics)

Mandatory Prerequisites

Recommended Literature

- K. Burg, H. Haf, F. Wille, A. Meister, Vektoranalysis - Höhere Mathematik für Ingenieure, Naturwissenschaftler und Mathematiker, Springer Vieweg
- E. Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons
- L. Papula, Mathematik für Ingenieure und Naturwissenschaftler Band 3, Springer Vieweg
- R. E. Walpole, R. H. Myers, S. L. Myers, K. Ye, Probability & Statistics for Engineers & Scientists, Prentice Hall
- S. M. Ross, Probability and Statistics for Engineers and Scientists, Elsevier
- S. M. Ross, Stochastic Processes, John Wiley & Sons
- U. Krengel, Einführung in die Wahrscheinlichkeitstheorie und Statistik
- A. Koop, H. Moock, Lineare Optimierung, Springer
- R. Reinhardt, A. Hoffmann, T. Gerlach, Nichtlineare Optimierung, Springer
- M. Ulbrich, S. Ulbrich, Nichtlineare Optimierung, Birkhäuser

Use of the Module in Other Study Programs

- HIM in Master Communication Systems and Networks 2020
- HIM in Master Communication Systems and Networks 2024
- HIM in Master Elektrotechnik 2020
- HIM in Master Elektrotechnik und Informationstechnik 2024
- HIM in Master Technische Informatik 2020

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.14 IBD - InnoBioDiv

Module ID	IBD_MaTIN2024
Module Name	InnoBioDiv
Type of Module	Elective Modules
Recognized Course	IBD - InnoBioDiv student projects
ECTS credits	5
Language	englisch
Duration of Module	0.5 Semester
Recommended Semester	1-2
Frequency of Course	every term
Module Coordinator	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Uwe Dettmar/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden können in einer Forschungsgruppe ein Experiment teamorientiert planen, durchführen, auswerten und dokumentieren, indem sie auf biologisches und technisches Basiswissen und auf die zur Verfügung gestellten Ressourcen (ein IoT basiertes Mess- und Steuersystem inklusive FarmBot, Sensorik und Aktorik, Materialien und Geräte im Gewächshaus des Instituts für Pflanzenwissenschaften, Checklisten) sowie weitere frei verfügbare Informationsquellen zugreifen, um die Auswirkungen des Klimawandels auf die Wachstumsleistung von Pflanzen und die Biodiversität im Boden erfahrbar zu machen und dadurch Erkenntnisse zu generieren, die für die Gesellschaft im Rahmen des Klimawandels von Relevanz sind.

Module Contents

Seminar

Development of project ideas, discussion and further development of the projects

Project

The students acquire...

- the ability to develop and implement concepts for the adaptation of plants to climate change.
- the ability to plan, conduct and analyze experiments in the fields of plant physiology, soil biology and technology.
- the ability to statistically evaluate and present experimental data.
- the ability to present and communicate scientific results.
- the ability to collaborate interdisciplinary and interculturallly and to exchange ideas with students from different MINT research areas.
- Experience in planning and implementing projects and in teamwork

At the end, students will have

- a deep understanding of the interactions between climate parameters, plant growth and soil biodiversity.
- basic knowledge of modern technologies such as robotics, sensor technology and the Internet of Things in the context of plant research.
- an awareness of the importance of sustainability, resource conservation and security of supply in the context of population growth and climate change.

Teaching and Learning Methods

- Seminar
- Project

Examination Types with Weights

- accompanying: project work [100%]

Workload	150 Hours
Contact Hours	23 Hours $\pm$ 2 SWS
Self-Study	127 Hours

Recommended Prerequisites	- fluent English since working intercultural and interdisciplinary teams. - basic knowledge in IoT and robotics desirable - ability to work in a team - basic knowledge in plant biology are not mandatory
Mandatory Prerequisites	▪ Seminar requires attendance in the amount of: 8 Stunden ▪ Project requires attendance in the amount of: 5 meetings for project discussions
Recommended Literature	▪ https://farm.bot/ ▪ Arif, Tarik M.: Deep Learning on Embedded Systems: A Hands-On Approach Using Jetson Nano and Raspberry Pi, Wiley, 2025, ISBN:978-1-394-26927-3 ▪ Agrawal, D. P. (2017). Embedded Sensor Systems. Springer. ▪ Marwedel, Peter: Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things, Springer, 2021, ISBN 978-3-030-60910-8 ▪ L. Urry, S. Wassermann: Campbell Biology AP Edition (12th Edition), Pearson, ISBN-13: 978-0-13-648687-9 ▪ Taiz, L., Møller, I. M., Murphy, A., & Zeiger, E. (2022). Plant Physiology and Development. Oxford University Press.
Use of the Module in Other Study Programs	▪ IBD in Master Communication Systems and Networks 2020 ▪ IBD in Master Communication Systems and Networks 2024 ▪ IBD in Master Elektrotechnik und Informationstechnik 2024 ▪ IBD in Master Medientechnologie 2024 ▪ IBD in Master Technische Informatik 2020
Permanent Links to Organization	<u>Course description in the learning platform</u>
Specifics and Notes	Block course from the beginning of October to mid-November (7 weeks), optional preparation time to build up basic knowledge in the last week of September
Last Update	19.7.2025, 14:32:16

6.15 IIS - Intelligent Information Systems

Module ID	IIS_MaTIN2024
Module Name	Intelligent Information Systems
Type of Module	Elective Modules
Recognized Course	IIS - Intelligent Information Systems
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
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)

Die Studierenden kennen die verschiedenen Möglichkeiten zur Darstellung von Wissen und können die Vor – und Nachteile einer Darstellungsform bewerten.

Die Studierenden erarbeiten sich grundlegende Kenntnisse zur Theorie und Anwendung von deklarativen Programmiersprachen bzw. Regelsystemen.

Die Studierenden kennen gängige Typen von Optimierungs- bzw. Suchproblemen und können geeignete deklarative Lösungsansätze identifizieren.

Die Studierenden kennen die wichtigsten Inferenzmethoden und können diese einordnen bzw. bewerten.

Die Studierenden kennen die Resolutionsmethode und das Verfahren der Unifikation und können diese für eine Problemstellung anwenden.

Die Studierenden kennen die wichtigsten Formen der Operationalisierung deklarativer Ausdrücke und können diese bzgl. ihrer Effizienz bei einem Lösungsansatz bewerten.

Die Studierenden können für reale Problemstellungen eine geeignete Wissensrepräsentation wählen und eine Lösung mit einem deklarativen Programm erarbeiten.

Die Studierenden können aktuelle deklarative Anfragesprachen klassifizieren und hinsichtlich ihrer Ausdrucksmächtigkeit bewerten.

Die Studierenden können mit gängigen deklarativen Programmiersprachen umgehen.

Die Studierenden können Aufgaben in einem kleinen Team lösen.

Die Studierenden können Programmcode verstehen und um Funktionalität erweitern. Sie können das Verhalten einer programmierten Lösung bewerten und durch geeignete Modifikationen verbessern.

Die Studierenden können internationale wissenschaftliche Literatur analysieren, einordnen, in ihren Kontext stellen und präsentieren.

Module Contents**Lecture / Exercises**

Foundations of Knowledge Representation

- First-order logic
- relational, functional, tree-based, graph-oriented fact representation (semantic networks, ontologies)
- rule-based systems

Automatic reasoning and inference methods

- resolution principle (incl. unification)
- forward and backward chaining
- fixpoint semantics

Declarative Programming languages

- Functional programming
- relational (logical) programming , e.g., Prolog, Datalog, SQL and SPARQL

Outlook on current research issues, e.g., query languages, parallel algorithms, distributed systems, combinatorial optimization and language processing.

Students have acquired basic knowledge about methods for representing knowledge, automatic reasoning as well as declarative programming languages. They understand the various ways of operationalizing declarative expressions and are able to realize suitable programming solutions for given problems.

Lab

Representing knowledge by sets of tuples, relations, semantic networks as well as logic-based systems

Implementing calculation problems with a functional programming language (e.g. Haskell) using expressions, algebraic data types, infinite data structures and higher-order functions

Solving search problems with a logical programming language and recursive expressions

Formulating relational queries over knowledge bases (e.g. using SPQAQL or Datalog)

Teaching and Learning

- Lecture / Exercises

Methods

- Lab

Examination Types with Weights

- accompanying: exercise lab [ungraded] and
- final: (digital) written exam or oral examination [100%]

Workload

150 Hours

Contact Hours

45 Hours $\triangleq$ 4 SWS

Self-Study

105 Hours

Recommended Prerequisites

programming skills, knowledge about data structures and algorithms

Mandatory Prerequisites**Recommended Literature**

- G. Hutton: Programming in Haskell, 2nd Ed., Cambridge University Press, 2016
- L. Sterling, E. Shapiro: The Art of Prolog, 2nd Ed., MIT Press, 1994
- Uwe Schöning. Logik für Informatiker. 5. Auflage, Spektrum Akademischer Verlag, 2000
- Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph. Foundations of Semantic Web Technologies. CRC Press 2009.
- S.J. Russell, P. Norvig: Artificial Intelligence. A Modern Approach, 2. Aufl. Prentice Hall, 2003

Use of the Module in Other Study Programs

IIS in Master Technische Informatik 2020

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.16 ITF - IT-Forensik

Module ID	ITF_MaTIN2024
Module Name	IT-Forensik
Type of Module	Elective Modules
Recognized Course	ITF - IT forensics
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	Jürgen Bornemann/Lehrbeauftragter

Learning Outcome(s)

- **WAS** Studierende spüren digitale Beweise auf und stellen Sie zwecks Verwertbarkeit für weiterführende Analysen sicher,
- **WOMIT** indem sie anhand fallbezogener Aufgabenstellungen und mittels forensischer IT-Tools Schwachstellen entdecken und Beweise in Dateisystemen und IT-Infrastrukturen sichern,
- **WOZU** um im Berufsleben Gefahren vermeiden, erkennen und abwehren können und ggf. gutachterlich tätig zu werden.

Module Contents

Lecture / Exercises

Basic concepts of cyber security and digital forensics
 Typical vulnerabilities, threats and risks
 Dangers with mobile systems, home office, WLANs
 Basics and working methods of IT forensics
 Forensic documentation creation
 Common tools for forensic investigations
 Recognize and secure digital evidence
 Open source forensics
 File system forensics
 Forensic analysis of mobile systems
 Vulnerabilities, threats, attacks on network structures
 KALI Linux - Operating System for Vulnerability and Pentesting

Project

Students can work on case-related forensic tasks and incidents independently or in working groups using the knowledge they have acquired. They show how to secure, analyze and document digital evidence.

