

**Technology**  
**Arts Sciences**  
**TH Köln**

Fakultät 07 für Informations-, Medien- und Elektrotechnik

## **Master Informatik und Systems-Engineering 2024**

### **Modulhandbuch**

Version: 1.1.2025-06-07-10-55-08

Die neueste Version dieses Modulhandbuchs ist verfügbar unter:

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

# 1. Studiengangsbeschreibung

Die Digitalisierung betrifft sämtliche Lebensbereiche: Vom Spielzeug, über das Smartphone bis zum Auto oder Eigenheim steckt Hard- und Software. Alltägliche Abläufe wie der Einkauf oder die reine Kommunikation sind digital durchdrungen. Ein tiefes Verständnis der grundlegenden Technik in ganzer Breite und Tiefe einschließlich theoretischer Modelle, Kommunikations- und Interaktions-Möglichkeiten sowie Hard- und Software-Architekturen als auch intelligenter, selbstlernender Systeme wird daher immer wichtiger.

## Ausrichtung des Studiengangs

Der anwendungsorientierte Masterstudiengang Informatik und Systems-Engineering ist auf eine Regelstudienzeit von 3 Semestern ausgelegt. Der Studiengang baut konsekutiv auf dem Bachelorstudiengang Informatik und Systems-Engineering (vormals Technische Informatik) auf. Er ist aber auch als Zweitstudium zu einem Bachelorstudiengang der Informatik, Elektrotechnik oder Medientechnik geeignet. Die Studierenden erhalten ein tiefgehendes Wissen und Verständnis von fortgeschrittenen Konzepten, Methoden und Technologien der Informatik und der IT-nahen Kommunikationstechnik. Sie dringen dabei bis zum aktuellen Stand der Wissenschaft vor. Ein hoher Praktikums- und Projektanteil stellt sicher, dass die Studierenden das in den einzelnen Fächern Gelernte in Beziehung zueinander bringen und zur Lösung anspruchsvoller praktischer und theoretischer Probleme einsetzen können.

## Berufsfelder

Die Berufsaussichten und Zukunftschancen für Absolventen und Absolventinnen der Technischen Informatik sind sehr gut. Insofern legen Studierende mit einem guten Abschluss eine aussichtsreiche Basis für ihre Karriere. Absolventen und Absolventinnen des Studiengangs sind besonders für Planungs-, Entwicklungs- und Leitungsaufgaben prädestiniert. Ihnen erschließen sich vielfältige Tätigkeitsfelder sowohl in Firmen der Informations- und Kommunikationstechnik, in angrenzenden Gebieten wie der Automobilindustrie und der Automatisierungstechnik als auch bei Dienstleistern wie Banken und Versicherungen. Absolventen und Absolventinnen können mit dem erworbenen akademischen Grad »Master of Science« ebenso ein weiterführendes Promotionsstudium aufnehmen oder eine Stelle im höheren Dienst öffentlicher Institutionen besetzen. Studienverlauf In dem 3-semesterigen Studiengang werden zunächst vertiefte Fachkenntnisse in der Theoretischen Informatik, der Technischen Informatik, der Mathematik sowie fachübergreifende Kompetenzen und Soft-Skills vermittelt. Die fachliche Spezialisierung erfolgt durch die individuelle Zusammenstellung von neun Wahlmodulen aus verschiedenen Disziplinen der Technischen Informatik sowie weiteren Disziplinen der Informatik und Kommunikationstechnik. Diese Fächer vermitteln ein tiefgehendes Wissen und Verständnis der fortgeschrittenen Konzepte, Methoden und Technologien der gewählten Disziplinen. Auf diese Weise können die Studierenden die fachliche Ausrichtung des Studiums entsprechend ihrer persönlichen Interessen gestalten. Im Studium ist die aktive Teilnahme an einem aktuellen Forschungsprojekt des Instituts fest verankert. Hierbei praktizieren die Studierenden die wissenschaftlich fundierte Analyse und Lösung neuartiger Problemstellungen. Der Studiengang schließt mit der Anfertigung einer Masterarbeit ab: Methoden und Techniken des wissenschaftlichen Arbeitens werden selbstständig auf eine anspruchsvolle Aufgabenstellung angewendet.

Studienbegleitend wird die professionelle Weiterbildung zum Cisco Certified Network Associate Security (CCNA Security) angeboten.

## 2. AbsolventInnenprofil

AbsolventInnen des Studiengangs M. Sc. Informatik und Systems-Engineering verfügen über vertiefte wissenschaftliche und methodische Kompetenzen zur Gestaltung, Analyse und Entwicklung komplexer IT-Systeme. Im Unterschied zum Bachelorstudium liegt der Fokus auf Forschung, Innovation, Führungs- und Managementfähigkeiten sowie auf der systemübergreifenden Anwendung informatischer Technologien. Sie sind befähigt, anspruchsvolle Projekte in Entwicklung, Forschung oder Leitung selbstständig zu verantworten und gestalten aktiv technologische Transformationsprozesse mit.

Der Masterstudiengang Informatik und Systems-Engineering richtet sich an AbsolventInnen mit einem fundierten informatiknahen Erststudium und vertieft deren Kenntnisse gezielt im Spannungsfeld von Software, Hardware, verteilten Systemen, KI, Signalverarbeitung und Kommunikationstechnik.

Im Vergleich zum Bachelorstudiengang, der den Schwerpunkt auf die breite Vermittlung technischer Grundlagen und Anwendungen legt, fokussiert der Master auf:

- Wissenschaftliches Arbeiten und forschungsnahe Projektentwicklung
- Individuelle Spezialisierung durch Wahlmodule
- Interdisziplinäres Systemdenken . Führungs- und Innovationskompetenzen

Die AbsolventInnen des Masterstudiengangs entwickeln ein individuelles Profil in folgenden Bereichen:

- Sie beherrschen die Entwicklung, Analyse und Bewertung komplexer IT-Systeme unter Berücksichtigung technischer, gesellschaftlicher, rechtlicher und ökologischer Rahmenbedingungen.
- Sie sind qualifiziert für Führungsaufgaben in Forschungs- und Entwicklungsprojekten, einschließlich der Leitung interdisziplinärer und internationaler Teams.
- Durch intensive Projektarbeit und das Forschungsprojekt erwerben sie die Fähigkeit, anspruchsvolle Fragestellungen mit wissenschaftlicher Tiefe selbstständig zu bearbeiten.
- Sie können technische Innovationen in Bereichen wie verteilte Systeme, KI, Embedded Systems, Netzwerktechnologien und Multimedia-Kommunikation eigenverantwortlich entwickeln und in industrielle oder wissenschaftliche Anwendungen überführen.
- Sie sind befähigt, durch wissenschaftliches Arbeiten zum Erkenntnisgewinn in der Informatik beizutragen und sich für eine Promotion zu qualifizieren.
- Die Ausbildung fördert neben fachlichen Kompetenzen auch Selbstorganisation, ethisches Urteilsvermögen und interkulturelle Kommunikationsfähigkeit – zentrale Anforderungen in modernen Technologieberufen.
- Berufliche Einsatzfelder sind vielfältig: von Technologieentwicklung in der Industrie über IT-Strategie und Beratung bis hin zur Forschung an Hochschulen und Instituten sowie öffentlichen Einrichtungen.

### 3. Handlungsfelder

Zentrale Handlungsfelder im Studium sind Entwicklung und Design, Forschung und Innovation, Leitung und Management sowie Qualitätssicherung und Tests. Die Profil-Modulmatrix stellt dar, welche Handlungsfelder durch welche Module adressiert werden.

#### **Entwicklung und Design**

Interdisziplinäre Entwicklung und Testung von Algorithmen, Schaltungen, Software, Geräten, kommunikationstechnischen und medientechnologischen Systemen sowie komplexen Rechner-, Kommunikations- und Eingebetteten Systemen.

#### **Forschung und Innovation**

Wissenschaftliche Forschungsarbeit leisten und wissenschaftliche Erkenntnisse anwenden sowie erweitern, von der Grundlagenforschung bis hin zur Industrieforschung, mit der Qualifikation für ein Promotionsstudium.

#### **Leitung und Management**

Fachliche Führungs- und Projektverantwortung übernehmen, einschließlich der Koordination und Leitung von Arbeitsgruppen und international verteilt arbeitender Teams, sowie das Management von Planungs- und Fertigungsprozessen, Projektcontrolling und Produktmanagement.

#### **Qualitätssicherung und Tests**

Durchführung von Qualitätskontrollen und Tests für Produkte und Prozesse, Einsatz von Mess- und Prüftechnologien sowie Koordination von Zertifizierungsprozessen.

## 4. Kompetenzen

Die Module des Studiengangs bilden Studierende in unterschiedlichen Kompetenzen aus, die im Folgenden beschrieben werden. Die Profil-Modulmatrix stellt dar, welche Kompetenzen durch welche Module adressiert werden.

### **Entwicklung und Konzeption komplexer Systeme**

Fähigkeit, große Systeme unter Einbeziehung von elektrotechnischen, softwaretechnischen, mechanischen und optischen Aspekten zu entwerfen und umzusetzen, basierend auf einer gründlichen Anforderungsanalyse unter technischen, ökonomischen, ökologischen und gesellschaftlichen Gesichtspunkten.

### **Prüfung und Bewertung komplexer Systeme**

Planung, Durchführung und Analyse von Tests zur Verifikation und Validierung dieser Systeme, einschließlich der Berücksichtigung von Benutzerperspektiven und technisch-wirtschaftlichen Aspekten.

### **Wissenschaftliches Arbeiten und Forschung**

Beherrschung und Anwendung wissenschaftlicher Methoden, inklusive der Fähigkeit, relevante Literatur zu recherchieren, zu bewerten und zu zitieren, sowie Ergebnisse zu formulieren und zu präsentieren.

### **Projektmanagement und Teamarbeit**

Fähigkeiten in der Organisation, Leitung und Überwachung von Projekten und Teams, auch unter unsicheren Bedingungen, sowie im Treffen von fachlichen und organisatorischen Entscheidungen.

### **Selbstorganisation und autodidaktische Fähigkeiten**

Identifizierung persönlicher Fähigkeiten, effizientes Zeitmanagement und die Fähigkeit zum selbstgesteuerten Lernen.

### **Kommunikation und interkulturelle Kompetenz**

Fähigkeit, wissenschaftliche und technische Ergebnisse überzeugend sowohl in deutscher als auch in englischer Sprache darzustellen und zu verteidigen, unter Einbeziehung internationaler und interdisziplinärer Kontexte.

### **Technische und naturwissenschaftliche Grundlagen**

Umfassendes und vertieftes MINT-Fachwissen und dessen Anwendung auf reale und theoretische Probleme.

### **Nachhaltigkeit und gesellschaftliche Verantwortung**

Bewertung und Entwicklung nachhaltiger und gesellschaftlich verantwortlicher Technologien, einschließlich der Berücksichtigung ethischer Werte.

### **Analyse, Simulation und Abstraktion**

Fähigkeit, komplexe Systeme zu analysieren, wesentliche Merkmale zu abstrahieren und Probleme modellbasiert zu lösen.

### **Führungs- und Entscheidungsverantwortung**

Übernehmen von Verantwortung in fachlichen Führungsaufgaben, Entwicklung von Lösungsstrategien für komplexe Aufgabenstellungen.

### **Anwendung ethischer Werte und Prinzipien in der Praxis**

Einschließen gesellschaftlicher und ethischer Überlegungen in technische Entscheidungen und Designprozesse.

### **Integratives Denken und Handeln in interdisziplinären Teams**

Koordination und Integration von Beiträgen verschiedener Fachgebiete zur Lösung komplexer Aufgaben.

### **Innovation und Kreativität**

Entwickeln neuer Lösungen und Konzepte bei der Bewältigung technischer Herausforderungen.

## 5. Studienverlaufspläne

Im Folgenden sind studierbare Studienverlaufspläne dargestellt. Andere Studienverläufe sind ebenso möglich. Beachten Sie bei Ihrer Planung dabei jedoch, dass jedes Modul in der Regel nur einmal im Jahr angeboten wird. Beachten Sie auch, dass in einem bestimmten Semester und Wahlbereich ggf. mehrer Module gewählt werden müssen, um die dargestellte Summe an ECTS-Kreditpunkten zu erlangen.

### 5.1 Studienverlaufsplan

Sem.	Kürzel	Modulbezeichnung	Pflicht (PF) Wahl- bereich (WB)	ECTS	Prüfungslast	Prüfungsformen mit Gewichtung
1	THI	Theoretische Informatik	PF	5	1	▪ abschließend: mündliche Prüfung [100%]
	VMT	Vertiefung Mathematik	WB	5	≤ 2	▪ wahlabhängig
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ wahlabhängig
	W	Allgemeiner Wahlbereich	WB	10	≤ 4	▪ wahlabhängig
	X	Fachübergreifende Kompetenzen und Soft Skills	WB	5	≤ 2	▪ wahlabhängig
2	FP	Forschungsprojekt	PF	10	0	
	VTI	Vertiefung Technische Informatik	WB	10	≤ 4	▪ wahlabhängig
	W	Allgemeiner Wahlbereich	WB	10	≤ 4	▪ wahlabhängig
3	MAA	Masterarbeit	PF	27	1	▪ abschließend: Abschlussarbeit [100%]
	KOLL	Kolloquium zur Masterarbeit	PF	3	1	▪ abschließend: Kolloquium [100%]

## 5.2 Alternativer Studienverlaufsplan

Sem.	Kürzel	Modulbezeichnung	Pflicht (PF) Wahl- bereich (WB)	ECTS	Prüfungslast	Prüfungsformen mit Gewichtung
1	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ wahlabhängig
	VMT	Vertiefung Mathematik	WB	5	≤ 2	▪ wahlabhängig
	THI	Theoretische Informatik	PF	5	1	▪ abschließend: mündliche Prüfung [100%]
2	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ wahlabhängig
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ wahlabhängig
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ wahlabhängig
3	X	Fachübergreifende Kompetenzen und Soft Skills	WB	5	≤ 2	▪ wahlabhängig
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ wahlabhängig
	VTI	Vertiefung Technische Informatik	WB	5	≤ 2	▪ wahlabhängig
4	FP	Forschungsprojekt	PF	10	0	
	W	Allgemeiner Wahlbereich	WB	5	≤ 2	▪ wahlabhängig
5	MAA	Masterarbeit	PF	27	1	▪ abschließend: Abschlussarbeit [100%]
	KOLL	Kolloquium zur Masterarbeit	PF	3	1	▪ abschließend: Kolloquium [100%]

## 6. Module

Im Folgenden werden die Module des Studiengangs in alphabetischer Reihenfolge beschrieben.