Teaching and Learning Methods

- Lecture / Exercises
- Project

Examination Types with Weights

- accompanying: project work [100%]

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

**Recommended
Prerequisites**

**Mandatory
Prerequisites**

**Recommended
Literature**

Use of the Module in Other Study Programs	▪ ITF in Master Communication Systems and Networks 2020
	▪ ITF in Master Communication Systems and Networks 2024
	▪ ITF in Master Elektrotechnik 2020
	▪ ITF in Master Elektrotechnik und Informationstechnik 2024
	▪ ITF in Master Medientechnologie 2020
	▪ ITF in Master Medientechnologie 2024
	▪ ITF in Master Technische Informatik 2020

Specifics and Notes

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

6.17 KOGA - Kombinatorische Optimierung und Graphenalgorithmen

Module ID	KOGA_MaTIN2024
Module Name	Kombinatorische Optimierung und Graphenalgorithmen
Type of Module	Elective Modules
Recognized Course	KOGA - Combinatorial Optimization and Graph Algorithms
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Hubert Randerath/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hubert Randerath/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden sind in der Lage Verfahren und Konzepte der Graphentheorie und der Kombinatorischen Optimierung zur Beschreibung und algorithmischen Lösung von Problemstellungen der Informatik, der Technik und des täglichen Lebens anzuwenden.

Sie haben die Fertigkeit Verfahren und Konzepte der Graphentheorie und der Kombinatorischen Optimierung zur Beschreibung und algorithmischen Lösung von Problemstellungen der Informatik, der Technik und des täglichen Lebens anzupassen.

Sie können algorithmische Denk- und Arbeitsweisen wie Komplexität von Problemklassen, Effizienz von Algorithmen und Approximation, die sie induktiv an Optimierungsaufgaben in Netzwerken und gewichteten Graphen erlernt haben, anwenden.

Module Contents

Lecture / Exercises

- Basics of Graph Theory und Combinatorial Optimization
- Minimal Spanning Trees: algorithms of Kruskal, Prim und Tarjan, Greedy algorithms, matroids, Steiner trees, network design
- Linear Programs: structure, modelling, normalization, Simplex algorithm, Theory of Duality
- Weighted Matchings and the Route Inspection Problem: Weighted Matchings in Bipartite Graphs and non-bipartite Graphs, algorithms of Floyd-Warshall and Fleury
- Network Flows: Network Theory Basics, Dinic's algorithms, cost-optimal flows
- selected discrete and combinatorial optimization problems: Travelling Salesman, Channel Assignment Problem, scheduling problems, routing problems

Exercises / Lab

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Exercises / Lab
Examination Types with Weights	▪ accompanying: oral contribution and ▪ final: (digital) written exam or oral examination [100%]
Workload	150 Hours
Contact Hours	57 Hours $\triangleq$ 5 SWS
Self-Study	93 Hours
Recommended Prerequisites	Basic knowledge in graph theory Basic knowledge in algorithmics
Mandatory Prerequisites	▪ Lecture / Exercises requires attendance in the amount of: 1 Vortragstermin ▪ Exercises / Lab requires attendance in the amount of: 1 Termin

**Recommended
Literature**

- Mathematik zum Studienbeginn, Arnfried Kemnitz, Springer Spektrum Verlag
- Algorithmische Graphentheorie, Volker Turau und Christian Weyer, De Gruyter Verlag
- Graphentheoretische Konzepte und Algorithmen, Sven Krumke und Harald Noltemeier, Springer Vieweg Verlag
- Einführung in die angewandte Wirtschaftsmathematik, Jürgen Tietze, Springer Spektrum Verlag
- Graph Algorithms - Practical Examples in Apache Spark & Neo4j, Mark Needham and Amy Hodler, O'Reilly Verlag

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

KOGA in Master Technische Informatik 2020

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.18 KOLL - Kolloquium zur Masterarbeit

Module ID	KOLL_MaTIN2024
Module Name	Kolloquium zur Masterarbeit
Type of Module	Mandatory Module
Recognized Course	MAKOLL - Colloquium
ECTS credits	3
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers
Learning Outcome(s)	- Darstellung von Forschungsergebnissen in einer Präsentation in vorgegebenem engen zeitlichen Rahmen - Fachliche und außerfachliche Bezüge der eigenen Arbeit darstellen und begründen - Eigene Lösungswege und gewonnene Erkenntnisse darstellen und diskutieren
Module Contents	
Colloquium	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
Teaching and Learning Methods	Colloquium
Examination Types with Weights	▪ final: Colloquium [100%]
Workload	90 Hours
Contact Hours	0 Hours $\triangleq$ 0 SWS
Self-Study	90 Hours
Recommended Prerequisites	
Mandatory Prerequisites	▪ Module MAA: Die Masterarbeit 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 Master Communication Systems and Networks 2020 ▪ KOLL in Master Communication Systems and Networks 2024 ▪ KOLL in Master Elektrotechnik 2020 ▪ KOLL in Master Elektrotechnik und Informationstechnik 2024 ▪ KOLL in Master Medientechnologie 2020 ▪ KOLL in Master Medientechnologie 2024 ▪ KOLL in Master Technische Informatik 2020
Specifics and Notes	See also examination regulations §29.
Last Update	19.7.2025, 14:32:16

6.19 KRY - Cryptography

Module ID	KRY_MaTIN2024
Module Name	Cryptography
Type of Module	Elective Modules
Recognized Course	KRY - Cryptography
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Heiko Knospe/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Heiko Knospe/Professor Fakultät IME

Learning Outcome(s)

Was: Die Studierenden lernen die mathematischen Grundlagen der Kryptographie kennen. Es werden Kenntnisse der wichtigsten kryptographischen Methoden und Algorithmen vermittelt (HF 1). Die Studierenden verstehen verschiedene Arten von Sicherheitsanforderungen und analysieren die Sicherheit von kryptographischen Verfahren.

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 Kryptographie bearbeitet.

Wozu: Kryptographie wird eingesetzt um die grundlegenden Ziele der Informationssicherheit zu erreichen. Die Studierenden lernen die Implementierung und Anwendung von kryptographischen Algorithmen und entwickeln Konzepte um Systeme, Netzwerke und Anwendungen gegen Angriffe zu sichern (HF 2).

Module Contents

Lecture / Exercises

- * Mathematical Fundamentals
- * Encryption Schemes and Definitions of Security
- * Elementary Number Theory
- * Algebraic Structures
- * Block Ciphers
- * Stream Ciphers
- * Hash Functions
- * Message Authentication Codes
- * Public-Key Encryption and the RSA Cryptosystem
- * Key Establishment
- * Digital Signatures
- * Elliptic Curve Cryptography
- * Outlook: Post-quantum cryptography

Lab

- Solve mathematical and cryptographical problems in Python / SageMath: working with large integers and residue classes, factorization, primality and prime density, RSA key generation and encryption / decryption, Diffie-Hellman key exchange.
- Write code to encrypt and decrypt files using the AES block cipher and different operation modes. Analyze the statistical properties of AES ciphertext.
- Write code for RSA key generation, key encapsulation / decapsulation and hybrid encryption / decryption.

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

Examination Types with Weights	■ accompanying: exercise lab [ungraded] and ■ final: (digital) written exam or oral examination [100%]
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	Mathematics (Bachelor level) and programming skills.
Mandatory Prerequisites	■ Lab requires attendance in the amount of: 3 Termine ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ M. Bellare, P. Rogaway, Introduction to Modern Cryptography, UCSD CSE ■ H. Delfs, H. Knebl, Introduction to Cryptography, Springer ■ S. Goldwasser, M. Bellare, Lecture Notes on Cryptography, MIT ■ J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography, Springer ■ J. Katz, Y. Lindell, Introduction to Modern Cryptography, CRC Press ■ H. Knospe, A Course in Cryptography, American Mathematical Society ■ C. Paar, J. Pelz, Understanding Cryptography. Springer ■ N.P. Smart, Cryptography Made Simple, Springer ■ K. H. Rosen, Discrete Mathematics and its Applications, McGraw-Hill ■ V. Shoup, A Computational Introduction to Number Theory and Algebra, Cambridge University Press
Use of the Module in Other Study Programs	■ KRY in Master Communication Systems and Networks 2020 ■ KRY in Master Communication Systems and Networks 2024 ■ KRY in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.20 LCSS - Large and Cloud-based Software-Systems

Module ID	LCSS_MaTIN2024
Module Name	Large and Cloud-based Software-Systems
Type of Module	Elective Modules
Recognized Course	LCSS - Large and Cloud-based Software-Systems
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. René Wörzberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. René Wörzberger/Professor Fakultät IME

Learning Outcome(s)

Students are capable of

- designing architectures for complex and mission critical enterprise software systems,
- implementing these systems and
- operate them in the Cloud

by

- knowing and trading conflicting interests and concerns of stakeholders,
- knowing quality attributes and their trade-offs,
- specifying architecturally significant requirements in quality attribute scenarios,
- analysing design decisions with respect to their effects on quality attributes and stake-holder interests and concerns,
- presenting and documenting architectures by means of suitable views, notations and tools,
- applying methods (like RESTful API design) and tools in order to implement design decisions,
- using cloud resources like virtual machines, containers and storages in order to operate a system in the cloud,

in order to

- be able to produce long-term usable software systems in subsequent lectures and projects and
- to be able to act as an IT architect, e.g. in an IT department of a larger enterprise.

Module Contents**Lecture / Exercises**

- Formally sound handling of quality requirements for availability, performance, capacity and cost efficiency
- Advantages and disadvantages of basic system architecture styles, such as microservice architectures
- Scaling of systems and individual tiers, also with regard to possible deployment strategies such as canary or AB deployment, as well as associated load balancing strategies (e.g. consistent hashing)
- Advanced applications of virtualization, in particular container virtualization and orchestration, for example with Docker and Kubernetes
- Selection of suitable communication patterns and protocols, in particular HTTP and derivatives such as websockets, server-sent events and gRPC
- Selection of appropriate API technologies and design philosophies such as REST and GraphQL
- Use of basic security protocols such as TLS, OAuth2, JWT and OpenID Connect
- Asynchronous, event-driven communication via messaging and streaming platforms such as Apache Kafka
- Selection of suitable database models (relational, key-value-, graph-, document-oriented), necessary consistency level, as well as sharding using the example of PostgreSQL, Neo4J, Apache Cassandra and Redis
- Strategies for caching data, in particular HTTP responses (web caching).

Project

- Be able to formulate and present a research question in the topic area of the course.
- Design an application prototype that serves to investigate the research question.
- Develop the application prototype and run it in the cloud
- Design and conduct test scenarios and experiments with the application prototype to answer the research question.