### 6.1 ACC - Advanced Channel Coding

<b>Modulkürzel</b>	ACC_MaTIN2024
<b>Modulbezeichnung</b>	Advanced Channel Coding
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
<b>Dozierende*r</b>	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 programs, 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.

**Modulinhalte**

**Vorlesung / Übungen**

Vorlesung und Übungen werden in einer Lehrveranstaltung kombiniert. Nach der Vorstellung von neuem Lernstoff durch den Dozenten in Form von kurzen Blöcken wird dieser direkt von den Studierenden durch kurze Matlab- und Python-Übungen angewendet und vertieft. Längere Übungsaufgaben werden bereits zu Hause vorbereitet und die verschiedenen Lösungsvorschläge in der Präsenzveranstaltung besprochen.

Inhalte:

- 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, Euklidean 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

Die Studierenden lernen die o.g. Themen in der Vorlesung kennen, erwerben Grundwissen und vertiefen dieses durch Selbststudium mit Hilfe von Literatur, YouTube Videos und anderen Netzressourcen (selbstständige Informationsbeschaffung), sowie in Lerngruppen (Teamwork).

Durch kleine Übungsaufgaben und Programme wird in der Präsenzveranstaltung bereits ein aktiver Umgang mit den vorgestellten Verfahren ermöglicht. Umfangreichere Rechenaufgaben werden am Ende der Veranstaltung behandelt und die Lösungswege diskutiert, um dadurch den Studierenden relevante Problemstellungen vorzustellen und ihre Fähigkeit zur Lösungsfindung zu entwickeln.

Die Studierenden lernen darüber hinaus:

- nachrichtentechnische Systeme zu analysieren und deren Performanz zu ermitteln bzw. abzuschätzen.
- Verfahren der Quellen- und Kanalcodierung und Kryptologie zu vergleichen und zu bewerten
- Kenntnisse auf technische Problemstellungen anzuwenden

**Praktikum**

Vorhandene Simulationsumgebungen wie z.B. die Matlab Communication Toolbox oder AFF3CT ([aff3ct.github.io](https://github.com/aff3ct/aff3ct)) werden verwendet um:

- theoretische Ergebnisse aus Vorlesung und Übung zu überprüfen
- FEC Algorithmen zu implementieren
- BER zu simulieren und die Performanz zu ermitteln, sowie Codes zu vergleichen
- Programme zum Bearbeiten verwandter Probleme anzupassen
- sich mit Standardprogrammen zur Simulation vertraut zu machen
- Teamwork zu üben

**Lehr- und Lernmethoden**                   ▪ Vorlesung / Übungen  
   ▪ Praktikum

**Prüfungsformen mit Gewichtung**                   ▪ begleitend: mündlicher Beitrag oder Übungspraktikum unter Klausurbedingungen [unbenotet] und  
   ▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]

**Workload**                               150 Stunden

<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ 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.</li> <li>▪ - Grundwissen Lineare Algebra</li> <li>▪ - Grundwissen Stochastik</li> <li>▪ - Gute Programmierkenntnisse</li> </ul>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 2 Praktikumstermine und 1 Präsentation</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ R. E. Blahut. Algebraic Codes for Data Transmission. Cambridge University Press, Cambridge, 2003.</li> <li>▪ S. Lin and D. J. Costello. Error Control Coding. ISBN 0-13-042672-5. Prentice-Hall, 2004</li> <li>▪ T. M. Cover and J. A. Thomas. Elements of Information Theory. Wiley, New Jersey, 2006</li> <li>▪ A. Neubauer. Kanalcodierung. Schlembach, Wilburgstetten, 2006.</li> <li>▪ R. Roth. Introduction to Coding Theory. Cambridge, second edition, 2006</li> <li>▪ B. Sklar. Digital Communications. Prentice Hall PTR, Upper Saddle River, New Jersey, 2001</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ ACC in Master Communication Systems and Networks 2020</li> <li>▪ ACC in Master Communication Systems and Networks 2024</li> <li>▪ ACC in Master Technische Informatik 2020</li> </ul>
<b>Perma-Links zur Organisation</b>	<a href="#">ILU course page</a>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.2 AMC - Advanced Multimedia Communications

<b>Modulkürzel</b>	AMC_MaTIN2024
<b>Modulbezeichnung</b>	Advanced Multimedia Communications
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Andreas Grebe/Professor Fakultät IME
<b>Dozierende*r</b>	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.

### Modulinhalte

#### Vorlesung / Übungen

Inhalte zu Multimedia Anwendungen, Encodierung von Multimedia Daten, Integration von Daten, Audio und Video, Multimedia Verkehrsanforderungen, Multimedia Transportprotokolle, RTP und MPEG-TS, Verkehrsmodellierung Burst-Silence-Modell, Quality-of-Service (QoS), Multiservice Netze, IntServ, RSVP, DiffServ, ToS und DSCP, Verkehrsklassifikation, Verkehrsmessung, Traffic Shaping, Network Scheduling, Queueing (FIFO, RR, WRR, WFQ, CB-WFQ, PQ, LLQ), Congestion Avoidance (RED, WRED, CB-WRED), Quality-of-Experience (QoE), MOS Skala, Fehlererkennung, Fehlerkorrektur, FEC, Interleaving, Jitter Buffer.

Die Studierenden bewerten Technologien und Netzwerkarchitekturen von Multiservice-Netzwerken; sie analysieren die Anforderungen an Multimedia-Dienste und -Systeme, entwerfen Architekturen für Multiservice-Netzwerke, implementieren Multiservice-Netzwerke und analysieren Multimedia-Kommunikationsprotokolle und deren Leistungskennzahlen.

#### Praktikum

Vermittlung von Grundkenntnissen und Implementierungswissen zu Multiservice-Netzen oder Multimediaanwendungen in All-IP-Netzen inklusive Planung, Implementierung und Evaluation der Services. Protokollanalyse zur Funktionsanalyse, Performenzanalyse und Fehlerbehebung.

Studierende evaluieren Anforderungen an NGN Services und planen, implementieren und analysieren NGN Services auf Basis der SIP Signalisierung oder alternativer Signalisierungsprotokolle. Sie besitzen die Kompetenzen zur Funktionsanalyse und Fehlersuche durch deep packet inspection (DPI) Protokollanalyse. Sie evaluieren die Performanz von NGN Services in Bezug auf Zeitverhalten, Durchsatz, Verzögerungen, Jitter Robustheit bei Paketfehlern und Sicherheitsaspekten. Individuelle Projektvorschläge von Studierenden sind erwünscht.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>■ Vorlesung / Übungen</li> <li>■ Praktikum</li> </ul>
-------------------------------	--

<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [unbenotet] und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ 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).</li> <li>▪ Bachelor-Level Kenntnisse zu Protokollen und Schichtenmodellen, Internetprotokollen (UDP, TCP, IP, HTTP, FTP), IP Adressierung (IPv4, IPv6), Routingtechniken (IP Routing, Funktionsweise eines Router, Routingprotokolle, RIP, OSPF), Übertragungssystemen und Schicht-2-Protokollen, Ethernet. Verständins von verteilten Systemen und Applikationen, Socketbegriff und Client-/Server-Programmierung, Request-Response Pattern, Publishg-Subscribe Pattern.</li> </ul>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 6 Meilensteintermine und Projektvorstellungen</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ J. Kurose, K. Ross: Computer Networking: A Top-Down Approach, Global Edition, Prentice Hall, 7th ed., 2016</li> <li>▪ A. S. Tanenbaum, D. J. Wetherall: Computer Networks, Pearson , 5th ed., 2013</li> <li>▪ W. Stallings: Foundations of Modern Networking, Pearson Education, 2016</li> <li>▪ H. W. Barz, G. A. Bassett: Multimedia Networks, John Wiley &amp; Sons, 2016</li> <li>▪ T. Szigeti, C. Hattingh, R. Barton, B. Kenneth: End-to-End QoS Network Design: Quality of Service for Rich-Media &amp; Cloud Networks (2nd Edition) End-to-End QoS Network Design: Quality of Service for Rich-Media &amp; Cloud Networks, Cisco Press, 2nd Ed. 2013</li> <li>▪ R. Steinmetz, K. Nahrstedt: „Multimedia Systems“, Springer 2004</li> <li>▪ R. Steinmetz, „Multimedia-Technologie“, Springer 2000</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ AMC in Master Communication Systems and Networks 2020</li> <li>▪ AMC in Master Communication Systems and Networks 2024</li> <li>▪ AMC in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

### 6.3 AMS - Special Aspects of Mobile Autonomous Systems

<b>Modulkürzel</b>	AMS_MaTIN2024
<b>Modulbezeichnung</b>	Special Aspects of Mobile Autonomous Systems
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Chunrong Yuan/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung**

Mobile autonome Systeme  
 Kognitive und Verhalten-basierte Robotik  
 Umweltmodellierung und räumliche Kognition  
 Interaktion und Navigation

**Projekt**

Im Team: Entwicklung eines autonomen Systems mit kognitiven Fähigkeiten und intelligenten Verhalten.  
 Kognitive Fähigkeiten sind z.B.: Objekte mit Sensorik autonom erkennen, ihre räumlichen Positionen bzw. Bewegungen schätzen, das Umfeld modellieren, interpretieren und Karten davon erstellen usw.  
 Intelligente Verhalten lassen sich u.a. durch derartiges Handeln demonstrieren: Autonome und kollisionsfreie Navigation in unbekanntem Umgebungen, Holen bzw. Transportieren von Gegenständen zum bestimmten Zweck, natürliche Interaktionen und Kollaborationen unter Menschen und Robotern.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung</li> <li>▪ Projekt</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [unbenotet] und</li> <li>▪ abschließend: mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\pm$ 3 SWS
<b>Selbststudium</b>	116 Stunden

<b>Empfohlene Voraussetzungen</b>	Kompetenz in der Entwicklung von Software und Projekten Kenntnisse in der Signalverarbeitung und Mathematik
<b>Zwingende Voraussetzungen</b>	Projekt erfordert Anwesenheit im Umfang von: 1 Präsentation
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"><li>▪ Siegart et.al.: Introduction to autonomous mobile robots, MIT Press, 2010</li></ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	AMS in Master Technische Informatik 2020
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.4 ARP - Alternative Rechnerarchitekturen und Programmiersprachen

<b>Modulkürzel</b>	ARP_MaTIN2024
<b>Modulbezeichnung</b>	Alternative Rechnerarchitekturen und Programmiersprachen
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. René Wörzberger/Professor Fakultät IME
<b>Dozierende*r</b>	Prof. Dr. Georg Hartung/Professor Fakultät IME im Ruhestand
<b>Learning Outcome(s)</b>	<p>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.</p>
<b>Modulinhalte</b>	
<b>Vorlesung / Übungen</b>	<p>Kenntnisse über die jeweilige Modellierungsmethode, Programmierverfahren oder Architektur und ihrer Programmierung ("Topics"); Einübung erster Fertigkeiten des Topic in Übungen</p>
<b>Projekt</b>	<p>Anwendung des Topic auf eine selbstgewählte Aufgabenstellung, Analyse der Mittel des Topic am konkreten Beispiel, Synthese mit eigenen Erfahrungen, Teamwork (Bearbeitung in kleiner Gruppe)</p>
<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Projekt</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [50%] und</li> <li>▪ abschließend: mündliche Prüfung oder (elektronische) Klausur [50%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>- Erfahrungen in der Anwendung imperativer Programmiersprachen, insb. C</li> <li>- Grundkenntnisse und Erfahrungen in der Nutzung von Betriebssystemen, insb. Linux</li> <li>- Grundkenntnisse und Erfahrungen im Software Engineering</li> <li>- Grundkenntnisse in Rechneraufbau und Funktionsweise, einschließlich Funktionsweise wichtiger digitaler Bausteine</li> <li>- Grundkenntnisse in Formalen Sprachen und Automatentheorie</li> </ul>
<b>Zwingende Voraussetzungen</b>	Projekt erfordert Anwesenheit im Umfang von: 4 Termine

- Empfohlene Literatur**
- 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.

---

**Verwendung des Moduls in weiteren Studiengängen** ARP in Master Technische Informatik 2020

---

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.5 AVT - Audio- und Videotechnologien

<b>Modulkürzel</b>	AVT_MaTIN2024
<b>Modulbezeichnung</b>	Audio- und Videotechnologien
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr.-Ing. Klaus Ruelberg/Professor Fakultät IME
<b>Dozierende*r</b>	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 Modaluonsverfahren.

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

### Modulinhalte

#### Vorlesung / Übungen

Quellencodierung für Audio- und Videosignale

Kanalmodelle und Kanalcodierung (Fehlerkorrektur & digitale Modulationsverfahren

Broadcast-Übertragungssysteme (DVB - Digital Video Broadcasting)

Aktuelle Verfahren zur Audio- und Videocodierung in Hard- und Software implementieren

Algorithmen und Verfahren zur Audio- und Videocodierung entwickeln

An der Entwicklung und Implementierung von digitalen Rundfunksystemen mitarbeiten

#### Übungen / Praktikum

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>■ Vorlesung / Übungen</li> <li>■ Übungen / Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>■ begleitend: mündlicher Beitrag [unbenotet] und</li> <li>■ abschließend: mündliche Prüfung oder (elektronische) Klausur [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	57 Stunden $\pm$ 5 SWS
<b>Selbststudium</b>	93 Stunden
<b>Empfohlene Voraussetzungen</b>	keine

**Zwingende Voraussetzungen**      Übungen / Praktikum erfordert Anwesenheit im Umfang von: 1 Termin

---

**Empfohlene Literatur**

- 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

---

**Verwendung des Moduls in weiteren Studiengängen**

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

---

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung**      16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links**      Modul Lehrveranstaltung

---

## 6.6 CI - Computational Intelligence

<b>Modulkürzel</b>	CI_MaTIN2024
<b>Modulbezeichnung</b>	Computational Intelligence
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	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.