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

Examination Types with Weights	▪ accompanying: project work [60%] and ▪ final: (digital) written exam or oral examination [40%]
---------------------------------------	---

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

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

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

Recommended Prerequisites	- advanced programming skills - basic knowledge of web technologies - basic knowledge of databases - basic knowledge in software architectures - basic knowledge of Unified Modeling Language (UML)
----------------------------------	---

Mandatory Prerequisites	Project requires attendance in the amount of: 4 Termine
--------------------------------	---

Recommended Literature	▪ Lecture Notes Large and Cloud-based Software Systems ▪ H. Adkins et al.: Building Secure and Reliable Systems, O'Reilly Media, 2020 ▪ I. Gregorik: High Performance Browser Networking, O'Reilly Media, 2013 ▪ M. Kleppmann: Designing Data-Intensive Applications, O'Reilly Media, 2017
-------------------------------	---

Use of the Module in Other Study Programs	▪ LCSS in Master Communication Systems and Networks 2024 ▪ LCSS in Master Medientechnologie 2024 ▪ LCSS in Master Technische Informatik 2020
--	--

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

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.21 MAA - Masterarbeit

Module ID	MAA_MaTIN2024
Module Name	Masterarbeit
Type of Module	Mandatory Module
Recognized Course	MAA - Master thesis
ECTS credits	27
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers

Learning Outcome(s)

Das Modul vermittelt folgende Kenntnisse und Fertigkeiten:

- Komplexe Aufgabenstellungen beurteilen
- Selbstständiges Verfassen eines längeren wissenschaftlichen Textes
- Gute Praxis des wissenschaftlichen Arbeitens anwenden
- Darstellung von Forschungsergebnissen in Form eines wissenschaftlichen Artikels nach den Vorgaben gängiger Fachzeitschriften bzw. Konferenzen
- Selbstständiges und systematisches Bearbeiten einer komplexen ingenieurwissenschaftlichen Aufgabenstellung unter Verwendung wissenschaftlicher Methoden
- Lösungsstrategien entwickeln und umsetzen
- Wissenschaftliche Literatur recherchieren und auswerten
- Eigene Arbeit bewerten und einordnen

Individuelle Vereinbarung des Studierenden mit einem Dozenten der MT bzw. F07 über eine qualifizierte Ingenieur Tätigkeit mit einer studiengangsbezogenen Aufgabenstellung mit wissenschaftlichem Anspruch. Die Masterarbeit kann auch extern in einer Forschungsorganisation, einem Wirtschaftsunternehmen o.ä. durchgeführt werden. Die Betreuung erfolgt durch den Dozenten. Die Masterarbeit adressiert die Entwicklung komplexer Medientechnologien unter interdisziplinären Bedingungen (HF1) und das wissenschaftliche Arbeiten um wissenschaftliche Erkenntnisse zu erweitern (HF2)."

Module Contents

Thesis

The Master'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	Thesis
Examination Types with Weights	▪ final: Thesis [100%]
Workload	810 Hours
Contact Hours	0 Hours $\pm$ 0 SWS
Self-Study	810 Hours
Recommended Prerequisites	See examination regulations §26
Mandatory Prerequisites	see exam regulations §26 paragraph 1

**Recommended
Literature**

Use of the Module in Other Study Programs	▪ MAA in Master Communication Systems and Networks 2020 ▪ MAA in Master Communication Systems and Networks 2024 ▪ MAA in Master Elektrotechnik 2020 ▪ MAA in Master Elektrotechnik und Informationstechnik 2024 ▪ MAA in Master Medientechnologie 2020 ▪ MAA in Master Medientechnologie 2024 ▪ MAA in Master Technische Informatik 2020
--	--

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
--------------------	---------------------

6.22 MCI - Mensch-Computer-Interaktion

Module ID	MCI_MaTIN2024
Module Name	Mensch-Computer-Interaktion
Type of Module	Elective Modules
Recognized Course	MCI - Human Computer Interaction
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Jonas Schild/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Jonas Schild/Professor Fakultät IME

Learning Outcome(s)

WAS:

Das Modul vermittelt folgende Kenntnisse und Fertigkeiten:

- Grundlagen der Mensch-Computer-Interaktion: Definitionen, Normen, Modelle, Prinzipien
- Interaktive Systeme aus Hard- und Software konzipieren, implementieren und analysieren
- User Experience verstehen und Prinzipien des UX Engineerings anwenden
- Wiss. Fragestellungen vor einem Forschungshintergrund der HCI entwickeln
- Geeignete Nutzerstudien nach wiss. und ethischen Kriterien konzipieren, planen und durchführen
- statistische und deskriptive Daten wissenschaftlich analysieren, veranschaulichen und diskutieren
- in heterogenen Teams zusammenarbeiten, sich koordinieren und präsentieren

WOMIT:

Die Kompetenzen werden zunächst über eine Vorlesung durch die Dozenten vermittelt und danach im Praktikum anhand konkreter Aufgabenstellung von den Studierenden vertieft. Im seminaristischen Teil der Lehrveranstaltung recherchieren die Studierenden zu vorgegebenen Themen anhand von Fachartikeln und weiteren Informationsquellen über neue Konzepte der Mensch-Computer Interaktion und stelle diese dar in einer Präsentation dar.

WOZU:

Die Studierenden erlernen das eigenständige Durchführen von Forschungsprozesse auf dem Gebiet der Mensch-Computer-Interaktion, um im interdisziplinären Team auf Grundlage von selbst entwickelten komplexen, interaktiven Systemen (HF1) aktuelle Fragestellungen aus dem Bereich der Mensch-Computer-Interaktion wissenschaftlich untersuchen (HF2) und dabei die Effektivität und Wirkung von interaktiven Systemen auf Nutzende testen und einschätzen zu können (HF4).

Module Contents**Lecture**

Models and design principles of interactive systems
 Principles of context-, task- and user-oriented development of interactive systems
 Basics of barrier-free access to interactive systems
 Relevant standards and guidelines: EN ISO 9241, ISO 14915, HHS
 Control options: Dedicated input/output devices, voice control, gesture control
 Best Practices and Style Guides: Desktop / Web / Mobile / Hybrid Applications
 Usability evaluation (analytical/empirical, heuristics, expert interviews, focus groups, user studies)
 Evaluation methods (thinking aloud, eye-tracking, (semi-)structured interviews)

Experimental Research: Research Question, Hypotheses, Errors of 1st and 2nd Kind
 Experiment Design: Between Group, Within Group, Split-Plot, Reliability of Experimental Results
 Statistical analysis: scale levels, descriptive statistics, T-tests, ANOVA, regression, correlation
 Surveys: sampling and sample selection, sources of error, questionnaires, evaluation of surveys

Lab

Capturing and understanding textual tasks
 Recording tasks and creating models from them
 Implementing UI components on the basis of the models created
 Testing and securing developments
 Checking and evaluating work results of comolitions
 Applying MCI research methods and terminology

Seminar**Teaching and Learning Methods**

- Lecture
- Lab
- Seminar

Examination Types with Weights

- accompanying: project work [100%]

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites none

Mandatory Prerequisites

- Lab requires attendance in the amount of: 2 Termine
- Seminar requires attendance in the amount of: Vorträge und Schlusspräsentation

Recommended Literature

- A. M. Heinecke: Mensch-Computer-Interaktion, Basiswissen für Entwickler und Gestalter, 2. Auflage, Springer, 2011
- B. Shneiderman, C. Plaisant: Designing the User Interface: Strategies for Effective Human-Computer Interaction, Addison Wesley, 2009
- S. Swink: Game Feel: A Game Designer's Guide to Virtual Sensation, Morgan Kaufmann Game Design Books, 2008
- T. Sylvester: Designing Games: A Guide to Engineering Experiences, O'Reilly, 2013
- J. Lazar, J.H. Feng, H. Hochheiser, Research Methods in Human-Computer-Interaction, Wiley, 2012

Use of the Module in Other Study Programs

- MCI in Master Medientechnologie 2020
- MCI in Master Medientechnologie 2024
- MCI in Master Technische Informatik 2020

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.23 MLWR - Maschinelles Lernen und wissenschaftliches Rechnen

Module ID	MLWR_MaTIN2024
Module Name	Maschinelles Lernen und wissenschaftliches Rechnen
Type of Module	Elective Modules
Recognized Course	MLWR - Machine Learning and Scientific Computing
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Beate Rhein/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Beate Rhein/Professor Fakultät IME

Learning Outcome(s)

Was:

fortgeschrittene Methoden des maschinellen Lernens auf typische Datensätze der technischen Informatik anwenden
Fallstricke des Maschinellen Lernens in der Vorgehensweise erkennen
für eine Aufgabenstellung das geeignete Verfahren bestimmen und anwenden können
Qualität von Datensätzen beurteilen und verbessern
Datenschutzgesetze kennen
weit verbreitete Software des maschinellen Lernens anwenden

Womit:

Die Methoden werden anhand eines Vortrags oder per Lernvideos vermittelt und in Vorlesung und Übung direkt angewendet. Jeder Student wird ein Projekt durchführen (je nach Anzahl der Studierenden in Gruppenarbeit), bei der er sich Teile des Stoffes selber erarbeitet.

Wozu:

Maschinelles Lernen 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**

Approximation methods
metamodeling
regression

Multi-criteria optimization
formulation
Pareto front
algorithms
visualization

Advanced Cluster Analysis

Association Pattern Mining

Outlier Detection

Advanced classification procedures

possibly text recognition, web mining, time series analysis

Lab

Apply and program methods of approximation, multicriteria optimization or machine learning
efficiently implement numerical methods
Evaluate the complexity of algorithms

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights

- accompanying: project work [20%] and
- final: oral examination [80%]

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites Basic knowledge of probability theory and machine learning

Mandatory Prerequisites

- Lecture / Exercises requires attendance in the amount of: 6 Stunden
- Lab requires attendance in the amount of: 2 Termine
- Participation in final examination only after successful participation in Lab

Recommended Literature

- A. Geron: Hand-on Machine Learning, O'Reilly Verlag
- J. Alammar: Hands-on Large Language Models, O'Reilly Verlag

Use of the Module in Other Study Programs

- MLWR in Master Communication Systems and Networks 2020
- MLWR in Master Communication Systems and Networks 2024
- MLWR in Master Elektrotechnik 2020
- MLWR in Master Elektrotechnik und Informationstechnik 2024
- MLWR in Master Medientechnologie 2024
- MLWR in Master Technische Informatik 2020

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.24 NGN - Next Generation Networks

Module ID	NGN_MaTIN2024
Module Name	Next Generation Networks
Type of Module	Elective Modules
Recognized Course	NGN - Next Generation Networks
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

What?

Understanding architectures and service signalling in Next Generation Networks (All-IP Networks). Competences to evaluate, analyze, design, implement and test NGN components and service areas with heterogeneous service requirements.

How?

Based on Bachelor-level competences on IP networking and services, students learn standards, design principles, architectures and sample implementations of Next Generation Networks and services in lectures and exercises. In a small team and organized as semester project, students develop their own NGN service solution, optionally based on existing systems, and learn how to design, implement and analyze their own service solution.

What for?

To be able to design, analyze, select, use and apply actual and future network services, based on All-IP networks for enterprise networks, telecommunication networks and mobile networks.