## Modulinhalte

### *Vorlesung / Übungen*

Optimierungsstrategien

- Problem-Klassifikationen
- Gradientenverfahren
- Simplex-Algorithmen
- Multikriterielle Optimierung und Pareto-Ansätze

Künstliche neuronale Netze

- Künstliche Neuronen
- Netzstrukturen
- Lernalgorithmen

Fuzzy Logik

- Fuzzifizierung
- Inferenz
- Defuzzifizierung

Evolutionäre Algorithmen

- Gen-Repräsentationen
- Selektionsverfahren
- Rekombinations-Methoden
- Mutations-Operatoren

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

Sie kennen das Prinzip des Simplex-Algorithmus und können eine Problemstellung in die für ihn geeignete Standardform überführen und eine Lösung erarbeiten

Die Studierenden können neuronale Netze einordnen und ihre Anwendbarkeit auf Problemstellungen bewerten

Sie können die Parameter neuronaler Netze variieren und ihren Einfluss abschätzen

Sie können Lernverfahren klassifizieren und die Arbeitsweise des Backpropagation Verfahrens beschreiben

Sie kennen die Methodik der Fuzzy Logik und können eine Problemstellung darauf abbilden und das resultierende Systemverhalten begründen

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

Die Studierenden können lineare Probleme mit einem Simplex-Algorithmus lösen

Sie können nichtlineare Probleme der Modellbildung und Klassifizierung mit einem neuronalen Netz lösen

Sie können unscharf definierte Aufgaben mit Hilfe von Fuzzy Logik lösen

Sie können schwierige Probleme mit Heuristiken der evolutionären Algorithmen lösen

---

### *Praktikum*

Anwendung künstlicher neuronaler Netze auf Klassifizierungsaufgaben

Variation und multikriterielle Optimierung von System-Parametern

Fuzzy-basierte Regelung eines Zwei-Größen Regelkreises

Die Studierenden können mit üblichen Werkzeugen der Computational Intelligence umgehen

Die Studierenden können Systemparameter variieren, Messreihen durchführen und Ergebnisse darstellen, bewerten und diskutieren

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

Die Studierenden können Aufgaben in einem kleinen Team lösen

Sie können Optimierungsaufgaben strukturieren und systematisch bearbeiten

Sie können das Verhalten eines Systems bewerten und durch geeignete Modifikationen verbessern

Sie können mit internationaler wissenschaftlicher Literatur umgehen, sie verstehen und Anderen gegenüber darstellen

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung oder Projektarbeit oder Praktikumsbericht</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\triangleq$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Vektorfunktionen, Gradienten
<b>Zwingende Voraussetzungen</b>	Praktikum erfordert Anwesenheit im Umfang von: 2 Termine
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ Domschke W., Drexl A.: Einführung in Operations Research; Springer</li> <li>▪ Zell, A.: Simulation Neuronaler Netze; Oldenbourg</li> <li>▪ Nauck, D. et al.: Neuronale Netze und Fuzzy-Systeme; Vieweg</li> <li>▪ Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing; Springer</li> <li>▪ Gerdes, I. et al.: Evolutionäre Algorithmen; Vieweg</li> <li>▪ Grosse et al.: Taschenbuch der praktischen Regelungstechnik, Fachbuchverlag Leipzig</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	CI in Master Technische Informatik 2020
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.7 DLO - Deep Learning und Objekterkennung

<b>Modulkürzel</b>	DLO_MaTIN2024
<b>Modulbezeichnung</b>	Deep Learning und Objekterkennung
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Jan Salmen/Professor Fakultät IME
<b>Dozierende*r</b>	Prof. Dr. Jan Salmen/Professor Fakultät IME

### Learning Outcome(s)

Die Studierenden lernen, wie Neuronale Netze eingesetzt werden können um vielfältige Aufgaben der Objekterkennung in Bildern zu lösen.  
 Zu diesem Zweck wird an Hand ausgewählter Beispiele eines Neuronalen Netz trainiert dessen Leistungsfähigkeit evaluiert.  
 Damit werden die Studierenden in die Lage versetzt, Deep-Learning-Algorithmen in der beruflichen Praxis zu entwickeln und deren Leistungsfähigkeit zu beurteilen.

### Modulinhalte

#### Vorlesung

Es passiert selten, dass eine Entwicklung so große und weitreichende Auswirkungen hat, wie jüngst das Deep Learning. Betroffen von diesem rasanten Fortschritt sind viele Teilbereiche der Informatik, darunter Bildverarbeitung und hier insbesondere Objekterkennung.

Im Kurs "Deep Learning und Objekterkennung" können die Studierenden lernen, wie künstliche neuronale Netze heute eingesetzt werden, um vielfältige praxisrelevante Aufgaben zu lösen. Dabei lernen sie typische Probleme und Herausforderungen beim Training der tiefen Netze kennen, etwa Überanpassung an Trainingsdaten oder Herausforderungen durch unzureichende Trainingsdaten. Es werden aktuelle Ansätze vorgestellt, die es erlauben, viele solcher Herausforderungen zu meistern und trotzdem zuverlässige Lösungen zu finden.

Die Studierenden lernen schließlich spezielle neuronale Netze kennen, etwa Faltungsnetzwerke, rekurrente Netze, GANs, Autoencoder, usw.

#### Praktikum

Künstliche Neuronale Netze trainieren  
 Evaluation der Leistung von künstlichen neuronalen Netzen

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung oder Schriftliche Prüfung im Antwortwahlverfahren</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\pm$ 3 SWS
<b>Selbststudium</b>	116 Stunden
<b>Empfohlene Voraussetzungen</b>	

**Zwingende Voraussetzungen**      Praktikum erfordert Anwesenheit im Umfang von: 4 Termine

---

- Empfohlene Literatur**
- 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
- 

- Verwendung des Moduls in weiteren Studiengängen**
- 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
- 

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung**      16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links**      Modul Lehrveranstaltung

## 6.8 DMC - Digital Motion Control

<b>Modulkürzel</b>	DMC_MaTIN2024
<b>Modulbezeichnung</b>	Digital Motion Control
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Jens Onno Krahl/Professor Fakultät IME
<b>Dozierende*r</b>	Prof. Dr. Jens Onno Krahl/Professor Fakultät IME
<b>Learning Outcome(s)</b>	
<ul style="list-style-type: none"> <li>▮ Servomotoren kennenlernen und betreiben</li> <li>▮ Servoumrichter kennenlernen und verwenden</li> <li>▮ Digitale Regelalgorithmen nutzen</li> <li>▮ Prozessidentifikation und Parameterestimation</li> <li>▮ Auslegung von Antriebssystemen</li> </ul>	
<b>Modulinhalte</b>	
<b>Vorlesung / Übungen</b>	
<ul style="list-style-type: none"> <li>▮ Aufbau von Servomotoren</li> <li>▮ Aufbau von Servoumrichtern</li> <li>▮ Digitale Regelalgorithmen</li> <li>▮ Prozessidentifikation</li> <li>▮ Auslegung von Antriebssystemen</li> </ul>	
<b>Praktikum</b>	
<ul style="list-style-type: none"> <li>▮ Direct Digital Control</li> <li>▮ Quasi-Stetige Regelung</li> <li>▮ Prädiktor / Beobachter</li> <li>▮ Parametrierung einer Regelung</li> <li>▮ Auswertung von Bode Diagrammen</li> <li>▮ Handlungskompetenz demonstrieren</li> <li>▮ Inbetriebnahme eines Servoreglers</li> <li>▮ Minimierung von Schleppfehlern</li> </ul>	
<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▮ Vorlesung / Übungen</li> <li>▮ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▮ begleitend: Praktikumsbericht [unbenotet] und</li> <li>▮ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\pm$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	RT, DSS

**Zwingende Voraussetzungen**      Praktikum erfordert Anwesenheit im Umfang von: 3 Termine

---

- Empfohlene Literatur**
- Krah, Jens Onno, Vorlesungsskript MC
  - Krah, Jens Onno: Vorlesungsskript RT (Download)
  - Handbuch ServoStar 300: [www.danahermotion.net](http://www.danahermotion.net)
  - Schultz, G.: Regelungstechnik, Oldenbourg Verlag, München-Wien
  - Lutz, Wendt: Taschenbuch der Regelungstechnik, Verlag Harri Deutsch
- 

- Verwendung des Moduls in weiteren Studiengängen**
- DMC in Master Elektrotechnik 2020
  - DMC in Master Elektrotechnik und Informationstechnik 2024
  - DMC in Master Technische Informatik 2020
- 

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung**      16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links**      Modul Lehrveranstaltung

## 6.9 DSP - Digital Signal Processing

<b>Modulkürzel</b>	DSP_MaTIN2024
<b>Modulbezeichnung</b>	Digital Signal Processing
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME
<b>Dozierende*r</b>	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.

### Modulinhalte

#### Vorlesung / Übungen

Signals, Systems and Digital Signal Processing

Discrete-Time Linear Time-Invariant Systems

Ideal Sampling and Reconstruction

Fourier-Transform of Discrete-Time Signals

Discrete Fourier-Transform

Random Signals

Advanced Sampling Techniques

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

Students can calculate the output signal via convolution and determine the frequency response of a given system

Students can implement discrete-time LTI systems in software

#### Praktikum

Review of Probability and Random Variables: Moments, Averages and Distribution Functions

Analysis of Random Signals: Ensemble Averages, Correlation Functions, Power Spectral Density, Random Signals and LTI Systems

Introduction to Advanced Open-Source DSP Software Tools

Applying DSP algorithms in DSP Software for Wireless Communications or Wireless Sensing Applications

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: (Zwischen-)Testat [unbenotet] und</li> <li>▪ abschließend: mündliche Prüfung [100%]</li> </ul>

<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	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
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 8 Termine</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ John G. Proakis and Dimitris K. Manolakis. Digital Signal Processing (4th Edition). Prentice Hall, 2006.</li> <li>▪ Alan V. Oppenheim, Ronald W. Schaffer. Discrete-Time Signal Processing (3rd Edition). Prentice Hall, 2007.</li> <li>▪ Vinay Ingle and John Proakis. Digital Signal Processing using MATLAB. Cengage Learning Engineering, 2011.</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ DSP in Master Communication Systems and Networks 2020</li> <li>▪ DSP in Master Communication Systems and Networks 2024</li> <li>▪ DSP in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.10 ESD - Embedded Systems Design

<b>Modulkürzel</b>	ESD_MaTIN2024
<b>Modulbezeichnung</b>	Embedded Systems Design
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Markus Cremer/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung / Übungen**

Vorlesung und Übungen werden in einer Lehrveranstaltung kombiniert. Nach der Vorstellung von neuem Lernstoff durch den Dozenten in Form von kurzen Blöcken wird dieser direkt von den Studierenden durch Anwendung in ihrer eigenen Projektarbeit umgesetzt und vertieft. Lösungen und Probleme bei der Umsetzung des Lernstoffs in die Praxis werden in der Präsenzveranstaltung gemeinsam mit dem Dozenten diskutiert.

Inhalte:

- Entwicklungsprozess von Embedded Systems
- Finden einer Produktidee zur Verwendung als Modul-Projekt
- Lastenheft und Pflichtenheft
- Recherche und Erstellung Hardware- und Firmwarekonzept
- Proof-of-Concept-Phase
- Erstellung von Schaltplänen
- Leiterplattentechnologie, Herstellungs- und Bestückungsprozesse von Leiterplatten
- Erstellung von Leiterplattenlayouts
- 3D-Modellierung von Gehäusen
- Erstellung der notwendigen Dokumentation der Hardware für die Produktion
- Firmware-Entwicklung
- Aufbau und Validierung des Prototyps
- Finale Projektdokumentation

Die Studierenden lernen die o.g. Themen in der Vorlesung kennen, erwerben Grundwissen und vertiefen dieses durch Selbststudium mit Hilfe von Literatur, YouTube Videos und anderen Netzressourcen (selbstständige Informationsbeschaffung), sowie in Lerngruppen (Teamwork). Die Studierenden lernen den Umgang mit der Software „Altium Designer“ durch selbständiges Durcharbeiten des „Altium Online Curriculum“, das sie mit einem Zertifikat abschließen.

**Projekt**

Nachdem die Studierenden eine eigene Produktidee aus dem Bereich der Embedded Systems gefunden haben, beginnen Sie damit, einen industrie-typischen Entwicklungsprozess für Embedded Systems selbständig zu durchlaufen. Sie starten mit der Spezifikationsphase (Lastenheft, Realisierungskonzepte, Pflichtenheft) und treten dann in die Hardwareentwicklung ein (Schaltpläne, Leiterplattenlayout, Mechanik, Produktionsdokumente). Hier liegt der Hauptfokus der Lehrveranstaltung. Parallel zur Hardwareentwicklung werden Proofs-of-Concept und die Firmwareentwicklung durchgeführt. Nach Abschluss dieser Entwicklungsphasen bestücken die Studierenden ihre selbstentwickelten Leiterplatten und bauen so den ersten Prototyp ihrer Produktidee auf. Final erstellen die Studierenden eine Dokumentation ihres Projekts und stellen ihre Ergebnisse in einer Präsentation vor.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Projekt</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Grundkenntnisse der Elektrotechnik (einfache analoge und digitale Schaltungen) Grundkenntnisse Embedded Systems (Grundlagen Mikrocontroller inkl. Implementierung von Firmware)
<b>Zwingende Voraussetzungen</b>	

**Empfohlene Literatur**

- 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>

**Verwendung des Moduls in weiteren Studiengängen**

- ESD in Master Communication Systems and Networks 2024
- ESD in Master Elektrotechnik und Informationstechnik 2024
- ESD in Master Medientechnologie 2024

**Perma-Links zur Organisation** [ILU-Kurs](#)

**Besonderheiten und Hinweise**

**Letzte Aktualisierung** 16.3.2025, 17:36:53

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.11 ETH - Ethik

<b>Modulkürzel</b>	ETH_MaTIN2024
<b>Modulbezeichnung</b>	Ethik
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	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.