Module Contents

Lecture / Exercises

Achieve basic understanding and implementation knowledge on Next Generation Network (NGN) definition by ITU-T, IP Multimedia Subsystem by 3GPP, and ETSI, and Next Generation Internet (NGI) definition by IETF, ITU-T standards, Multimedia Services in NGN, VoIP, Video-over-IP, RTP encapsulation, Service Signaling, SIP protocol, SIP Digest Authentication, SDP service description and capabilities, SIP servers, Session Border Controller (SBC), SIP Gateway Technologies, SIP routing, NAT Gateways, NAT solution, SRR, STUN, TURN, IMS in mobile networks, IMS in fixed-line networks, VoIP in enterprise networks. IMS in virtualized core network.

Students evaluate requirements for NGN services and plan, implement and analyze NGN services based on SIP signalling or alternative signalling protocols. They are competent in functional analysis and troubleshooting by deep packet inspection (DPI) protocol analysis. They evaluate the performance of NGN services in terms of timing, throughput, latency and delays, jitter, robustness in case of packet errors, and security aspects.

Lab

Naming, structuring and classifying concepts and technologies for NGNs or NGIs. Demonstrate network analysis techniques and tools, know methods for NGN services and network planning.

Working on a small project in a tiny team (2-3 team members) on actual technologies in the area of NGN services and NGI services. Set-up an NGN/NGI environment and NGN service, including planning, implementation and evaluation of security aspects and protocol analysis plus performance evaluation.

The results are reviewed during the course period, summarised in a report and presented to the class. Individual project proposals by students are welcome.

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

Examination Types with Weights	■ accompanying: exercise lab [ungraded] and ■ final: (digital) written exam or oral examination [100%]
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	■ Modul BSN: Bachelor Level Networking Knowledge and Skills like taught in BSN. Fundamentals of Networks and Protocols (typically Bachelor Level) Layered Communications and Protocol Stacks (ISO/OSI, IETF TCP/IP, IEEE), LAN, MAN, WAN, Fixed Line and Mobile Network Fundamentals, Data Link-Technologies (Ethernet, WiFi), IP-Networking (IPv4, IPv6), IP Routing Protocols (static Routes, RIP, OSPF, BGP), Transport Protocols (TCP (incl. Flow Control / Congestion Control), UDP) and Port Numbers, Application Protocols (HTTP, Request-Response Pattern, Publish-Subscribe Pattern). ■ Bachelor-level knowledge of protocols and layer models, Internet protocols (UDP, TCP, IP, HTTP, FTP), IP addressing (IPv4, IPv6), routing techniques (IP routing, functionality of a router, routing protocols, RIP, OSPF), transmission systems and layer 2 protocols, Ethernet. ■ Understanding distributed systems and applications, sockets and client/server programming, request-response patterns, publish-subscribe patterns.
Mandatory Prerequisites	■ Lab requires attendance in the amount of: 6 Meilensteintermine und Projektvorstellungen ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ J. Kurose, K. Ross: Computer Networking: A Top-Down Approach, Global Edition, Prentice Hall, 7th ed., 2016 ■ A. S. Tanenbaum, D. J. Wetherall: Computer Networks, Pearson, 5th ed., 2013 ■ U. Trick, F. Weber: SIP und Telekommunikationsnetze: Next Generation Networks und Multimedia over IP – konkret, De Gruyter Oldenbourg Verlag, 4. Auflage 2015 ■ J. F. Durkin: Voice-enabling the Data Network, Cisco Press 2010 ■ G. Camarillo, M.A. García-Martín: The 3G IP Multimedia Subsystem (IMS), John Wiley Verlag, 2006 ■ 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 ■ J. van Meggelen, R. Bryant, L. Madsen: Asterisk: The Definitive Guide: Open Source Telephony for the Enterprise, O'Reilly Media, 5th Ed. 2019
Use of the Module in Other Study Programs	■ NGN in Master Communication Systems and Networks 2020 ■ NGN in Master Communication Systems and Networks 2024 ■ NGN in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.25 PAP - Parallele Programmierung

Module ID	PAP_MaTIN2024
Module Name	Parallele Programmierung
Type of Module	Elective Modules
Recognized Course	PAP - Parallel Programming
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
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)

Medientechnische und interaktive Systeme beinhalten rechenintensive Berechnungen. Um Anforderungen an die Verarbeitung in Echtzeit erfüllen zu können, sind daher Kompetenzen und Wissen über die Grundlagen für die Analyse (HF1, HF2), den Entwurf (HF1, HF2), die Implementierung (HF1, HF2) und die Bewertung (HF1, HF2) paralleler Computerprogramme erforderlich.

Folgende Kenntnisse und Kompetenzen werden im Detail vermittelt:

- Grundlegende Konzepte, Modelle und Technologien der parallel Verarbeitung benennen, strukturieren, einordnen und abgrenzen
- Aufgabenstellungen in Bezug auf die Programmierung paralleler Programme analysieren und strukturieren, einschlägige parallele Hardwarearchitektur zuordnen und auf Paralleldesign übertragen
- Parallele Programme unter Einsatz geeigneter Tools analysieren und Ergebnisse nachvollziehbar darstellen
- Leistungsfähigkeit paralleler Programme abschätzen und analysieren
- Information aus englischen Originalquellen und Standards ableiten

Kenntnisse und Basisfertigkeiten werden in der Vorlesung vermittelt. Begleitend dazu werden in den Übungen Kompetenzen und Fertigkeiten ausgebaut und inhaltliche Themen vertieft.

Module Contents

Lecture

- Basic concepts, models and technologies of parallel processing (parallelism, concurrency, SISD, SIMD, MISD, MIMD, loose- and closely coupled systems, distributed systems)
- Parallel performance measures (speedup, efficiency)
- Architecture of GPUs
- Parallel Algorithms for GPUs

Exercises / Lab

- Analyze and structure tasks related to programming parallel programs, assign relevant parallel hardware architecture and transfer to parallel design
- Implement parallel programs (multicore hardware with threads and GPUs)
- Analyze parallel programs using suitable tools and present results in a comprehensible way
- Estimate and analyze performance of parallel programs
- Derive information from original English sources and standards

Teaching and Learning

Methods

- Lecture
- Exercises / Lab

Examination Types with Weights

- accompanying: exercise lab [ungraded] and
- final: (digital) written exam or oral examination or term paper [100%]

Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	The exercises require programming knowledge and the use of console-oriented programs in Linux-based operating systems.
Mandatory Prerequisites	Exercises / Lab requires attendance in the amount of: 2 Termine
Recommended Literature	■ Wen-mei W. Hwu, David B. Kirk, Izzat El Hajj: Programming Massively Parallel Processors A Hands-on Approach - 4th Edition, 2022 ■ Andrew S. Tanenbaum, Herbert Bos: Modern Operating Systems, 4th Edition, 2015 ■ Jason Sanders: CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Longman, 2010 ■ R. Oechsle: Parallele und verteilte Anwendungen in Java, Hanser, 2011 ■ P. Pacheco: An Introduction to Parallel Programming, Morgan Kaufmann, 2011
Use of the Module in Other Study Programs	■ PAP in Master Medientechnologie 2020 ■ PAP in Master Medientechnologie 2024 ■ PAP in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.26 PHTM - Philosophische Handlungstheorie Master

Module ID	PHTM_MaTIN2024
Module Name	Philosophische Handlungstheorie Master
Type of Module	Elective Modules
Recognized Course	PHTM - Philosophical theory of action
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
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	▪ accompanying: oral contribution [ungraded] and ▪ final: oral examination [ungraded]
Workload	150 Hours
Contact Hours	34 Hours $\triangleq$ 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

PHTM in Master Technische Informatik 2020

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.27 PLET - Projektleitung

Module ID	PLET_MaTIN2024
Module Name	Projektleitung
Type of Module	Elective Modules
Recognized Course	PLET - Project management
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Michael Gartz/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Michael Gartz/Professor Fakultät IME ▪ Prof. Dr. Uwe Oberheide/Professor Fakultät IME

Learning Outcome(s)

Was: Die Studierenden haben organisatorische Kompetenz erworben und können Projekt planen, durchführen, dokumentieren, Produktanforderungen analysieren, Machbarkeit bewerten und Produktqualität planen. Sie können Projektstrukturpläne und Projektzeitpläne erstellen, Projektmeilensteine planen, Projektrisiken erkennen und mildern. Sie können den Einsatz von Personal und Sachressource planen, Reviews planen, Produktverifikation planen.

Die Studierenden haben Projektführungskompetenz erworben und können die Projektsteuerung mit agilen, evolutionären Vorgehensmodellen und dem Timeboxmodell durchführen. Sie können Projektmanagementwerkzeuge einsetzen, den Projektfortschritt überwachen / steuern und Projektergebnisse freigeben. Sie können den Entwicklungsprozess fortlaufend optimieren in unklaren Situationen entscheiden. Sie können den Entwicklungsverlauf dokumentieren, Projektberichte verfassen und verteidigen.

Die Studierenden haben Personalführungskompetenz erworben und können Aufgaben auf Teammitglieder nach individuellen Qualifikationen und Neigungen verteilen.

Sie können die Teambildung fördern, das Team koordinieren und zielorientiert und respektvoll kommunizieren und verbindliche Absprachen treffen und einfordern. Sie können Teamprozesse moderieren, potenzielle Konfliktsituationen erkennen und auflösen und Handlungsalternativen abwägen.

Womit: indem sie die in dem Teamleiter Seminar erlernten Kompetenzen und Fertigkeiten und die in dem Projektleiter-Workshop erlernten Projektleitungs-Tools und Kompetenzen anwenden.

Wozu: um später in den verschiedensten Industriebereichen Projekte mittels agilen, evolutionären Vorgehensmodellen, wie z.B. SCRUM, zu planen, durchzuführen, zu managen und zum Erfolg zu bringen.