### Modulinhalte

#### Vorlesung / Übungen

- technische, rechtliche und ethische Grundlagen von Datenschutz, Datensicherheit und Privacy,
- Analysemethode für Zielkonflikte zwischen funktionalen Anforderungen und ELSI (ethical / legal / social implications)
- Begriffe aus der Handlungsethik

#### Seminar

- Anwendung ethischer Grundlagen auf konkrete Szenarien

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>■ Vorlesung / Übungen</li> <li>■ Seminar</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>■ begleitend: Fachgespräch und</li> <li>■ abschließend: mündlicher Beitrag</li> </ul>
<b>Workload</b>	150 Stunden

**Präsenzzeit** 45 Stunden  $\cong$  4 SWS

**Selbststudium** 105 Stunden

**Empfohlene Voraussetzungen** kein

**Zwingende Voraussetzungen**

**Empfohlene Literatur**

**Verwendung des Moduls in weiteren Studiengängen** ETH in Master Technische Informatik 2020

**Besonderheiten und Hinweise**

**Letzte Aktualisierung** 16.3.2025, 17:36:53

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.12 FP - Forschungsprojekt

<b>Modulkürzel</b>	FP_MaTIN2024
<b>Modulbezeichnung</b>	Forschungsprojekt
<b>Art des Moduls</b>	Pflichtmodul
<b>ECTS credits</b>	10
<b>Sprache</b>	deutsch und englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	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 Form darstellen,

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

Exemplarische inhaltliche Operationalisierung

### Modulinhalte

#### *Forschungsprojekt*

Ein Professor aus der Fakultät 07 vergibt das Thema des Forschungsprojekts. Dieses Thema stammt aus dem Forschungsgebiet des Professors und ist evtl. an ein bestehendes Forschungsprojekt angelehnt. Zunächst soll sich der Student durch eine fundierte Literaturrecherche in das Themengebiet einarbeiten und den aktuellen Stand der Wissenschaft erarbeiten. Darauf aufbauend soll der Studierende dann ein gegebenes Problem wissenschaftlich untersuchen und lösen. Die Ergebnisse sollen in einem schriftlichen Bericht dokumentiert werden. Hierzu gehört die Darstellung der Grundlagen, der Vorgehensweise, der Lösungsalternativen und die kritische Bewertung der erarbeiteten Lösung. In der mündlichen Ergebnispräsentation soll der Studierende seine Arbeitsergebnisse vorstellen und verteidigen. Sowohl die schriftliche wie auch die mündliche Ergebnispräsentation sollen in Englisch erfolgen.

<b>Lehr- und Lernmethoden</b>	Forschungsprojekt
<b>Prüfungsformen mit Gewichtung</b>	
<b>Workload</b>	300 Stunden
<b>Präsenzzeit</b>	12 Stunden $\cong$ 1 SWS
<b>Selbststudium</b>	288 Stunden
<b>Empfohlene Voraussetzungen</b>	
<b>Zwingende Voraussetzungen</b>	

---

**Empfohlene Literatur**

---

**Verwendung des Moduls in weiteren Studiengängen**      FP in Master Technische Informatik 2020

---

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung**      16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links**      Modul Lehrveranstaltung

---

### 6.13 HIM - Advanced Mathematics

<b>Modulkürzel</b>	HIM_MaTIN2024
<b>Modulbezeichnung</b>	Advanced Mathematics
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch und englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Heiko Knospe/Professor Fakultät IME
<b>Dozierende*r</b>	<ul style="list-style-type: none"><li>▪ Prof. Dr. Heiko Knospe/Professor Fakultät IME</li><li>▪ Prof. Dr. Hubert Randerath/Professor Fakultät IME</li><li>▪ Prof. Dr. Beate Rhein/Professor Fakultät IME</li></ul>

#### 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 (beispielsweise 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).

## Modulinhalte

### *Vorlesung / Übungen*

Eine Kombination von Themen aus folgenden Bereichen:

- Vektoranalysis
- Wahrscheinlichkeitstheorie, Statistik und Multivariate Statistik
- Stochastische Prozesse
- Optimierung

#### 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
- Anwendung von Verfahren der Vektoranalysis zur Lösung von Problemen der Natur- und Ingenieurwissenschaften.
- Anwendung von Verfahren der deskriptiven und induktiven Statistik auf ein- und mehrdimensionale Daten.
- Planung und Durchführung von statistischen Tests.
- Fähigkeit aus Daten relevante Informationen zu gewinnen.
- Anwendung von Optimierungsstrategien zur Lösung von Problemen.

<b>Lehr- und Lernmethoden</b>	Vorlesung / Übungen
<b>Prüfungsformen mit Gewichtung</b>	▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\pm$ 3 SWS
<b>Selbststudium</b>	116 Stunden
<b>Empfohlene Voraussetzungen</b>	Differential- und Integralrechnung für mehrere Variablen sowie Lineare Algebra (Mathematik auf Bachelor-Niveau)
<b>Zwingende Voraussetzungen</b>	
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ K. Burg, H. Haf, F. Wille, A. Meister, Vektoranalysis - Höhere Mathematik für Ingenieure, Naturwissenschaftler und Mathematiker, Springer Vieweg</li> <li>▪ E. Kreyszig, Advanced Engineering Mathematics, John Wiley &amp; Sons</li> <li>▪ L. Papula, Mathematik für Ingenieure und Naturwissenschaftler Band 3, Springer Vieweg</li> <li>▪ R. E. Walpole, R. H. Myers, S. L. Myers, K. Ye, Probability &amp; Statistics for Engineers &amp; Scientists, Prentice Hall</li> <li>▪ S. M. Ross, Probability and Statistics for Engineers and Scientists, Elsevier</li> <li>▪ S. M. Ross, Stochastic Processes, John Wiley &amp; Sons</li> <li>▪ U. Krengel, Einführung in die Wahrscheinlichkeitstheorie und Statistik</li> <li>▪ A. Koop, H. Moock, Lineare Optimierung, Springer</li> <li>▪ R. Reinhardt, A. Hoffmann, T. Gerlach, Nichtlineare Optimierung, Springer</li> <li>▪ M. Ulbrich, S. Ulbrich, Nichtlineare Optimierung, Birkhäuser</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ HIM in Master Communication Systems and Networks 2020</li> <li>▪ HIM in Master Communication Systems and Networks 2024</li> <li>▪ HIM in Master Elektrotechnik 2020</li> <li>▪ HIM in Master Elektrotechnik und Informationstechnik 2024</li> <li>▪ HIM in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.14 IBD - InnoBioDiv

<b>Modulkürzel</b>	IBD_MaTIN2024
<b>Modulbezeichnung</b>	InnoBioDiv
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	0.5 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
<b>Dozierende*r</b>	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
<b>Learning Outcome(s)</b>	
<p>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.</p>	
<b>Modulinhalte</b>	
<b>Seminar</b>	
<p>Entwickeln von Projektideen , Diskussion und Weiterentwicklung der der Projekte</p>	
<b>Projekt</b>	
<p>Die Studierenden erwerben...</p> <ul style="list-style-type: none"> <li>- die Fähigkeit, Konzepte zur Anpassung von Pflanzen an den Klimawandel zu entwickeln und umzusetzen.</li> <li>- die Fähigkeit, Experimente im Bereich der Pflanzenphysiologie, der Bodenbiologie und der Technik zu planen, durchzuführen und zu analysieren.</li> <li>- die Fähigkeit, experimentelle Daten statistisch auszuwerten und zu präsentieren.</li> <li>- die Fähigkeit, wissenschaftliche Ergebnisse zu präsentieren und zu kommunizieren.</li> <li>- die Fähigkeit zur interdisziplinären und interkulturellen Zusammenarbeit und dem Austausch von Ideen mit Studierenden aus verschiedenen MINT-Forschungsbereichen.</li> <li>- Erfahrungen in der Planung und Durchführung von Projekten und in der Teamarbeit</li> </ul> <p>Die Studierenden besitzen am Ende</p> <ul style="list-style-type: none"> <li>- ein tiefes Verständnis für die Wechselwirkungen zwischen Klimaparametern, Pflanzenwachstum und Bodenbiodiversität.</li> <li>- grundlegende Kenntnisse über moderne Technologien wie Robotik, Sensorik und das Internet of Things im Kontext der Pflanzenforschung.</li> <li>- das Bewusstsein für die Bedeutung von Nachhaltigkeit, Ressourcenschonung und Versorgungssicherheit im Kontext des Bevölkerungswachstums und des Klimawandels.</li> </ul>	
<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Seminar</li> <li>▪ Projekt</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	23 Stunden $\cong$ 2 SWS

<b>Selbststudium</b>	127 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>- gute englische Sprachkenntnisse, da in interkulturellen, interdisziplinären Teams gearbeitet wird.</li> <li>- Grundkenntnisse zum IoT und in der Robotik sind wünschenswert</li> <li>- Teamfähigkeit</li> <li>- Grundkenntnisse in der Pflanzenbiologie werden nicht vorausgesetzt</li> </ul>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Seminar erfordert Anwesenheit im Umfang von: 8 Stunden</li> <li>▪ Projekt erfordert Anwesenheit im Umfang von: 5 meetings for project discussions</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ <a href="https://farm.bot/">https://farm.bot/</a></li> <li>▪ 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</li> <li>▪ Agrawal, D. P. (2017). Embedded Sensor Systems. Springer.</li> <li>▪ 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</li> <li>▪ L. Urry, S. Wassermann: Campbell Biology AP Edition (12th Edition), Pearson, ISBN-13: 978-0-13-648687-9</li> <li>▪ Taiz, L., Møller, I. M., Murphy, A., &amp; Zeiger, E. (2022). Plant Physiology and Development. Oxford University Press.</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ IBD in Master Communication Systems and Networks 2020</li> <li>▪ IBD in Master Communication Systems and Networks 2024</li> <li>▪ IBD in Master Elektrotechnik und Informationstechnik 2024</li> <li>▪ IBD in Master Medientechnologie 2024</li> <li>▪ IBD in Master Technische Informatik 2020</li> </ul>
<b>Perma-Links zur Organisation</b>	<a href="#">InnoBioDiv: Student Projects</a>
<b>Besonderheiten und Hinweise</b>	Blockveranstaltung jeweils von Anfang Oktober bis Mitte November (7 Wochen), Optionale Vorbereitungszeit zum Aufbau von Grundkenntnissen in der letzten Septemberwocheeeüte
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.15 IIS - Intelligent Information Systems

<b>Modulkürzel</b>	IIS_MaTIN2024
<b>Modulbezeichnung</b>	Intelligent Information Systems
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Andreas Behrend/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung / Übungen**

Grundlagen der Wissensrepräsentation

- Prädikatenlogik
- relationale, funktionale, baum- bzw. graphbasierte Faktenrepräsentationen (semantische Netze bzw. Ontologien)
- Regelsysteme

Automatisches Schließen und Inferenzmethoden

- Resolutionsprinzip (inkl. Unifikation)
- Vorwärts- oder rückwärtsgerichtete Verkettung
- Fixpunktsemantik

Deklarative Programmiersprachen

- funktionale Programmierung
- relationale (logische) Programmierung, z.B. Prolog, Datalog, SQL und SPARQL

Ausblick auf aktuelle Forschung, z.B. Datenbanksprachen, Parallele Algorithmen, verteilte Systeme, Kombinatorische Optimierung sowie Sprachverarbeitung.

Die Studierenden erarbeiten sich grundlegende Kenntnisse zur Theorie und Anwendung von Methoden zur Wissensrepräsentation, des automatischen Schließens sowie der deskriptiven Programmierung. Sie haben die Operationalisierungskonzepte, die den verschiedenen Klassen von deskriptiven Sprachen zugrunde liegen, verstanden und können für Problemstellungen geeignete Programmierlösungen erarbeiten.

**Praktikum**

Darstellung von Wissen mittels Tupelmengen, Relationen, semantischen Netzen sowie logikbasierten Systemen. Implementierung von Berechnungsproblemen mittels einer funktionalen Programmiersprache (z.B. Haskell) unter Verwendung von Ausdrücken, (algebraischen) Datentypen, unendlichen Datenstrukturen sowie Funktionen höherer Ordnung in Haskell. Das Lösen von Suchproblemen mittels logischer Programmierung und insbesondere rekursiver Ausdrücke. Formulieren von Anfragen mittels relationaler Sprachen (z.B. SPARQL oder Datalog) über Wissensbasen.

**Lehr- und Lernmethoden**                   ▪ Vorlesung / Übungen  
   ▪ Praktikum

**Prüfungsformen mit Gewichtung**                   ▪ begleitend: Übungspraktikum [unbenotet] und  
   ▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]

**Workload**                             150 Stunden

**Präsenzzeit**                         45 Stunden  $\cong$  4 SWS

**Selbststudium**                     105 Stunden

**Empfohlene Voraussetzungen**     Programmierenkenntnisse, Datenstrukturen und Algorithmen

**Zwingende Voraussetzungen**

- Empfohlene Literatur**
- 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

---

**Verwendung des Moduls in weiteren Studiengängen** IIS in Master Technische Informatik 2020

---

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:53

---

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.16 ITF - IT-Forensik

<b>Modulkürzel</b>	ITF_MaTIN2024
<b>Modulbezeichnung</b>	IT-Forensik
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	Jürgen Bornemann/Lehrbeauftragter

### Learning Outcome(s)

Teilnehmer sind sich der Gefahren durch Angriffe von außen und innen auf technische Systeme bewusst werden und sind in der Lage, digitale Beweise aufzuspüren, sicherzustellen und weiter verwertbar zu analysieren.

### Modulinhalte

#### Vorlesung / Übungen

- | Grundbegriffe der Cyber Security und digitale Forensik
- | Typische Schwachstellen, Bedrohungen und Risiken
- | Gefahren bei mobilen Systemen, Home-Office, WLAN's
- | Grundlagen und Arbeitsweisen der IT-Forensik
- | Forensische Dokumentationserstellung
- | Gängige Werkzeuge für forensische Untersuchungen
- | Digitale Beweise erkennen und sichern
- | Open-Source-Forensik
- | Dateisystem-Forensik
- | Forensische Analyse mobiler Systeme
- | Schwachstellen, Bedrohungen, Angriffe auf Netzwerkstrukturen
- | KALI Linux – Operating System für Vulnerability und Pentesting

### Projekt

Studierenden können fallbezogene forensische Aufgaben und Vorfälle mit dem jeweiligen erlernten Wissen eigenständig oder in Arbeitsgruppen bearbeiten. Sie zeigen dabei, wie sie digitale Beweise sicherstellen, analysieren und dokumentieren.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Projekt</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\triangleq$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	

---

**Zwingende  
Voraussetzungen**

---

**Empfohlene Literatur**

- Verwendung des Moduls in weiteren Studiengängen**
- 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 Technische Informatik 2020

---

**Besonderheiten und  
Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:53

---

**Web-Modulhandbuch-  
Editor-Links** Modul Lehrveranstaltung

## 6.17 KOGA - Kombinatorische Optimierung und Graphenalgorithmen

<b>Modulkürzel</b>	KOGA_MaTIN2024
<b>Modulbezeichnung</b>	Kombinatorische Optimierung und Graphenalgorithmen
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Hubert Randerath/Professor Fakultät IME
<b>Dozierende*r</b>	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.

### Modulinhalte

#### Vorlesung / Übungen

- KOGA-Grundlagen: Grundbegriffe der Graphentheorie und der Kombinatorischen Optimierung
- Minimale aufspannende Bäume: Algorithmen von Kruskal, Prim und Tarjan, Greedy-Algorithmen, Matroide, Steinerbäume, Netzwerk-Design
- Lineare Programme: Struktur, Modellierung, Transformation in die Standardform, Simplex-Verfahren, Dualitätstheorie
- Gewichtete Matchings und das Chinesische Briefträger Problem: Gewichtete Matchings in bipartiten Graphen, Gewichtete Matchings in nicht-bipartiten Graphen, Algorithmus von Floyd-Warshall, Algorithmus von Fleury, Effizienter Algorithmus für das Chinesische Briefträger Problem
- Flüsse in Netzwerken: Grundlagen der Netzwerktheorie, Algorithmus von Dinic, Kostenminimale Flüsse
- Spezielle Diskrete und Kombinatorische Optimierungsprobleme: Travelling Salesman Problem, das Frequenzzuweisungsproblem, Scheduling-Probleme, Routing-Probleme

#### Übungen / Praktikum

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Übungen / Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: mündlicher Beitrag und</li> <li>▪ begleitend: mündlicher Beitrag und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	57 Stunden $\pm$ 5 SWS
<b>Selbststudium</b>	93 Stunden
<b>Empfohlene Voraussetzungen</b>	Grundlagenwissen Graphentheorie Grundlagenwissen Algorithmik

<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"><li>▪ Vorlesung / Übungen erfordert Anwesenheit im Umfang von: 1 Vortragstermin</li><li>▪ Übungen / Praktikum erfordert Anwesenheit im Umfang von: 1 Termin</li></ul>
----------------------------------	---

---

**Empfohlene Literatur**

<b>Verwendung des Moduls in weiteren Studiengängen</b>	KOGA in Master Technische Informatik 2020
--	---

---

**Besonderheiten und Hinweise**

<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
------------------------------	---------------------

<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung
---------------------------------------	-------------------------

## 6.18 KOLL - Kolloquium zur Masterarbeit

<b>Modulkürzel</b>	KOLL_MaTIN2024
<b>Modulbezeichnung</b>	Kolloquium zur Masterarbeit
<b>Art des Moduls</b>	Pflichtmodul
<b>ECTS credits</b>	3
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	3
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	verschiedene Dozenten*innen / diverse lecturers
<b>Learning Outcome(s)</b>	
<ul style="list-style-type: none"> <li>- Darstellung von Forschungsergebnissen in einer Präsentation in vorgegebenem engen zeitlichen Rahmen</li> <li>- Fachliche und außerfachliche Bezüge der eigenen Arbeit darstellen und begründen</li> <li>- Eigene Lösungswege und gewonnene Erkenntnisse darstellen und diskutieren</li> </ul>	
<b>Modulinhalte</b>	
<i>Kolloquium</i>	
<p>Das Kolloquium dient der Feststellung, ob die Studentin oder der Student befähigt ist, die Ergebnisse der Masterarbeit, ihre fachlichen und methodischen Grundlagen, fachübergreifende Zusammenhänge und außerfachlichen Bezüge mündlich darzustellen, selbständig zu begründen und ihre Bedeutung für die Praxis einzuschätzen</p>	
<b>Lehr- und Lernmethoden</b>	Kolloquium
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ abschließend: Kolloquium [100%]</li> </ul>
<b>Workload</b>	90 Stunden
<b>Präsenzzeit</b>	0 Stunden $\pm$ 0 SWS
<b>Selbststudium</b>	90 Stunden
<b>Empfohlene Voraussetzungen</b>	
<b>Zwingende Voraussetzungen</b>	Siehe Prüfungsordnung §29, Abs. 2
<b>Empfohlene Literatur</b>	
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ KOLL in Master Communication Systems and Networks 2020</li> <li>▪ KOLL in Master Communication Systems and Networks 2024</li> <li>▪ KOLL in Master Elektrotechnik 2020</li> <li>▪ KOLL in Master Elektrotechnik und Informationstechnik 2024</li> <li>▪ KOLL in Master Medientechnologie 2020</li> <li>▪ KOLL in Master Medientechnologie 2024</li> <li>▪ KOLL in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	Siehe auch Prüfungsordnung §29.
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.19 KRY - Cryptography

<b>Modulkürzel</b>	KRY_MaTIN2024
<b>Modulbezeichnung</b>	Cryptography
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Heiko Knospe/Professor Fakultät IME
<b>Dozierende*r</b>	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).

### Modulinhalte

#### Vorlesung / Übungen

- \* 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

#### Praktikum

- 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.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
-------------------------------	--

<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [unbenotet] und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
--------------------------------------	--

<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\cong$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Mathematik (Bachelor Niveau) und Programmierkenntnisse.
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 3 Termine</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ M. Bellare, P. Rogaway, Introduction to Modern Cryptography, UCSD CSE</li> <li>▪ H. Delfs, H. Knebl, Introduction to Cryptography, Springer</li> <li>▪ S. Goldwasser, M. Bellare, Lecture Notes on Cryptography, MIT</li> <li>▪ J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography, Springer</li> <li>▪ J. Katz, Y. Lindell, Introduction to Modern Cryptography, CRC Press</li> <li>▪ H. Knospe, A Course in Cryptography, American Mathematical Society</li> <li>▪ C. Paar, J. Pelz, Understanding Cryptography. Springer</li> <li>▪ N.P. Smart, Cryptography Made Simple, Springer</li> <li>▪ K. H. Rosen, Discrete Mathematics and its Applications, McGraw-Hill</li> <li>▪ V. Shoup, A Computational Introduction to Number Theory and Algebra, Cambridge University Press</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ KRY in Master Communication Systems and Networks 2020</li> <li>▪ KRY in Master Communication Systems and Networks 2024</li> <li>▪ KRY in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:53
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.20 LCSS - Large and Cloud-based Software-Systems

<b>Modulkürzel</b>	LCSS_MaCSN2024
<b>Modulbezeichnung</b>	Large and Cloud-based Software-Systems
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. René Wörzberger/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung / Übungen**

- | Formal fundierter Umgang mit Qualitätsanforderungen an Verfügbarkeit, Performance, Kapazität und Kosteneffizienz
- | Vor- und Nachteile grundlegender Systemarchitekturstile, beispielsweise Microservice-Architekturen
- | Skalierung von Systemen und einzelnen Tiers, auch in Hinblick auf mögliche Deployment-Strategien wie Canary- oder AB-Deployment, sowie damit verbundene Load-Balancing-Strategien (z. B. Consisten Hashing)
- | Fortgeschrittene Einsatzmöglichkeiten von Virtualisierung, insbesondere Container-Virtualisierung und -Orchestrierung, beispielsweise mit Docker und Kubernetes
- | Auswahl geeigneter Kommunikationsmuster und -protokolle, insbesondere HTTP und Derivate wie Websockets, Server-sent Events und, gRPC
- | Auswahl zweckdienlicher API-Technologien und -Designphilosophien wie REST und GraphQL
- | Verwendung grundlegender Sicherheitsprotokolle wie TLS, OAuth2, JWT und OpenID Connect
- | Asynchrone, ereignisgetriebene Kommunikation über Messaging- und Streaming-Plattformen wie Apache Kafka
- | Auswahl geeigneter Datenbankmodelle (relational, Key-value-, Graph-, Dokumenten-orientiert), notwendiger Konsistenz-Level, sowie Sharding am Beispiel von PostgreSQL, Neo4J, Apache Cassandra und Redis
- | Strategien für das Caching von Daten, insbesondere von HTTP-Responses (Web Caching).

**Projekt**

- | Formulierung und Präsentation einer selbstgewählten Forschungsfrage aus dem Themenfeld der Lehrveranstaltung
- | Entwurf von Forschungsprototypen, Test-Szenarien, Messverfahren etc. zur Beantwortung der Forschungsfrage inkl. Dokumentation und paarweisem, konstruktiven Review und Aussprache vor Ort zwischen teilnehmenden Teams
- | Abschließende Präsentation der Forschungsergebnisse
- | Dokumentation der Forschungsergebnisse in einem Report gemäß IEEE-Vorlage

- Lehr- und Lernmethoden**
- Vorlesung / Übungen
  - Projekt

- Prüfungsformen mit Gewichtung**
- begleitend: Projektarbeit [60%] und
  - abschließend: (elektronische) Klausur oder mündliche Prüfung [40%]

**Workload** 150 Stunden

**Präsenzzeit** 45 Stunden  $\cong$  4 SWS

**Selbststudium** 105 Stunden

- Empfohlene Voraussetzungen**
- fortgeschrittene Programmierkenntnisse
  - grundlegende Kenntnisse in Web-Technologien
  - grundlegende Kenntnisse in Datenbanken
  - grundlegende Kenntnisse in Software-Architekturen
  - grundlegende Kenntnisse in der Unified Modeling Language (UML)

**Zwingende Voraussetzungen** Projekt erfordert Anwesenheit im Umfang von: 4 Termine

- Empfohlene Literatur**
- 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

- Verwendung des Moduls in weiteren Studiengängen**
- LCSS in Master Medientechnologie 2024
  - LCSS in Master Technische Informatik 2020
  - LCSS in Master Communication Systems and Networks 2024

**Perma-Links zur Organisation** [llu-Kurs](#)

---

**Besonderheiten und  
Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:53

---

**Web-Modulhandbuch-  
Editor-Links** Modul Lehrveranstaltung

## 6.21 MAA - Masterarbeit

<b>Modulkürzel</b>	MAA_MaTIN2024
<b>Modulbezeichnung</b>	Masterarbeit
<b>Art des Moduls</b>	Pflichtmodul
<b>ECTS credits</b>	27
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	3
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	verschiedene Dozenten*innen / diverse lecturers

### Learning Outcome(s)

Das Modul vermittelt folgende Kenntnisse und Fertigkeiten:

- Komplexe Aufgabenstellungen beurteilen
- Selbständiges 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 Ingenieurstätigkeit 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)."

### Modulinhalte

#### Abschlussarbeit

Die Masterarbeit ist eine schriftliche Hausarbeit. Sie soll zeigen, dass die oder der Studierende befähigt ist, innerhalb einer vorgegebenen Frist ein Thema aus ihrem oder seinem Fachgebiet sowohl in seinen fachlichen Einzelheiten als auch in den fachübergreifenden Zusammenhängen nach wissenschaftlichen und fachpraktischen Methoden selbstständig zu bearbeiten. Die interdisziplinäre Zusammenarbeit kann auch bei der Abschlussarbeit berücksichtigt werden.

<b>Lehr- und Lernmethoden</b>	Abschlussarbeit
<b>Prüfungsformen mit Gewichtung</b>	▪ abschließend: Abschlussarbeit [100%]
<b>Workload</b>	810 Stunden
<b>Präsenzzeit</b>	0 Stunden $\pm$ 0 SWS
<b>Selbststudium</b>	810 Stunden
<b>Empfohlene Voraussetzungen</b>	Siehe Prüfungsordnung §26
<b>Zwingende Voraussetzungen</b>	siehe Prüfungsordnung §26 Abs. 1

---

**Empfohlene Literatur**

---

- Verwendung des Moduls in weiteren Studiengängen**
- 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

---

**Besonderheiten und Hinweise** Siehe auch Prüfungsordnung §24ff. Kontaktieren Sie frühzeitig einen Professor der Fakultät für die Erstbetreuung der Abschlussarbeit.

---

**Letzte Aktualisierung** 16.3.2025, 17:36:54

---

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.22 MCI - Mensch-Computer-Interaktion

<b>Modulkürzel</b>	MCI_MaTIN2024
<b>Modulbezeichnung</b>	Mensch-Computer-Interaktion
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Jonas Schild/Professor Fakultät IME
<b>Dozierende*r</b>	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).