Module Contents**Seminar**

Classifying and delimiting terms
explain characteristic properties of development projects
Goal orientation and innovation
Risk of failure
Special organisational form (teamwork)
Limited resources
Limited realization time
abstractly define technical and economic goals in development projects
abstractly define, explain and justify project management tasks
identify and explain basic success and failure factors in project management
unexpected technical problems
insufficient staff qualification
unclear or conflicting requirements
poor project management
Insufficient support from senior management
identify extended challenges arising from a division of labour in project processing
explain selected process models
linear models for business project management
phase model
V-model
agile process models for technical project management
SCRUM
timebox model
classify and compare process models with regard to development duration, organizational aspects, quality and cost aspects
professional quality control
Cost and schedule control in business management
Legal requirements for documentation and traceability of project decisions
characterize basic tasks and expected results in development projects
Planning and control of product quality
Planning and controlling the quality of the development process
overarching legal requirements
industry-specific specifications
company-internal specifications
project risk management
resource management
Documentation of the development process
Specification of the requirements for the product to be developed
Specification of the product design
Product development and manufacturing
product documentation
Verification and validation of the developed product
Product release and product monitoring
Characterize instruments for controlling team processes
plan essential management tasks, milestones and project documents with regard to the course element "Project
carry out essential management tasks mentally and identify project risks with foresight
handle essential project management tools in a target-oriented manner
for project (time) planning
for requirement specifications
Planning team building procedures, deriving expected challenges and meaningful measures

identify potential conflict situations in the team and discuss alternative actions

Project

Lead team

explain to the team members the basic procedure in the project

Capturing and classifying the competencies of team members

agree on goals in terms of content and deadlines

Project management

derive requirements specification in the team from the project order

and prioritize requirements

Create and maintain project plan

Identify project risks and plan meaningful mitigation measures, e.g. early feasibility studies

Create and maintain project schedule

Rough planning of tasks

Plan process

Planning effort, appointments, rooms

Plan Project Reviews

apply agile process model in conjunction with Timebox model to ensure a minimal project success

Define a minimum goal that can be achieved by the team.

define extended goals for fast teams

Drafting a final project report

Document and evaluate results

Document and evaluate the project process

Lead team

Monitoring and controlling goal achievement

Coordinate collaboration between team members

Recognizing and resolving conflict situations within the team

Project management

Planning and managing project sections

plan tasks for the next phase of the project in detail

and assign them meaningfully to the team members

Plan and moderate content reviews with team

members

Evaluate project results in the team

Modify the project section plan and, if necessary, the project plan according to the project procedure.

evaluate the approach of the current project phase

retrospectively and, if necessary, modify it for the next project phase.

Document project sections

plan access to shared laboratory resources

computer

tools

special workstations and measuring stations

special test environments

Prepare project decisions in the team

Teaching and Learning

■ Seminar

Methods

■ Project

Examination Types with Weights

■ accompanying: project work [ungraded]

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

Contact Hours	23 Hours $\pm$ 2 SWS
----------------------	----------------------

Self-Study	127 Hours
-------------------	-----------

Recommended Prerequisites	basic knowledge of project management basic experience as a member of project teams
Mandatory Prerequisites	▪ Project requires attendance in the amount of: 8 Termine ▪ Participation in final examination only after successful participation in Project
Recommended Literature	▪ Hans-D. Litke, „Projektmanagement, Methoden, Techniken, Verhaltensweisen, Evolutionäres Projektmanagement“, Hanser ▪ Ken Schwaber: Agiles Projektmanagement mit Scrum (Microsoft Press) ▪ Litke, Kunow, Schulz-Wimmer, „Projekt-Management“, Taschenguide , Haufe ▪ Stefan Kreiser, Skripte der Vorlesung Software Engineering f.d. Automatisierungstechnik: „Projektmanagement, Vorgehensmodelle“, ILIAS ▪ Stanley E.Portny, „Projektmanagement für Dummies“, Wiley ▪ Marcus Heidbrink, „Das Projektteam“, Haufe ▪ Video Tutorial für SCRUM: http://www.video2brain.com/de/videotraining/agile-softwareentwicklung-mit-scrum
Use of the Module in Other Study Programs	▪ PLET in Master Elektrotechnik 2020 ▪ PLET in Master Elektrotechnik und Informationstechnik 2024 ▪ PLET in Master Technische Informatik 2020
Specifics and Notes	
Last Update	22.8.2025, 10:17:37

6.28 PLSYP - Projektleitung Systementwicklungs-Projekt

Module ID	PLSYP_MaTIN2024
Module Name	Projektleitung Systementwicklungs-Projekt
Type of Module	Elective Modules
Recognized Course	SYP - System Development Project
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

- Studierende leiten ein Team von Systementwicklungs-Projekt-Teilnehmenden (SYP) an
- indem sie
 - ein interessantes und geeignetes Entwicklungs-Thema ausarbeiten,
 - die Fortschritte des Teams über den gesamten Entwicklungsprozess überwachen,
 - regelmäßige Status-Meetings mit den Teammitgliedern abhalten
 - der Gesamtprojektleitung regelmäßig in weiteren Status-Meetings berichten
- um (später) in einem (beruflichen) Umfeld Verantwortung für die (technische) Leitung eines Entwicklungsprojekts übernehmen zu 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 ▪ accompanying: project work [100%]

Workload 150 Hours

Contact Hours 12 Hours $\pm$ 1 SWS

Self-Study	138 Hours
-------------------	-----------

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

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

Recommended Literature	▪ keine
-------------------------------	---

Use of the Module in Other Study Programs	▪ SYP in Bachelor Technische Informatik 2020 ▪ SYP in Bachelor Informatik und Systems-Engineering 2024 ▪ PLSYP in Master Technische Informatik 2020
--	---

Permanent Links to Organization	<u>Ilu course</u>
--	-----------------------------------

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

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

6.29 QC - Quantum Computing

Module ID	QC_MaTIN2024
Module Name	Quantum Computing
Type of Module	Elective Modules
Recognized Course	QC - Quantum Computing
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every term
Module Coordinator	Prof. Dr. Heiko Knospe/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME ▪ Prof. Dr. Heiko Knospe/Professor Fakultät IME

Learning Outcome(s)

Was: Die Studierenden lernen die Grundlagen des Quantum Computing kennen. Es werden die mathematischen Grundlagen des Quantum Computing und Kenntnisse über Quanten-Schaltkreise und wichtige Quanten-Algorithmen vermittelt (HF 1).

Womit: Der Dozent/die Dozentin vermittelt Wissen und Basisfertigkeiten in der Vorlesung. In der Übung bearbeiten die Studierenden unter Anleitung Aufgaben und entwickeln Quantenalgorithmen (HF 1).

Wozu: Quantencomputer können mit Hilfe von verschränkten Qubits eine große Zahl von Eingangswerten gleichzeitig verarbeiten und bestimmte schwere Probleme lösen (HF 2).

Module Contents

Lecture / Exercises

- Fundamental concepts
- Single Quantum Bits
- Bloch Sphere
- Quantum Key Distribution
- Hilbert spaces
- Bra-Ket notation
- Inner product, Outer product, Tensor product
- Hermitian and unitary operators
- Multiple Qubit Systems
- Entangled states
- Measurement
- Quantum Gates and Circuits
- Realizing unitary transformations as Quantum Circuits
- Deutsch and Deutsch-Josza algorithms
- Discrete Fourier Transform and Quantum Fourier Transform
- Quantum Algorithms: Shor, Grover, HHL (optional)
- Quantum complexity classes
- Quantum Error Correction (optional)

Practical quantum computing and coding using the Qiskit SDK.

Teaching and Learning Methods Lecture / Exercises

Examination Types with Weights

- final: (digital) written exam or oral examination [100%]

Workload 150 Hours

Contact Hours 34 Hours $\pm$ 3 SWS

Self-Study	116 Hours
Recommended Prerequisites	
Mandatory Prerequisites	
Recommended Literature	▪ M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press. ▪ E. Rieffel, W. Polak, Quantum Computing, MIT Press. ▪ B. Zygelman, A First Introduction to Quantum Computing and Information, Springer. ▪ H.Y. Wong, Introduction to Quantum Computing, Springer. ▪ Matthias Homeister, Quantum Computer verstehen, Springer.
Use of the Module in Other Study Programs	QC in Master Communication Systems and Networks 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.30 QEKS - Qualitätsgesteuerter Entwurf komplexer Softwaresysteme

Module ID	QEKS_MaTIN2024
Module Name	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme
Type of Module	Elective Modules
Recognized Course	SEKM - Software Engineering by Components and Pattern
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Stefan Kreiser/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Stefan Kreiser/Professor Fakultät IME

Learning Outcome(s)

Studierende sind im Hinblick auf die Qualität eines Softwaresystems in der Lage:

- zur vorhersagbaren, effizienten Entwicklung eines Softwaresystems bzw. einer Softwarearchitektur zielgerichtet angemessene Wiederverwendungsstrategien und professionelle Modellierungs- und Entwicklungswerkzeuge sowie den Rahmenbedingungen insgesamt angemessene Projektstrukturen einzusetzen.
 - die Softwarearchitektur für komplexe, verteilte Automatisierungssysteme unter Berücksichtigung der spezifischen Anforderungen hinsichtlich der besonderen Zielsetzung des jeweiligen Automatisierungssystems zu analysieren, zu konzipieren, zu entwerfen, zu implementieren, zu prüfen und zu bewerten.
 - die besonderen Anforderungen an die Servicequalität, an die Einsatzumgebung und die organisatorischen Rahmenbedingungen für die Entwicklung, die sich aus dem Entwicklungsprozess und einem angemessenen Lebenszyklusmanagement ergeben, zu erkennen und im Hinblick auf ihre Relevanz für die Softwarearchitektur des Automatisierungssystems zu analysieren und zu bewerten.
-

Module Contents**Lecture / Exercises**

Terminology
value vs. cost of a technical software
distributed software system, concurrency
software quality, quality of service, refactoring
complexity (algorithmic, structural), emergence
re-use, symmetry and symmetry operations, abstraction, invariants
quality controlled re-use, methodical approaches
variants of white box re-use
black box re-use
grey box re-use (hierarchical approach to re-use)
re-use in automation control software systems
determinism
benefits and challenges
tailoring process models and personnel structures in projects
meet requirements in development projects predictably (product quality, cost, deadlines)
distributed development, maintenance and support of software systems
software pattern
pattern description using UML
essential architectural pattern
construction pattern
structural pattern
behavioural pattern
class based (static) vs. object based (dynamic) pattern
essential pattern for concurrent and networked real time systems
encapsulation and role based extension of layered architectures
concurrency structures to optimize throughput and system response latency
distributed event processing
process synchronisation
construction and use of pattern catalogues, pattern languages
pattern based design of complex software systems
components and frameworks
design principles
interface architecture
active and passive system elements
design, programming and test
quality
configuration and use
using middleware systems to develop architectures of technical software systems
ORB architectures, e.g. CORBA and TAO
integrated system platforms, e.g. MS .NET
multi agent systems (MAS)
agent architectural models
collaboration between agents
agent languages
considering cases for MAS application

use pattern to design complex software systems
extract and discuss purpose, limitation of use, invariant and configurable parts of pattern from english and german literature sources
understand implementation skeletons of pattern and map them to problem settings with limited technical focus
discuss benefits of using object oriented programming languages
derive recurrent settings in the development of complex software systems
implement pattern on exemplary settings and test resulting implementations
reasonably combine pattern to solve recurring problem settings with a broader technical focus
use UML2 notations
use professional UML2 IDE for round-trip-engineering
integrate software system based on exemplary implementations of the pattern to combine
conduct integration test, assess software quality and optimize software system

construct black-box-components based on pattern
analyse component based software architectures
derive suitable scope from architectural specs
understand and discuss development process to construct software systems
find active and passive system elements and derive system run time behaviour
understand abstract system interfaces to interconnect, configure and activate components
understand abstract system interfaces to exchange applicational run time data
understand system extension points (functional and structural system configuration layer)
analyse distribution architectures
understand basic system services (describe and reason service usage, relate to system tasks)
relate pattern to structure making architectural software artefacts
derive suitable range of applications for a given distribution architecture
understand engineering process to construct user applications (application layer)
discuss attributes and limitation of usage of interconnection protocols
find designated system extension points
compare MAS to conventional distribution architectures
agent vs. component
architectural models
activation of agents
deployment of agents
protocols for interconnection and collaboration
range of applications and and limitation of usage