**Modulinhalte**

**Vorlesung**

Modelle und Gestaltungsprinzipien interaktiver Systeme  
 Relevante Definitionen, Normen und Richtlinien, Kognitive Aspekte  
 Heuristiken, Best Practices und Style Guides  
 Steuerungsmöglichkeiten: Dedizierte Ein-/Ausgabegeräte und Steuerungsmethoden  
 Interaktion in Computerspielen, Structure of Games, Game Input, Game Feel: Metrics, Input, Response, Experiences  
 User Experience Engineering: Fun, Flow, Immersion, Presence, Decision Engineering, Information Balancing  
 Prinzipien spezieller interaktiver Systeme wie Mobile, Context Aware Computing, 3D Interaction  
 Experimentelle Forschung: Wiss Fragestellung, Hypothesen, technikethische Kriterien  
 Evaluations-Methoden (Self-reporting tools, Physiopsychologische Verfahren, Nutzungsmetriken)  
 Experiment Design: Between Group, Within Group, Ablauf, Vorbereitung, Datenschutz  
 Statistische Analyse: Skalenniveaus, Deskriptive Statistik, T-Tests, ANOVA, Regression, Korrelation  
 Umfragen: Stichproben und Stichprobenauswahl, Fehlerquellen, Fragebögen, Evaluation von Umfragen

**Praktikum**

Methoden und Begriffe der MCI-Forschung anwenden  
 Interaktive Prototyen konzipieren und implementieren  
 Mit Interaktionsmethoden und forschungsnahen Ein-/Ausgabesystemen experimentieren  
 Nutzerstudien konzipieren, durchführen und analysieren  
 Quantiative und/oder Qualitative Methoden der User Experience Analyse anwenden  
 Ergebnisse präsentieren, diskutieren und reflektieren  
 In Teams zusammenarbeiten und koordinieren  
 Forschungsberichte verfassen

**Seminar**

Wiss. Literatur lesen, wiedergeben und verdeutlichen  
 Wiss. Methoden der Mensch-Maschine-Interaktion am aktuellen Forschungsstand aufbereiten  
 Wiss. Recherche- und Zitationsarbeit  
 Präsentieren von aktuellen Forschungsarbeiten

**Lehr- und Lernmethoden**

- Vorlesung
- Praktikum
- Seminar

**Prüfungsformen mit Gewichtung**

- begleitend: Projektarbeit [100%]

**Workload** 150 Stunden

**Präsenzzeit** 45 Stunden  $\pm$  4 SWS

**Selbststudium** 105 Stunden

**Empfohlene Voraussetzungen**  
 Programmierkenntnisse  
 Computergrafik

**Zwingende Voraussetzungen**

- Praktikum erfordert Anwesenheit im Umfang von: 2 Termine
- Seminar erfordert Anwesenheit im Umfang von: Vorträge und Schlusspräsentation

**Empfohlene Literatur**

- 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

---

**Verwendung des Moduls in weiteren Studiengängen**

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

---

**Besonderheiten und Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:54

---

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

**6.23 MLWR - Maschinelles Lernen und wissenschaftliches Rechnen**

<b>Modulkürzel</b>	MLWR_MaTIN2024
<b>Modulbezeichnung</b>	Maschinelles Lernen und wissenschaftliches Rechnen
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Beate Rhein/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung / Übungen**

- Übersicht Maschinelles Lernen
- End-to-End Projekt Maschinelles Lernen
  - Datenvorbereitung
  - Skalierung
- Klassifikationsverfahren
  - Performanzmaße
  - Verfahren
- Regressionsverfahren
  - Klassische Verfahren
  - Verfahren des Maschinellen Lernens
- Unüberwachtes Lernen
- Einführung in Neuronale Netze
  - Perzeptron
  - Feed Forward Neural Network
  - Architektur
  - Training
- Einführung in große Sprachmodelle
  - Embeddinges
  - Transformer Architektur
  - Klassifikation und Regression mit LLMs
  - Retrieval Augmented Generation
- Erklärbares und faires Maschinelles Lernen

**Praktikum**

Anwendung und Programmierung von Verfahren der Approximation, der multikriteriellen Optimierung oder des maschinellen Lernens  
 numerische Verfahren effizient implementieren  
 Algorithmen hinsichtlich ihrer Komplexität bewerten

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [20%] und</li> <li>▪ begleitend: Projektarbeit [20%] und</li> <li>▪ abschließend: mündliche Prüfung [80%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\pm$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Grundkenntnisse in Wahrscheinlichkeitsrechnung und maschinellem Lernen
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen erfordert Anwesenheit im Umfang von: 6 Stunden</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Vorlesung / Übungen</li> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 2 Termine</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>

- Empfohlene Literatur**
- A. Geron: Hand-on Machine Learning, O'Reilly Verlag
  - J. Alamar: Hands-on Large Language Models, O'Reilly Verlag

- Verwendung des Moduls in weiteren Studiengängen**
- 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 Technische Informatik 2020

---

**Besonderheiten und Hinweise**

**Letzte Aktualisierung** 16.3.2025, 17:36:54

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

## 6.24 NGN - Next Generation Networks

<b>Modulkürzel</b>	NGN_MaTIN2024
<b>Modulbezeichnung</b>	Next Generation Networks
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Andreas Grebe/Professor Fakultät IME
<b>Dozierende*r</b>	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.

### Modulinhalte

#### *Vorlesung / Übungen*

Vermittlung von Grundkenntnissen und Implementierungswissen über die Definition von Next Generation Network (NGN) durch ITU-T, IP Multimedia Subsystem (IMS) durch 3GPP und ETSI sowie Next Generation Internet (NGI) Definition durch IETF, ITU-T Standards, Multimedia Services in NGN, VoIP, Video-over-IP, RTP Kapselung, Service Signaling, SIP-Protokoll, SIP Digest Authentication, SDP-Servicebeschreibung und -Fähigkeiten, SIP-Server, Session Border Controller (SBC), SIP-Gateway-Technologien, SIP-Routing, NAT-Gateways, NAT-Lösung, SRR, STUN, TURN, IMS in Mobilfunknetzen, IMS in Festnetzen, VoIP in Unternehmensnetzen. IMS in virtuellen Netzwerk-Core.

Studierende evaluieren Anforderungen an NGN Services und planen, implementieren und analysieren NGN Services auf Basis der SIP Signalisierung oder alternativer Signalisierungsprotokolle. Sie besitzen die Kompetenzen zur Funktionsanalyse und Fehlersuche durch deep packet inspection (DPI) Protokollanalyse. Sie evaluieren die Performanz von NGN Services in Bezug auf Zeitverhalten, Durchsatz, Verzögerungen, Jitter Robustheit bei Paketfehlern und Sicherheitsaspekten.

#### *Praktikum*

Konzepte und Technologien für NGN oder NGI benennen, strukturieren, einordnen. Netzanalysetechniken und Tools beherrschen, Methoden für NGN Services und zur Netzplanung kennen.

Projektpraktikum in einem kleinen Team (2-3 Teammitglieder) zu aktuellen Technologien im Bereich der NGN-Dienste und NGI-Dienste.

NGN/NGI Umgebung und NGN Service planen, implementieren und analysen inklusive der Sicherheitsaspekte und Protokollanalyse mit Evaluierung der Performance.

Die Ergebnisse werden während des Praktikums überprüft, in einem Bericht zusammengefasst und in der Klasse präsentiert.

Individuelle Projektvorschläge von Studierenden sind erwünscht.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [unbenotet] und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\pm$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Modul NGN: Bachelor Level Networking Knowledge and Skills like prerequisite to NGN. 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).</li> <li>▪ Bachelor-Level Kenntnisse zu Protokollen und Schichtenmodellen, Internetprotokollen (UDP, TCP, IP, HTTP, FTP), IP Adressierung (IPv4, IPv6), Routingtechniken (IP Routing, Funktionsweise eines Router, Routingprotokolle, RIP, OSPF), Übertragungssystemen und Schicht-2-Protokollen, Ethernet. Verständins von verteilten Systemen und Applikationen, Socketbegriff und Client-/Server-Programmierung, Request-Response Pattern, Publishg-Subscribe Pattern.</li> </ul>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Praktikum erfordert Anwesenheit im Umfang von: 6 Meilensteintermine und Projektvorstellungen</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ J. Kurose, K. Ross: Computer Networking: A Top-Down Approach, Global Edition, Prentice Hall, 7th ed., 2016</li> <li>▪ A. S. Tanenbaum, D. J. Wetherall: Computer Networks, Pearson , 5th ed., 2013</li> <li>▪ U.Trick, F. Weber: SIP und Telekommunikationsnetze: Next Generation Networks und Multimedia over IP – konkret, De Gruyter Oldenbourg Verlag, 4. Auflage 2015</li> <li>▪ J. F. Durkin: Voice-enabling the Data Network,Cisco Press 2010</li> <li>▪ G. Camarillo, M.A. García-Martín: The 3G IP Multimedia Subsystem (IMS), John Wiley Verlag, 2006</li> <li>▪ W. Stallings: Foundations of Modern Networking, Pearson Education, 2016</li> <li>▪ J. Doherty: SDN and NFV Simplified, Pearson Education, 2016</li> <li>▪ J. Edelman: Network Programmability and Automation, O'Reilly 2018</li> <li>▪ J. van Meggelen, R. Bryant, L. Madsen: Asterisk: The Definitive Guide: Open Source Telephony for the Enterprise, O'Reilly Media, 5th Ed. 2019</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ NGN in Master Communication Systems and Networks 2020</li> <li>▪ NGN in Master Communication Systems and Networks 2024</li> <li>▪ NGN in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:54
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.25 PAP - Parallele Programmierung

<b>Modulkürzel</b>	PAP_MaTIN2024
<b>Modulbezeichnung</b>	Parallele Programmierung
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME
<b>Dozierende*r</b>	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.

### Modulinhalte

#### Vorlesung

- Grundlegende Konzepte, Modelle und Technologien der parallel Verarbeitung (Parallelität, Nebenläufigkeit, SISD, SIMD, MISD, MIMD, loose- und eng-gekoppelte Systeme, verteilte Systeme)
- Parallele Leistungsmaße (Speedup, Effizienz)
- Aufbau von GPUs
- Parallele Algorithmen für GPUs

#### Praktikum

- Aufgabenstellungen in Bezug auf die Programmierung paralleler Programme analysieren und strukturieren, einschlägige parallele Hardwarearchitektur zuordnen und auf Paralleldesign übertragen
- Parallele Programme implementieren (Multicore-HW mit Threads und GPUs)
- Parallele Programme unter Einsatz geeigneter Tools analysieren und Ergebnisse nachvollziehbar darstellen
- Leistungsfähigkeit paralleler Programme abschätzen und analysieren
- Information aus englischen Originalquellen und Standards ableiten

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung</li> <li>▪ Praktikum</li> </ul>
-------------------------------	--

<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [unbenotet] und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung oder Hausarbeit [100%]</li> </ul>
--------------------------------------	--

<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\cong$ 3 SWS
<b>Selbststudium</b>	116 Stunden
<b>Empfohlene Voraussetzungen</b>	Zur Bearbeitung der Übungsaufgaben werden Programmierkenntnisse und der Umgang mit konsolenorientierten Programmen in Linux-basierten Betriebssystemen vorausgesetzt.
<b>Zwingende Voraussetzungen</b>	Praktikum erfordert Anwesenheit im Umfang von: 2 Termine
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>■ P. Pacheco: An Introduction to Parallel Programming, Morgan Kaufmann, 2011</li> <li>■ R. Oechsle: Parallele und verteilte Anwendungen in Java, Hanser, 2011</li> <li>■ B. Goetz, J. Bloch, J. Bowbeer, D. Lea, D. Holmes, T. Peierls: Java Concurrency in Practice, Addison-Wesley Longmann 2006</li> <li>■ Jason Sanders: CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Longman, 2010</li> <li>■ Andrew S. Tanenbaum, Herbert Bos: Modern Operating Systems, 4th Edition, 2015</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>■ PAP in Master Medientechnologie 2020</li> <li>■ PAP in Master Medientechnologie 2024</li> <li>■ PAP in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:54
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.26 PLET - Projektleitung

<b>Modulkürzel</b>	PLET_MaTIN2024
<b>Modulbezeichnung</b>	Projektleitung
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Michael Gartz/Professor Fakultät IME
<b>Dozierende*r</b>	<ul style="list-style-type: none"> <li>▪ Prof. Dr. Michael Gartz/Professor Fakultät IME</li> <li>▪ Prof. Dr. Uwe Oberheide/Professor Fakultät IME</li> </ul>

### 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.

**Modulinhalte**

**Seminar**

Begriffe klassifizieren und abgrenzen; charakteristische Eigenschaften von Entwicklungsprojekten erläutern; technische und wirtschaftliche Ziele in Entwicklungsprojekten abstrakt definieren; Aufgaben des Projektmanagements abstrakt definieren, erläutern und begründen; /  
 grundlegende Erfolgs- und Misserfolgskriterien im Projektmanagement benennen und erläutern: 1) unerwartete Technische Probleme, 2) ungenügende Personalqualifikation, 3) unklare oder widersprüchliche Anforderungen, 4) schlechtes Projektmanagement, 5) ungenügende Unterstützung durch das Senior Management, 6) erweiterte Herausforderungen identifizieren, die durch eine arbeitsteilige Projektbearbeitung entstehen  
 ausgewählte lineare und agile Vorgehensmodelle erläutern:  
 Phasenmodell, V-Modell, SCRUM, Timebox-Modell /  
 Vorgehensmodelle einordnen und vergleichen / Projektentscheidungen  
 grundlegende Aufgaben und erwartete Ergebnisse in Entwicklungsprojekten charakterisieren /  
 Planung und Steuerung der Produktqualität und des Entwicklungsprozesses /  
 Projektrisikomanagement, Ressourcenmanagement /  
 Dokumentation des Entwicklungsverlaufs /  
 Spezifikation der Anforderungen und des Produktdesigns/  
 Produkt-Entwicklung, -Herstellung, -Dokumentation /  
 Verifikation und Validierung /  
 Produktfreigabe und Produktüberwachung  
 Instrumente zur Steuerung von Teamprozessen charakterisieren  
 für das Lehrveranstaltungselement "Projekt" wesentliche Managementaufgaben, Meilensteine und Projektdokumente planen  
 wesentliche Managementaufgaben gedanklich durchführen und vorausschauend Projektrisiken ermitteln  
 wesentliche Projektmanagementwerkzeuge für Projekt(zeit)planung und Anforderungsspezifikation zielgerichtet handhaben  
 Vorgehen zur Teambildung planen, zu erwartende Herausforderungen und sinnvolle Maßnahmen ableiten  
 potenzielle Konfliktsituationen im Team erkennen und Handlungsalternativen diskutieren

**Projekt**

Team leiten und dabei: 1) den Teammitgliedern das grundlegende Vorgehen im Projekt erläutern, 2) Kompetenzen der Teammitglieder erfassen und einordnen, 3) inhaltliche und terminliche Ziele vereinbaren  
 Projekt leiten /  
 Projektrisiken ermitteln und sinnvolle Milderungsmaßnahmen planen, z.B. frühe Machbarkeitsstudien /  
 Projektzeitplan erstellen und pflegen /  
 agiles Vorgehensmodell in Verbindung mit Timebox-Modell anwenden, um einen minimalen Projekterfolg sicherzustellen /  
 ein für das Team erreichbares Minimalziel definieren /  
 erweiterte Ziele für schnelle Teams definieren /  
 Projektabschlussbericht verfassen  
 Team leiten und dabei:  
 1) Zielerreichung kontrollieren und steuern,  
 2) Zusammenarbeit der Teammitglieder koordinieren,  
 3) Konfliktsituationen im Team erkennen und auflösen  
 Projekt leiten:  
 Projektabschnitte planen, Projekt detaillieren, Aufgaben sinnvoll Teammitgliedern zuordnen /  
 Inhaltliche Reviews mit den Teammitgliedern planen und durchführen /  
 Projektergebnisse im Team bewerten: Vorgehen im aktuellen Projektabschnitt retrospektiv bewerten und ggf. für den nächsten Projektabschnitt modifizieren /  
 Projektabschnitte dokumentieren / Zugriff auf gemeinschaftlich genutzte Laborressourcen planen / Projektentscheidungen mit dem Team treffen

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>■ Seminar</li> <li>■ Projekt</li> </ul>
-------------------------------	--

<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [unbenotet]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	23 Stunden $\cong$ 2 SWS
<b>Selbststudium</b>	127 Stunden
<b>Empfohlene Voraussetzungen</b>	<p>grundlegende Kenntnisse zum Projektmanagement</p> <p>grundlegende Erfahrungen als Mitglied von Projektteams</p>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Projekt erfordert Anwesenheit im Umfang von: 8 Termine</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Projekt</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ Hans-D. Litke, „Projektmanagement, Methoden, Techniken, Verhaltensweisen, Evolutionäres Projektmanagement“, Hanser</li> <li>▪ Ken Schwaber: Agiles Projektmanagement mit Scrum (Microsoft Press)</li> <li>▪ Litke, Kunow, Schulz-Wimmer, „Projekt-Management“, Taschenguide , Haufe</li> <li>▪ Stefan Kreiser, Skripte der Vorlesung Software Engineering f.d. Automatisierungstechnik: „Projektmanagement, Vorgehensmodelle“, ILIAS</li> <li>▪ Stanley E.Portny, „Projektmanagement für Dummies“, Wiley</li> <li>▪ Marcus Heidbrink, „Das Projektteam“, Haufe</li> <li>▪ Video Tutorial für SCRUM: <a href="http://www.video2brain.com/de/videotraining/agile-softwareentwicklung-mit-scrum">http://www.video2brain.com/de/videotraining/agile-softwareentwicklung-mit-scrum</a></li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ PLET in Master Elektrotechnik 2020</li> <li>▪ PLET in Master Elektrotechnik und Informationstechnik 2024</li> <li>▪ PLET in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:54
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.27 PLSYP - Projektleitung Systementwicklungs-Projekt

<b>Modulkürzel</b>	PLSYP_MaTIN2024
<b>Modulbezeichnung</b>	Projektleitung Systementwicklungs-Projekt
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. René Wörzberger/Professor Fakultät IME
<b>Dozierende*r</b>	Prof. Dr. René Wörzberger/Professor Fakultät IME
<b>Learning Outcome(s)</b>	
<ul style="list-style-type: none"> <li>▪ Studierende leiten ein Team von Systementwicklungs-Projekt-Teilnehmenden (SYP) an</li> <li>▪ indem sie             <ul style="list-style-type: none"> <li>▪ ein interessantes und geeignetes Entwicklungs-Thema ausarbeiten,</li> <li>▪ die Fortschritte des Teams über den gesamten Entwicklungsprozess überwachen,</li> <li>▪ regelmäßige Status-Meetings mit den Teammitgliedern abhalten</li> <li>▪ der Gesamtprojektleitung regelmäßig in weiteren Status-Meetings berichten</li> </ul> </li> <li>▪ um (später) in einem (beruflichen) Umfeld Verantwortung für die (technische) Leitung eines Entwicklungsprojekts übernehmen zu können.</li> </ul>	
<b>Modulinhalte</b>	
<b>Praktikum</b>	
SW-Erstellung im Team	
angemessene Kommunikation mit einem Kunden	
Bearbeitung des gesamten Software-Lebenszyklus	
eigenes Projekt zeitlich und inhaltlich strukturieren und organisieren	
Präsentationen erstellen und halten	
eigenen Zeitplan für das Projekt erstellen	
System-Anforderungen vom Kunden ermitteln und dokumentieren	
System gemäß Anforderungen im Team spezifizieren und modellieren	
System gemäß funktionaler Spezifikation und Qualitätsanforderungen im Team entwerfen	
System gemäß Entwurf im Team implementieren	
technische Deatails des realisierten Systems aussagekräftig darstellen	
implementiertes System auf Korrektheit prüfen	
Benutzung des realisierten Systems dokumentieren	
Verteidigung der eigenen Lösungen	
<b>Lehr- und Lernmethoden</b>	Praktikum
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	12 Stunden $\cong$ 1 SWS

<b>Selbststudium</b>	138 Stunden
<b>Empfohlene Voraussetzungen</b>	<p>Gute Kenntnisse in der Programmierung und im Software-Engineering.</p> <p>Die Kenntnisse können durch Nachweis der erfolgreich absolvierten Veranstaltungen</p> <ul style="list-style-type: none"><li>* Praktische Informatik 1 und 2 sowie des Software-Engineering-Praktikums oder</li><li>* Software-Praktikum oder</li><li>* eine eingangs durchgeführte unbenotete Leistungsprüfung nachgewiesen werden.</li></ul>
<b>Zwingende Voraussetzungen</b>	Praktikum erfordert Anwesenheit im Umfang von: 5 Termine
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"><li>▪ keine</li></ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"><li>▪ SYP in Bachelor Technische Informatik 2020</li><li>▪ SYP in Bachelor Informatik und Systems-Engineering 2024</li><li>▪ PLSYP in Master Technische Informatik 2020</li></ul>
<b>Perma-Links zur Organisation</b>	<a href="#">llu-Kurs</a>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	2.5.2025, 10:27:02
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

## 6.28 QC - Quantum Computing

<b>Modulkürzel</b>	QC_MaTIN2024
<b>Modulbezeichnung</b>	Quantum Computing
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Heiko Knospe/Professor Fakultät IME
<b>Dozierende*r</b>	<ul style="list-style-type: none"> <li>▪ Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME</li> <li>▪ Prof. Dr. Heiko Knospe/Professor Fakultät IME</li> </ul>

### 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).

### Modulinhalte

#### Vorlesung / Übungen

- 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.

<b>Lehr- und Lernmethoden</b>	Vorlesung / Übungen
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\pm$ 3 SWS

**Selbststudium** 116 Stunden

**Empfohlene  
Voraussetzungen**

**Zwingende  
Voraussetzungen**

**Empfohlene Literatur**

- 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.

**Verwendung des  
Moduls in  
weiteren Studiengängen** QC in Master Communication Systems and Networks 2024

**Besonderheiten und  
Hinweise**

**Letzte Aktualisierung** 16.3.2025, 17:36:54

**Web-Modulhandbuch-  
Editor-Links** Modul Lehrveranstaltung

## 6.29 QEKS - Qualitätsgesteuerter Entwurf komplexer Softwaresysteme

<b>Modulkürzel</b>	QEKS_MaTIN2024
<b>Modulbezeichnung</b>	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch und englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Stefan Kreiser/Professor Fakultät IME
<b>Dozierende*r</b>	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.

---

**Modulinhalte****Vorlesung / Übungen**

- Begriffe
  - Wert einer technischen Software
  - verteiltes Softwaresystem, Nebenläufigkeit
  - Softwarequalität, Dienstgüte, Refactoring
  - Komplexität (algorithmische, strukturelle), Emergenz
  - Wiederverwendung (Re-Use), Symmetrie und Symmetrioperationen, Abstraktion, Invarianten
- Methodische Ansätze zur qualitätsgesteuerten Wiederverwendung
  - Varianten für White Box Reuse
  - Black Box Reuse
  - Grey Box Reuse (Wiederverwendungshierarchie)
  - Re-Use in automatisierungstechnischen Softwaresystemen
    - Determinismus
    - Vorteile und Herausforderungen
  - angepasste Vorgehensmodelle und Personalstrukturen
    - vorhersagbare Zielerreichung in Entwicklungsprojekten (Produktqualität, Kosten, Zeit)
    - arbeitsteilige Entwicklung, Wartung und Pflege von Softwaresystemen
- Muster (Pattern)
  - Musterbeschreibung mit UML
  - grundlegende Architekturmuster
    - Erzeugungsmuster
    - Strukturmuster
    - Verhaltensmuster
    - klassenbasierte (statische) vs. objektbasierte (dynamische) Muster
  - grundlegende Muster für nebenläufige und vernetzte Echtzeitsysteme
    - Muster zur Kapselung und zur rollenbasierten Erweiterung von Layerarchitekturen
    - Muster für Nebenläufigkeitsstrukturen zur Durchsatzoptimierung und Latenzzeitminimierung
    - Muster zur verteilten Ereignisprozessierung
    - Muster zur Prozesssynchronisation
  - Aufbau und Nutzung von Musterkatalogen, Mustersprachen
  - musterbasierter Entwurf komplexer Softwaresysteme
- Komponenten und Frameworks
  - Designprinzipien
  - Schnittstellenarchitektur
  - aktive und passive Systemelemente
  - Entwurf, Programmierung und Test
    - Qualität
    - Konfiguration und Nutzung
- Middlewaresysteme in Architekturen technischer Softwaresysteme
  - ORB-Architekturen am Beispiel CORBA und TAO
  - integrierte Systemplattformen am Beispiel MS .NET
- Multiagentensysteme (MAS)
  - Architekturmodelle für Agenten
  - Kollaboration zwischen Agenten
  - Agentensprachen
  - Einsatzabwägung
- Muster zur Gestaltung komplexer Softwaresysteme einsetzen
  - Verwendungszweck, Einsatzgrenzen, invariante und parametrierbare Anteile von Mustern aus Literaturquellen in englischer und deutscher Sprache ableiten und diskutieren
  - Implementierungsskelette von Mustern nachvollziehen und auf Aufgabenstellungen mit eingeschränktem inhaltlichen Fokus transferieren
    - Vorteile objektorientierter Programmiersprachen diskutieren
    - wiederkehrende Aufgabenstellungen beim Entwurf komplexer SW-Systeme ableiten
    - Muster beispielhaft implementieren und Beispielimplementierungen prüfen
  - Muster sinnvoll kombinieren, um wiederkehrende Aufgabenstellungen mit verbreitertem inhaltlichen Fokus zu lösen
    - UML2-Notationen nutzen
    - Professionelles UML2-Entwurfswerkzeug für Round-Trip-Engineering nutzen

- Integration anhand der Beispielimplementierungen der zu kombinierenden Muster durchführen
- Integrationstest durchführen, Lösung bewerten und optimieren
- Black-Box-Komponenten musterbasiert konstruieren
- Komponentenbasierte Softwarearchitekturen analysieren
  - sinnvolle Anwendungsbereiche aus den Architekturvorgaben ableiten
  - Vorgehen zur Konstruktion von Anwendungen diskutieren (Anwendungsebene erkennen)
  - aktive und passive Systemelemente erkennen und Laufzeitverhalten ableiten
  - abstrakte Umgebungsschnittstellen zur Vernetzung, Konfiguration und Aktivierung von Komponenten erkennen
  - abstrakte Anwendungsschnittstellen zum Datenaustausch erkennen
  - Systemerweiterungspunkte finden (funktionale und strukturelle Parametrierungsebene erkennen)
- Verteilungsarchitekturen analysieren
  - Essenzielle Systemdienste erkennen, beschreiben, einordnen und begründen
  - strukturgebenden Architekturartefakten sinnvolle Lösungsmuster zuordnen
  - sinnvolle Anwendungsbereiche aus den Architekturvorgaben ableiten
  - Vorgehen zur Konstruktion von Anwendungen diskutieren (Anwendungsebene erkennen)
  - Eigenschaften und Einsatzgrenzen von Kommunikationsprotokollen diskutieren
  - vorgesehene Systemerweiterungspunkte finden
- Multiagentensysteme mit konventionellen Verteilungsarchitekturen vergleichen
  - Agent vs. Komponente
  - Architekturmodelle
  - Aktivierungsmechanismen
  - Verteilungsmechanismen
  - Kommunikationsprotokolle und Kollaborationsmechanismen
  - Einsatzgebiete und Einsatzgrenzen

### Seminar

anspruchsvolle Seminarthemen können z. B. aus den folgenden oder fachlich angrenzenden Themengebieten definiert werden: - wiederverwendbare Artefakte zum Aufbau der Architektur verteilter Softwaresysteme, - professionelle Verteilungsarchitekturen, - Multiagentensysteme, - besondere betriebswirtschaftliche, haftungsrechtliche und ethische Anforderungen bei Softwaresystemen mit (verteilter) künstlicher Intelligenz und deren Auswirkungen auf die Gestaltung von Softwarearchitekturen

eigene Arbeitsergebnisse und Arbeitsergebnisse des Teams schriftlich und mündlich kompakt und zielgruppengerecht präsentieren