Seminar

challenging seminar topics can be defined e.g. from the following or related subject areas

- reusable artifacts for building the architecture of distributed software systems,
- professional distribution architectures,
- Multiagent systems,
- special economic, liability and ethical requirements for software systems with (distributed) artificial intelligence and their effects on the design of software architectures

present personal work results and work results of the team in a compact and target-group-oriented way, both orally and in writing

Project

Develop software artifact of a distribution architecture for complex software systems
Carry out project planning in distributed teams with an agile process model
Perform extensive system analysis with respect to the role of the artifact in the distribution architecture
Determine design input requirements for the development of the artifact
Specify and model the software artifact based on the design input requirements
Select and justify design principles and patterns to achieve defined quality objectives
Derive interfaces, behavioral and structural models iteratively based on patterns in UML2 notations
Use professional UML2 design tool purposefully
Verify and evaluate models, correct model errors and optimize models
Programming software artifacts in C++
define meaningful test scenarios and verify software artifacts
Evaluate the quality of the software artifact
Present the team's project results to a professional audience in a compact and target-group-oriented way

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

Examination Types with	▪ accompanying: project work [ungraded] and
Weights	▪ final: oral examination [100%]

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

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

Self-Study	93 Hours
Recommended Prerequisites	▪ Modul PLET: oder aus einem (naturwissenschaftlich-technischen) Bachelorstudium: - grundlegende Kenntnisse in (agilem) Projektmanagement ▪ - programming skills in an object-oriented programming language, preferably C++ ▪ - knowledge of software modeling using Unified Modeling Language (UML) or other (formal) languages that support modeling of interfaces, behavior and structures ▪ - basic knowledge in (agile) project management, SCRUM oder XP ▪ - basic knowledge of essential software architectural models ▪ - basic knowledge of interconnection models in software systems (OSI, TCP/IP, Messaging)
Mandatory Prerequisites	▪ Project requires attendance in the amount of: 3 Termine ▪ Participation in final examination only after successful participation in Project
Recommended Literature	▪ D. Schmidt et.al.: Pattern-Oriented Software Architecture. Patterns for Concurrent and Networked Objects (Wiley) ▪ Gamma et.al.: Design Patterns, (Addison-Wesley) ▪ Martin Fowler: Refactoring, Engl. ed. (Addison-Wesley Professional) ▪ U. Hammerschall: Verteilte Systeme und Anwendungen (Pearson Studium) ▪ Andreas Andresen: Komponentenbasierte Softwareentwicklung m. MDA, UML2, XML (Hanser Verlag) ▪ T. Ritter et. al.: CORBA Komponenten. Effektives Software-Design u. Progr. (Springer) ▪ Bernd Oestereich: Analyse und Design mit UML 2.5 (Oldenbourg) ▪ OMG Unified Modeling Language Spec., www.omg.org/um ▪ I. Sommerville: Software Engineering (Addison-Wesley / Pearson Studium) ▪ K. Beck: eXtreme Programming (Addison-Wesley Professional) ▪ Ken Schwaber: Agiles Projektmanagement mit Scrum (Microsoft Press)
Use of the Module in Other Study Programs	▪ QEKS in Master Elektrotechnik 2020 ▪ QEKS in Master Elektrotechnik und Informationstechnik 2024 ▪ QEKS in Master Technische Informatik 2020
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.31 RA - Reflexion Auslandssemester

Module ID	RA_MaTIN2024
Module Name	Reflexion Auslandssemester
Type of Module	Elective Modules
Recognized Course	RA - Reflection on the semester abroad
ECTS credits	6
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers
Learning Outcome(s)	Die Studierenden reflektieren kulturelle, gesellschaftliche und strukturelle Gemeinsamkeiten und Unterschiede ihrer Heimathochschule/-land und der Gasthochschule/-land. Sie werden dadurch in die Lage versetzt, bewusste Entscheidungen hinsichtlich ihrer zukünftigen akademischen und beruflichen Mobilität zu treffen. Die Studierenden reflektieren die persönlichen Erfahrungen, die sie während ihres Auslandssemesters gemacht haben, um ihr allgemeines Wertebewusstsein kritisch zu hinterfragen und ggf. zu justieren.
Module Contents	
Seminar	
	Students are able to reflect on cultural, social and structural similarities and differences between their home university/country and the host university/country. This enables them to make informed decisions regarding their future academic and professional mobility. Students can reflect on the personal experiences they have had during their semester abroad in order to critically question and, if necessary, adjust their general awareness of values.
Teaching and Learning Methods	Seminar
Examination Types with Weights	▪ accompanying: oral contribution or term paper or learning portfolio [ungraded]
Workload	180 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	168 Hours
Recommended Prerequisites	As a rule, a one-semester or longer period of study at a foreign university is a prerequisite for participation.
Mandatory Prerequisites	Seminar requires attendance in the amount of: 1 Termin
Recommended Literature	
Use of the Module in Other Study Programs	▪ RA in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ RA in Bachelor Medientechnologie 2024 ▪ RA in Bachelor Informatik und Systems-Engineering 2024 ▪ RA in Master Communication Systems and Networks 2024 ▪ RA in Master Elektrotechnik und Informationstechnik 2024 ▪ RA in Master Medientechnologie 2024
Specifics and Notes	This course is aimed exclusively at students who have completed a semester abroad.

Last Update

19.7.2025, 14:32:16

6.32 RFSD - RF System Design

Module ID	RFSD_MaTIN2024
Module Name	RF System Design
Type of Module	Elective Modules
Recognized Course	RFSD - RF System Design
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

In general: Students will learn how high frequency components of wireless communication systems work

Module-specific:

students will get a general introduction in rf systems

they will learn in detail how transmitters and receivers in wireless communication systems work

they will learn in detail how the components of such systems (LNA, mixer, amplifier, oscillator, etc.) work

they will learn about limitation effects and noise in such systems

they will learn how to adapt the components to each other and how to plan and design the complete system (transmitter and / or receiver)

Module Contents

Lecture / Exercises

RF System, Applications

Noise in RF systems

noise classification and characterization

noise calculation

noise figure

noise matching

Linear and nonlinear circuit behaviour

theory

nonlinearities with mixers

nonlinearities with amplifiers

RF system components

receiver components

transmitter components

frequency generation

Lab

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

Examination Types with Weights	▪ accompanying: exercise lab [ungraded] and ▪ final: (digital) written exam or oral examination [100%]
---------------------------------------	---

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

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

Self-Study	105 Hours
Recommended Prerequisites	No formal requirements, but students should have knowledge in High Frequency and Microwave Topics
Mandatory Prerequisites	▪ Participation in final examination only after successful participation in Lecture / Exercises ▪ Lab requires attendance in the amount of: 3 Labortermine und 1 Präsentationstermin ▪ Participation in final examination only after successful participation in Lab
Recommended Literature	▪ Kraus & Carver Eletromagnetics, McGraw Hill, 2006. ▪ Michale Steer, Microwave and RF Design
Use of the Module in Other Study Programs	▪ RFSD in Master Communication Systems and Networks 2020 ▪ RFSD in Master Communication Systems and Networks 2024 ▪ RFSD in Master Elektrotechnik und Informationstechnik 2024 ▪ RFSD in Master Medientechnologie 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.33 THI - Theoretische Informatik

Module ID	THI_MaTIN2024
Module Name	Theoretische Informatik
Type of Module	Mandatory Module
Recognized Course	THI - Theoretical Computer Science
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Hubert Randerath/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hubert Randerath/Professor Fakultät IME

Learning Outcome(s)

(WAS) Die Studierenden erlernen formale Grundlagen der Informatik

(WOMIT) indem Sie

- den Umgang mit Typ2, Typ1 und Typ0-Sprachen erlernen und formale Maschinen konstruieren, die Sprachen des jeweiligen Typs erkennen,
- mit formalen Modellen der Informatik arbeiten,
- Kenntnisse der Berechenbarkeits, Entscheidbarkeits- und Komplexitätstheorie auf praktische Probleme anwenden,
- einen präzisen Algorithmenbegriff verwenden, um die Tragweite von Algorithmen zu beschreiben und die Komplexität von Algorithmen zu bestimmen,
- die prinzipielle Lösbarkeit algorithmischer Probleme untersuchen,

(WOZU) um in Forschungsergebnisse in späteren Lehrveranstaltungen und Abschlussarbeiten auf ein solides theoretisches Fundament stellen zu können.

Module Contents

Lecture / Exercises

An algorithm's complexity can be determined by analyzing its input and the algorithmic core, e.g., by means of the O-notation. The analysis might consist of a polynomial reduction of a known hard problem like the satisfiability problem in propositional logic of the unknown problem.

Teaching and Learning Methods Lecture / Exercises

Examination Types with Weights ■ final: oral examination [100%]

Workload 150 Hours

Contact Hours 34 Hours $\pm$ 3 SWS

Self-Study 116 Hours

Recommended Prerequisites Basics in automata theory and formal languages

Mandatory Prerequisites

Recommended Literature

- Theoretische Grundlagen der Informatik, Rolf Socher, Hanser Verlag
- Theoretische Informatik, Juraj Hromkovic, Teubner-Verlag
- Grundkurs Theoretische Informatik, Gottfried Vossen und Kurt-Ulrich Witt, Vieweg-Verlag
- Theoretische Informatik - kurzgefasst, Uwe Schöning, Spektrum Akademischer Verlag

Use of the Module in Other Study Programs	THI in Master Technische Informatik 2020
--	--

Specifics and Notes

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

6.34 VAE - Virtual Acoustic Environments

Module ID	VAE_MaTIN2024
Module Name	Virtual Acoustic Environments
Type of Module	Elective Modules
Recognized Course	VAE - Virtual Acoustic Environments (VAE)
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	1-2
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)

What: The students learn the basic concepts, the technology and perception-related aspects of virtual acoustic environments. The course will be strongly related to research aspects and projects

How: The students apply their knowledge on Signal Processing, Audio, and in the field of VR on different aspects of Virtual Acoustic Environments. Actual trends in research and state of the art applications will be integrated, tested, analyzed and evaluated.

Aim: The students shall be able to work on research topics which consider topics which are scientifically new and relevant. Aspects of scalability and commercialization play a role

Module Contents

Lecture

The basic concepts of headphone-based or loudspeaker-based VR-systems are introduced

Project

In one specific topic of Virtual Acoustic Environments the students shall get deep know-how on the technology and apply it to a practical problem and present the results

Lab

Teaching and Learning Methods	▪ Lecture ▪ Project ▪ Lab
Examination Types with Weights	▪ accompanying: project work [100%] and ▪ accompanying: exercise lab [ungraded]
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	Know-how on Acoustics and audio signal processing
Mandatory Prerequisites	

Recommended Literature	▪ Rozinska, A. "Immersive Sound" ▪ Blauert, J. "Spatial Hearing" ▪ Zotter, F., Frank, M. "Ambisonics: A Practical 3D Audio Theory for Recording, Studio Production, Sound Reinforcement, and Virtual Reality"
-------------------------------	---

Use of the Module in Other Study Programs	▪ VAE in Master Communication Systems and Networks 2020 ▪ VAE in Master Communication Systems and Networks 2024 ▪ VAE in Master Medientechnologie 2020 ▪ VAE in Master Medientechnologie 2024 ▪ VAE in Master Technische Informatik 2020
--	--

Specifics and Notes

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

6.35 VER - Virtuelle und erweiterte Realität

Module ID	VER_MaTIN2024
Module Name	Virtuelle und erweiterte Realität
Type of Module	Elective Modules
Recognized Course	VER - Virtual and Augmented Reality
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every winter term
Module Coordinator	Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME ▪ Prof. Dr. Stefan Grünvogel/Professor Fakultät IME

Learning Outcome(s)

WAS:

Das Modul vermittelt folgende Kenntnisse und Fertigkeiten:

- Virtual- und Augmented-Reality-Anwendungen konzipieren, aufbauen und bewerten
- Interaktions und Navigationsverfahren erstellen
- Basistechnologien der virtuellen und erweiterten Reality weiterentwickeln
- Werkzeuge und Methoden zur Entwicklung von VR/AR-Anwendungen verwenden
- Algorithmische und mathematische Grundlagen von VR/AR anwenden

WOMIT:

Die Kompetenzen werden zunächst über eine Vorlesung durch die Dozenten vermittelt und danach im Praktikum anhand konkreter Aufgabenstellung von den Studierenden vertieft. Im seminaristischen Teil der Lehrveranstaltung recherchieren die Studierenden zu vorgegebenen Themen anhand von Fachartikeln und weiteren Informationsquellen über neue Konzepte der virtuellen und erweiterten Realität und stelle diese dar in einer Präsentation dar.

WOZU:

Die sichere Anwendung der Grundlagen der virtuellen und erweiterten Realität ist Voraussetzung für die Entwicklung komplexer interaktiver medientechnischer Systeme (HF1). Weiterhin erlaubt das Grundlagenwissen die Bewertung bestehender Systeme und das wissenschaftliche Arbeiten in diesem Gebiet (HF2).

Module Contents**Lecture**

- Explain terms from the field of virtual and augmented reality
- Explain and compare data structures and algorithms for VR/AR applications
- 3D data formats
- Spatial data structures
- Describing Multimodal User Interfaces
- Selection of 3D objects
- Manipulation of 3D objects
- Navigation in virtual scenes
- system control
- Describe input and output devices and specific virtual and augmented reality hardware
- display technologies
- Stereo Displays
- Autostereoscopic Displays
- projection solutions
- Wearable Displays
- Head Mounted Displays
- Handheld Displays
- See-through Displays
- Workbench
- Cave
- Tiled Displays
- 3D-Audio
- Force Feedback Devices
- Haptic feedback
- input devices
- controller
- data gloves
- locomotion devices
- Explain algorithmic and mathematical basics
- stereoscopy
- tracking
- capture of position and orientation: Degrees of freedom
- tracking technologies
- Mechanical
- Optical
- Electromagnetic
- ultrasound
- inertial
- eye tracking
- head tracking
- object tracking
- Markerless Tracking
- Marker-Based Tracking
- rendering
- management of large 3D scenes
- haptic rendering
- stereo rendering
- real-time rendering
- collision detection
- intersections between primitives
- Discrete and continuous collision detection
- acceleration data structures
- collision response

Lab

- Design, build and evaluate virtual environments and augmented reality applications
 - Creating Interaction and Navigation Procedures
 - Further develop fundamental technologies of virtual and augmented reality
 - Use tools and methods to implement VR/AR applications
 - Apply algorithmic and mathematical principles of VR/AR
 - understand and grasp textual tasks
 - Testing and debugging your own application
-

Seminar

Apply Algorithmic and Mathematical fundamentals
Check interaction and navigation procedures
Independently obtaining and summarizing scientific literature
Present and discuss new concepts of virtual and augmented reality

Teaching and Learning Methods

- Lecture
- Lab
- Seminar

Examination Types with Weights

- accompanying: exercise lab [ungraded] and
- final: oral examination or term paper or project work [100%]

Workload 150 Hours

Contact Hours 45 Hours $\triangleq$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites
Computer Graphics
Computer Animation

Mandatory Prerequisites
Lab requires attendance in the amount of: 2 Termine

Recommended Literature

- R. Dörner et al., Virtual und Augmented Reality (VR/AR): Grundlagen und Methoden der Virtuellen und Augmentierten Realität, Springer Vieweg, 2019
- Schmalstieg und Höllerer, Augmented Reality – Principles and Practice, Addison Wesley, 2016
- T. Akenine-Möller, et al., Real-Time Rendering Fourth Edition, Taylor & Francis Ltd., 2018
- J. Jerald, The VR Book: Human-Centered Design for Virtual Reality, Acm Books, 2015

Use of the Module in Other Study Programs

- VER in Master Medientechnologie 2020
- VER in Master Medientechnologie 2024
- VER in Master Technische Informatik 2020

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.36 XGA - Gremienarbeit

Module ID	XGA_MaTIN2024
Module Name	Gremienarbeit
Type of Module	Elective Modules
Recognized Course	XGA - Participation in appointment committees
ECTS credits	0
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	1-2
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	
Learning Outcome(s)	Anerkennbar ist die Mitarbeit in Berufungskommissionen als studentisches Mitglied. Die Anzahl der anerkannten ECTS-Punkte richtet sich nach der Anzahl der nachgewiesenen Stunden in der Gremientätigkeit. Es wird 1ECTS-Punkt pro 25 Stunden Gremienarbeit angerechnet. Der/die Vorsitzende der Berufungskommission vergibt die ECTS und bescheinigt diese. Es wird erwartet, dass der/die Studierende sich aktiv in die Arbeit einbringt.
Module Contents	
Project	
Teaching and Learning Methods	Project
Examination Types with Weights	▪ final: technical discussion [ungraded]
Workload	0 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	-12 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 ▪ XIB in Bachelor Technische Informatik 2020 ▪ XIB2 in Bachelor Technische Informatik 2020 ▪ XGA in Bachelor 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

7. Electives Catalogs

The following shows which modules can be selected in a particular elective area. The following notes and regulations apply to all elective areas:

- **When choosing modules from elective catalogs, the conditions formulated in Specializations also apply.**
- The semester in which elective modules of an elective catalog can typically be taken can be found in the study plans.
- As a rule, modules are only offered in either the summer or winter semester. This means that any required accompanying examination can only be taken in this semester. The summative examinations for modules in Faculty 07 are usually offered in the examination period after each semester.
- A completed module is recognized for a maximum of one elective area, even if it is listed in several elective areas.
- There is an admission restriction for some modules. More information on this can be found in the announcements on admission restrictions.
- Registration and admission to non-faculty modules are subject to deadlines and other conditions set by the faculty or university offering the module. Their admission cannot be guaranteed. Students must contact the relevant external lecturer in good time to find out whether they are allowed to take part in an external module and what they need to do to register and participate.
- Upon application, a suitable modules can be added to the elective area. Such an application must be submitted informally to the head of degree program at least four months before the planned participation in that module. The examination board decides on the acceptance of the application in consultation with the head of degree program and suitable teaching staff. A study achievement to be recognized
 - must fit in with the intended graduate profile of the degree program and contribute to its achievement,
 - must be oriented towards learning outcomes and must not serve solely to impart knowledge,
 - must correspond to the qualification level of a Master's degree program,
 - must represent a meaningful increase in competence against the background of the intended course of study,
 - must have been completed by an examination and
 - must not be identical in terms of content and learning outcomes to coursework that has already been completed.
- Modules are not listed below,
 - which in the past were only recognized for an elective catalog as part of individual recognition procedures or
 - which in the past were only recognized for an elective catalog as part of a stay abroad and the associated individual learning agreement.

Stays abroad

- Students who have integrated a stay abroad into their studies and have completed coursework at a foreign university can have this recognized upon application and with the approval of the examination board.
- A Learning Agreement must be concluded with the Faculty's Recognition Officer before the start of the stay abroad. In particular, it is agreed for which mandatory modules or elective catalogs the coursework completed abroad will be recognized.

7.1 VMT - Vertiefung Mathematik

Für dieses Wahlmodul stehen folgende Module zur Verfügung.

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

Modules of the faculty:

Module ID	Module Name	ECTS
HIM	Advanced Mathematics	5
KOGA	Kombinatorische Optimierung und Graphenalgorithmen	5
KRY	Cryptography	5
MLWR	Maschinelles Lernen und wissenschaftliches Rechnen	5

Modules of other faculties or universities:

Affiliation	Module Name	ECTS
TH Köln (Fakultät 10)	Multivariate Statistik	6

7.2 VTI - Vertiefung Technische Informatik

You must select modules of 15 ECTS credit points in total out of this catalog.

Modules of the faculty:

Module ID	Module Name	ECTS
AMC	Advanced Multimedia Communications	5
AMS	Special Aspects of Mobile Autonomous Systems	5
ARP	Alternative Rechnerarchitekturen und Programmiersprachen	5
CI	Computational Intelligence	5
ESD	Embedded Systems Design	5
IIS	Intelligent Information Systems	5
LCSS	Large and Cloud-based Software-Systems	5
NGN	Next Generation Networks	5
QC	Quantum Computing	5
QEKs	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme	5

Modules of other faculties or universities:

Affiliation	Module Name	ECTS
TH Köln (Fakultät 10)	Advanced Machine Learning	6
Universität Köln	Quantum Information Theory	6

7.3 W - Allgemeiner Wahlbereich

You must select modules of 20 ECTS credit points in total out of this catalog.

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

- Electives Catalog VTI - Vertiefung Technische Informatik
- Electives Catalog VMT - Vertiefung Mathematik

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
ACC	Advanced Channel Coding	5
AMC	Advanced Multimedia Communications	5
AMS	Special Aspects of Mobile Autonomous Systems	5
ARP	Alternative Rechnerarchitekturen und Programmiersprachen	5
AVT	Audio- und Videotechnologien	5
AVV	Algorithmen der Videosignalverarbeitung	5
CI	Computational Intelligence	5
CSO	Computersimulation in der Optik	5
DLO	Deep Learning und Objekterkennung	5
DMC	Digital Motion Control	5
DSP	Digital Signal Processing	5
ESD	Embedded Systems Design	5
HIM	Advanced Mathematics	5
IBD	InnoBioDiv	5
IIS	Intelligent Information Systems	5
ITF	IT-Forensik	5
KOGA	Kombinatorische Optimierung und Graphenalgorithmen	5
KRY	Cryptography	5
LCSS	Large and Cloud-based Software-Systems	5
MCI	Mensch-Computer-Interaktion	5
MLWR	Maschinelles Lernen und wissenschaftliches Rechnen	5
NGN	Next Generation Networks	5
PAP	Parallele Programmierung	5
QC	Quantum Computing	5
QEKs	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme	5
QM	Quantenmechanik	5
RFSD	RF System Design	5
SIM	Simulation in der Ingenieurwissenschaft	5
VAE	Virtual Acoustic Environments	5
VER	Virtuelle und erweiterte Realität	5

Modules of other faculties or universities:

Affiliation	Module Name	ECTS
TH Köln (Fakultät 10)	Advanced Machine Learning	6
TH Köln (Fakultät 10)	Data Driven Modeling	6
TH Köln (Fakultät 10)	Data Science and Ethics	6
TH Köln (Fakultät 10)	Domain-Driven Design of Large Software Systems	6
TH Köln (Fakultät 09)	Modellierung von Energiesystemen	5
TH Köln (Fakultät 10)	Multivariate Statistik	6
Universität Köln	Quantum Information Theory	6
Hochschule Bonn-Rhein-Sieg	Virtuelle Private Netze	6

7.4 X - Fachübergreifende Kompetenzen und Soft Skills

In diesem Wahlbereich können Module zu außerfachlichen, nicht-technischen Themen belegt werden. Im Folgenden werden nur Module dargestellt, die regelmäßig angeboten werden. Es sind aber auch einmalig oder unregelmäßig angebotene Module in diesem Wahlbereich wählbar, beispielsweise Module, die von der Kompetenzwerkstatt angeboten werden. Die Anerkennung eines solchen, unten nicht aufgeführten Moduls für diesen Wahlbereich muss per E-Mail an die Studiengangleitung vor der Teilnahme geklärt werden.

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

Modules of the faculty:

Module ID	Module Name	ECTS
ERMK	Entrepreneurship, Gewerblicher Rechtsschutz, Market Knowledge	5
ETH	Ethik	5
PHTM	Philosophische Handlungstheorie Master	5
PLET	Projektleitung	5
PLSYP	Projektleitung Systementwicklungs-Projekt	5
XGA	Gremienarbeit	0

Modules of other faculties or universities:

Affiliation	Module Name	ECTS
TH Köln (Kompetenzwerkstatt)	Als Führungskraft begeistern - Die Basics im Leadership	2
TH Köln (Fakultät 10)	Data Science and Ethics	6
TH Köln (Kompetenzwerkstatt)	Design Thinking	3
TH Köln (Kompetenzwerkstatt)	Digitale Arbeitswelt	5
TH Köln (Kompetenzwerkstatt)	Entrepreneurship - Grundlagenveranstaltung	6
TH Köln (Kompetenzwerkstatt)	Entwicklung von Geschäftsszenarios bei Existenzgründung	6
TH Köln (Kompetenzwerkstatt)	Intellectual Activism - feministische Verbindungen zwischen Forschung und Praxis	3
TH Köln (Kompetenzwerkstatt)	Interkulturelle Teamarbeit - Vielfalt als Erfolgsfaktor	3
TH Köln (Kompetenzwerkstatt)	Kommunikative Kompetenz in Führungssituationen	2
TH Köln (Kompetenzwerkstatt)	Konflikte verstehen und effektiv lösen	2
TH Köln (Kompetenzwerkstatt)	Konfliktlösungs- und Verhandlungstechniken	6
TH Köln (Fakultät 10)	Leadership Principles and Strategic Management	6
TH Köln (Kompetenzwerkstatt)	Persönliche Karriereentscheidungen mit dem Entscheidungsnavi	1
TH Köln (Kompetenzwerkstatt)	Praxisprojekt: Komplexe Herausforderungen der digitalen Arbeitswelt	6
TH Köln (Kompetenzwerkstatt)	Produktentwicklungsmethoden	6
TH Köln (Fakultät 10)	Projektmanagement (deutsch)	6
TH Köln (Fakultät 10)	Projektmanagement (englisch)	6
TH Köln (Kompetenzwerkstatt)	Rhetorik in der Gesprächsführung	3
TH Köln (Kompetenzwerkstatt)	Rhetorik in der Verhandlungstechnik	3
TH Köln (Fakultät 04)	Risk Management 1	5
TH Köln (Kompetenzwerkstatt)	Selbstlernmodul Moderation	2
TH Köln (Kompetenzwerkstatt)	Stärken stärken und Schwächen nutzen	1
TH Köln (Fakultät 10)	StartUp Bootcamp	5
TH Köln (Fakultät 04)	Steuern	6

Affiliation	Module Name	ECTS
TH Köln (Kompetenzwerkstatt)	Unternehmensführung im öffentlichen Sektor	6

8. Specializations

There are no specializations in the study program.

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 - Entwicklung und Design	HF2 - Forschung und Innovation	HF3 - Leitung und Management	HF4 - Qualitätssicherung und Te...	K.1 - Entwicklung und Konzeptio...	K.2 - Prüfung und Bewertung kom...	K.3 - Wissenschaftliches Arbeit...	K.4 - Projektmanagement und Tea...	K.5 - Selbstorganisation und au...	K.6 - Kommunikation und interku...	K.7 - Technische und naturwisse...	K.8 - Nachhaltigkeit und gesell...	K.9 - Analyse, Simulation und A...	K.10 - Führungs- und Entscheidun...	K.11 - Anwendung ethischer Werte...	K.12 - Integratives Denken und H...	K.13 - Innovation und Kreativität...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
ACC	Advanced Channel Coding	●	●		●	●	●	●	●	●				●	●		●	●	●	●		
AMC	Advanced Multimedia Communications	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●		●		
AMS	Special Aspects of Mobile Autonomous Systems	●	●			●		●		●		●		●							●	●
ARP	Alternative Rechnerarchitekturen und Programmiersprachen	●	●		●	●	●	●	●	●	●	●		●	●		●	●				
AVT	Audio- und Videotechnologien	●	●	●	●	●	●	●	●	●	●	●		●	●		●	●				
CI	Computational Intelligence	●	●	●	●	●	●	●		●	●	●		●			●	●				
DLO	Deep Learning und Objekterkennung	●	●		●	●	●	●	●		●			●		●						
DMC	Digital Motion Control	●	●		●	●	●			●	●	●	●	●		●	●					
DSP	Digital Signal Processing	●	●		●	●	●	●		●				●			●			●		
ESD	Embedded Systems Design	●				●	●		●	●	●	●		●				●		●	●	●
ETH	Ethik					●	●	●		●	●		●			●	●	●	●		●	
FP	Forschungsprojekt	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●
HIM	Advanced Mathematics				●		●	●						●						●		
IBD	InnoBioDiv					●	●	●		●	●	●						●	●	●	●	●
IIS	Intelligent Information Systems	●	●	●	●	●	●	●		●	●	●		●			●	●				
ITF	IT-Forensik	●			●	●	●	●	●	●				●	●		●	●				

Abbr.	Module Name	HF1 - Entwicklung und Design	HF2 - Forschung und Innovation	HF3 - Leitung und Management	HF4 - Qualitätssicherung und Te...	K.1 - Entwicklung und Konzeptio...	K.2 - Prüfung und Bewertung kom...	K.3 - Wissenschaftliches Arbeit...	K.4 - Projektmanagement und Tea...	K.5 - Selbstorganisation und au...	K.6 - Kommunikation und interku...	K.7 - Technische und naturwisse...	K.8 - Nachhaltigkeit und gesell...	K.9 - Analyse, Simulation und A...	K.10 - Führungs- und Entscheidun...	K.11 - Anwendung ethischer Werte...	K.12 - Integratives Denken und H...	K.13 - Innovation und Kreativität...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
KOGA	Kombinatorische Optimierung und Graphenalgorithmen	●	●		●	●	●	●		●	●	●					●	●				
KOLL	Kolloquium zur Masterarbeit	●	●	●	●					●	●											
KRY	Cryptography	●	●		●	●	●			●				●						●		
LCSS	Large and Cloud-based Software-Systems	●	●	●	●	●	●	●	●	●	●	●		●			●	●		●		●
MAA	Masterarbeit	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●			●	●
MCI	Mensch-Computer-Interaktion	●	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	
MLWR	Maschinelles Lernen und wissenschaftliches Rechnen							●		●			●	●		●	●	●			●	●
NGN	Next Generation Networks	●	●		●	●	●	●	●	●	●			●				●		●		
PAP	Parallele Programmierung	●	●		●	●	●	●		●	●	●		●				●				
PHTM	Philosophische Handlungstheorie Master																		●		●	
PLET	Projektleitung	●	●	●	●			●	●		●		●		●	●	●		●		●	
PLSYP	Projektleitung Systementwicklungs-Projekt			●			●		●						●		●		●			
QC	Quantum Computing	●	●																	●		
QEKs	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
RA	Reflexion Auslandssemester																	●	●	●		
RFSD	RF System Design		●		●	●	●	●			●			●				●		●	●	
THI	Theoretische Informatik	●	●		●	●	●			●	●	●					●	●				

Abbr.	Module Name	HF1 - Entwicklung und Design	HF2 - Forschung und Innovation	HF3 - Leitung und Management	HF4 - Qualitätssicherung und Te...	K.1 - Entwicklung und Konzeptio...	K.2 - Prüfung und Bewertung kom...	K.3 - Wissenschaftliches Arbeit...	K.4 - Projektmanagement und Tea...	K.5 - Selbstorganisation und au...	K.6 - Kommunikation und Interku...	K.7 - Technische und naturwisse...	K.8 - Nachhaltigkeit und gesell...	K.9 - Analyse, Simulation und A...	K.10 - Führungs- und Entscheidun...	K.11 - Anwendung ethischer Werte...	K.12 - Integratives Denken und H...	K.13 - Innovation und Kreativität...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
VAE	Virtual Acoustic Environments	●	●		●	●	●	●	●						●		●				●	●
VER	Virtuelle und erweiterte Realität	●	●		●	●	●	●	●	●	●	●	●	●		●	●	●				
XGA	Gremienarbeit																					

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
1.1	2025-06-24-18-55-09	1. Reakkreditierte Version	Link
1.0	2024-12-06-08-45-55	1. Begutachtete Version für Reakkreditierung 2024 2. Neues Layout für sämtliche Modulhandbücher	Link

Impressum

Datenschutzhinweis

Haftungshinweis

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