### Projekt

- Softwareartefakt einer Verteilungsarchitektur für komplexe Softwaresysteme entwickeln
  - Projektierung in verteilten Teams mit agilem Vorgehensmodell durchführen
  - umfangreiche Systemanalyse hinsichtlich der Rolle des Artefakts in der Verteilungsarchitektur durchführen
  - Anforderungen an das Softwareartefakt ermitteln
  - Softwareartefakt basierend auf den Anforderungen spezifizieren und modellieren
    - Designprinzipien und Muster zum Erreichen definierter Qualitätsziele auswählen und begründen
    - Schnittstellen-, Verhaltens- und Strukturmodelle musterbasiert in UML2-Notationen iterativ herleiten
    - Professionelles UML2-Entwurfswerkzeug zielgerichtet einsetzen
    - Modelle verifizieren und bewerten, Modellfehler korrigieren und Modelle optimieren
  - Softwareartefakt in C++ programmieren
  - sinnvolle Prüfscenarien definieren und Softwareartefakt verifizieren
  - Qualität des Softwareartefakts bewerten
- Arbeitsergebnisse des Teams kompakt und zielgruppengerecht präsentieren

**Lehr- und Lernmethoden**

- Vorlesung / Übungen
- Seminar
- Projekt

**Prüfungsformen mit Gewichtung**

- begleitend: Projektarbeit [unbenotet] und
- abschließend: mündliche Prüfung [100%]

**Workload** 150 Stunden

**Präsenzzeit** 57 Stunden  $\pm$  5 SWS

<b>Selbststudium</b>	93 Stunden
<b>Empfohlene Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Modul PL: oder aus einem (naturwissenschaftlich-technischen) Bachelorstudium: - grundlegende Kenntnisse in (agilem) Projektmanagement</li> <li>▪ - Programmierkenntnisse in einer objektorientierten Programmiersprache, bevorzugt C++</li> <li>- Kenntnisse in Software-Modellierung mit Hilfe der Unified Modeling Language (UML) oder anderen (formalen) Sprachen, die das Modellieren von Schnittstellen, Verhalten und Strukturen unterstützen</li> <li>- grundlegende Kenntnisse in (agilem) Projektmanagement</li> <li>- grundlegende Softwarearchitekturmodelle</li> <li>- Kommunikationsmodelle in Softwaresystemen (OSI, TCPIP, Messaging)</li> </ul>
<b>Zwingende Voraussetzungen</b>	<ul style="list-style-type: none"> <li>▪ Projekt erfordert Anwesenheit im Umfang von: 3 Termine</li> <li>▪ Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Projekt</li> </ul>
<b>Empfohlene Literatur</b>	<ul style="list-style-type: none"> <li>▪ D. Schmidt et.al.: Pattern-Oriented Software Architecture. Patterns for Concurrent and Networked Objects (Wiley)</li> <li>▪ Gamma et.al.: Design Patterns, (Addison-Wesley)</li> <li>▪ Martin Fowler: Refactoring, Engl. ed. (Addison-Wesley Professional)</li> <li>▪ U. Hammerschall: Verteilte Systeme und Anwendungen (Pearson Studium)</li> <li>▪ Andreas Andresen: Komponentenbasierte Softwareentwicklung m. MDA, UML2, XML (Hanser Verlag)</li> <li>▪ T. Ritter et. al.: CORBA Komponenten. Effektives Software-Design u. Progr. (Springer)</li> <li>▪ Bernd Oestereich: Analyse und Design mit UML 2.5 (Oldenbourg)</li> <li>▪ OMG Unified Modeling Language Spec., <a href="http://www.omg.org/um">www.omg.org/um</a></li> <li>▪ I. Sommerville: Software Engineering (Addison-Wesley / Pearson Studium)</li> <li>▪ K. Beck: eXtreme Programming (Addison-Wesley Professional)</li> <li>▪ Ken Schwaber: Agiles Projektmanagement mit Scrum (Microsoft Press)</li> </ul>
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ QEKS in Master Elektrotechnik 2020</li> <li>▪ QEKS in Master Elektrotechnik und Informationstechnik 2024</li> <li>▪ QEKS in Master Technische Informatik 2020</li> </ul>
<b>Besonderheiten und Hinweise</b>	
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:54
<b>Web-Modulhandbuch-Editor-Links</b>	Modul Lehrveranstaltung

### 6.30 RFSD - RF System Design

<b>Modulkürzel</b>	RFSD_MaTIN2024
<b>Modulbezeichnung</b>	RF System Design
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Rainer Kronberger/Professor Fakultät IME
<b>Dozierende*r</b>	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)

#### Modulinhalte

##### Vorlesung / Übungen

Hochfrequenzsysteme und Anwendungen  
 Rauschen in Hochfrequenzsystemen und Baugruppen  
 Charakterisierung, Berechnung und Anwendung  
 Lineares und nichtlineares Schaltungsverhalten  
 Nichtlinearität zur Mischung, nichtlineares Verhalten von Verstärkern  
 Hochfrequenzsystemkomponenten  
 Sender, Empfänger, Oszillatoren

##### Praktikum

Die Studierenden lernen die Funktions- und Wirkungsweise von hochfrequenten Schaltungen und Baugruppe kennen und lernen, wie die hochfrequente System e aufgebaut und entwickelt werden.

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung / Übungen</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Übungspraktikum [unbenotet] und</li> <li>▪ abschließend: (elektronische) Klausur oder mündliche Prüfung [100%]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\triangleq$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Hochfrequenztechnik und Mikrowellentechnik

- Zwingende Voraussetzungen**
- Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Vorlesung / Übungen
  - Praktikum erfordert Anwesenheit im Umfang von: 3 Labortermine und 1 Präsentationstermin
  - Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum

- Empfohlene Literatur**
- Kraus & Carver Eletromagnetics, McGraw Hilll, 2006.
  - Michale Steer, Microwave and RF Design

- Verwendung des Moduls in weiteren Studiengängen**
- RFSD in Master Communication Systems and Networks 2020
  - RFSD in Master Communication Systems and Networks 2024
  - RFSD in Master Medientechnologie 2024
  - RFSD in Master Medientechnologie 2024

**Besonderheiten und Hinweise**

**Letzte Aktualisierung** 2.5.2025, 10:27:02

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

### 6.31 THI - Theoretische Informatik

<b>Modulkürzel</b>	THI_MaTIN2024
<b>Modulbezeichnung</b>	Theoretische Informatik
<b>Art des Moduls</b>	Pflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr. Hubert Randerath/Professor Fakultät IME
<b>Dozierende*r</b>	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.

**Modulinhalte**

**Vorlesung / Übungen**

Die Bestimmung der Komplexität eines Algorithmus kann z.B. durch Analyse der Eingabeinstanz und des algorithmischen Kerns und Anwenden der O-Notation vorgenommen werden. Die Hartnäckigkeit eines algorithmischen Problems kann z.B. durch Anwenden einer geeigneten Reduktion auf ein etabliertes hartnäckiges Problem, wie beispielsweise dem aussagenlogischen Erfüllbarkeitsproblem, erreicht werden.

<b>Lehr- und Lernmethoden</b>	Vorlesung / Übungen
<b>Prüfungsformen mit Gewichtung</b>	▪ abschließend: mündliche Prüfung [100%]
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	34 Stunden $\cong$ 3 SWS
<b>Selbststudium</b>	116 Stunden
<b>Empfohlene Voraussetzungen</b>	Grundlagen der Automatentheorie und der Formalen Sprachen
<b>Zwingende Voraussetzungen</b>	
<b>Empfohlene Literatur</b>	
<b>Verwendung des Moduls in weiteren Studiengängen</b>	THI in Master Technische Informatik 2020

---

**Besonderheiten und  
Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:54

---

**Web-Modulhandbuch-  
Editor-Links** Modul Lehrveranstaltung

### 6.32 VAE - Virtual Acoustic Environments

<b>Modulkürzel</b>	VAE_MaTIN2024
<b>Modulbezeichnung</b>	Virtual Acoustic Environments
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	englisch
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Sommersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME
<b>Dozierende*r</b>	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

**Modulinhalte**

**Vorlesung**

Die grundlegenden Konzepte zur Erzeugung kophörerbasierter oder lautsprecherbasierter VR-Systeme werden vorgestellt.

**Projekt**

Es soll vertieftes Wissen in einem der Bereiche / Aspekte von virtuellen akustischen Umgebungen erarbeitet, angewendet und präsentiert werden

**Praktikum**

<b>Lehr- und Lernmethoden</b>	<ul style="list-style-type: none"> <li>▪ Vorlesung</li> <li>▪ Projekt</li> <li>▪ Praktikum</li> </ul>
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ begleitend: Projektarbeit [100%] und</li> <li>▪ begleitend: Übungspraktikum [unbenotet]</li> </ul>
<b>Workload</b>	150 Stunden
<b>Präsenzzeit</b>	45 Stunden $\approx$ 4 SWS
<b>Selbststudium</b>	105 Stunden
<b>Empfohlene Voraussetzungen</b>	Gundlagen Akustik, Signalverarbeitung
<b>Zwingende Voraussetzungen</b>	Teilnahme an abschließender Prüfung nur nach erfolgreicher Teilnahme an Praktikum

- Empfohlene Literatur**
- 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"

- Verwendung des Moduls in weiteren Studiengängen**
- 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

---

**Besonderheiten und Hinweise**

**Letzte Aktualisierung** 16.3.2025, 17:36:54

**Web-Modulhandbuch-Editor-Links** Modul Lehrveranstaltung

### 6.33 VER - Virtuelle und erweiterte Realität

<b>Modulkürzel</b>	VER_MaTIN2024
<b>Modulbezeichnung</b>	Virtuelle und erweiterte Realität
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	5
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Wintersemester
<b>Modul-Verantwortliche*r</b>	Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME
<b>Dozierende*r</b>	<ul style="list-style-type: none"> <li>▪ Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME</li> <li>▪ Prof. Dr. Stefan Grünvogel/Professor Fakultät IME</li> </ul>

#### 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).

**Modulinhalte**

**Vorlesung**

Datenstrukturen und Algorithmen für VR/AR-Anwendungen  
 Räumliche Datenstrukturen  
 Interaktion in VR/AR  
 Ein- und Ausgabegeräte  
 Stereoskopisches Rendering  
 Tracking  
 Echtzeitrendering für VR/AR-Anwendungen  
 Animation von Charakteren  
 Animation von deformierbaren Objekten  
 Kollisionserkennung und -behandlung

**Praktikum**

- Virtuelle Umgebungen und Augmented Reality-Anwendungen konzipieren, aufbauen und bewerten
- Interaktions und Navigationsverfahren erstellen
- Basistechnologien der virtuellen und erweiterten Reality weiterentwickeln
- Werkzeuge und Methoden zur Realisierung von VR/AR-Anwendungen verwenden
- Algorithmische und mathematische Grundlagen von VR/AR anwenden
- textuelle Aufgabenstellungen erfassen und verstehen
- Testen und debuggen der eigenen Anwendung

**Seminar**

Algorithmische und mathematische Grundlagen anwenden  
 Interaktions- und Navigationsverfahren prüfen  
 Selbstständig wissenschaftliche Literatur beschaffen und zusammenfassen  
 Neue Konzepte der virtuellen und erweiterten Realität darstellen und diskutieren

**Lehr- und Lernmethoden**

- Vorlesung
- Praktikum
- Seminar

**Prüfungsformen mit Gewichtung**

- begleitend: Übungspraktikum [unbenotet] und
- abschließend: mündliche Prüfung oder Hausarbeit oder Projektarbeit [100%]

**Workload** 150 Stunden

**Präsenzzeit** 45 Stunden  $\cong$  4 SWS

**Selbststudium** 105 Stunden

**Empfohlene Voraussetzungen**  
 Computergrafik  
 Computeranimation

**Zwingende Voraussetzungen** Praktikum erfordert Anwesenheit im Umfang von: 2 Termine

**Empfohlene Literatur**

- 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

**Verwendung des Moduls in weiteren Studiengängen**

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

---

**Besonderheiten und  
Hinweise**

---

**Letzte Aktualisierung** 16.3.2025, 17:36:54

---

**Web-Modulhandbuch-  
Editor-Links** Modul Lehrveranstaltung

### 6.34 XGA - Gremienarbeit

<b>Modulkürzel</b>	XGA_MaTIN2024
<b>Modulbezeichnung</b>	Gremienarbeit
<b>Art des Moduls</b>	Wahlpflichtmodul
<b>ECTS credits</b>	0
<b>Sprache</b>	deutsch, englisch bei Bedarf
<b>Dauer des Moduls</b>	1 Semester
<b>Empfohlenes Studiensemester</b>	1-2
<b>Häufigkeit des Angebots</b>	jedes Semester
<b>Modul-Verantwortliche*r</b>	Studiengangsleiter(in) Master Technische Informatik / Informatik und Systems-Engineering
<b>Dozierende*r</b>	
<b>Learning Outcome(s)</b>	
	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.
<b>Modulinhalte</b>	
<b>Projekt</b>	
<b>Lehr- und Lernmethoden</b>	Projekt
<b>Prüfungsformen mit Gewichtung</b>	<ul style="list-style-type: none"> <li>▪ abschließend: Fachgespräch [unbenotet]</li> </ul>
<b>Workload</b>	0 Stunden
<b>Präsenzzeit</b>	12 Stunden $\cong$ 1 SWS
<b>Selbststudium</b>	-12 Stunden
<b>Empfohlene Voraussetzungen</b>	
<b>Zwingende Voraussetzungen</b>	
<b>Empfohlene Literatur</b>	
<b>Verwendung des Moduls in weiteren Studiengängen</b>	<ul style="list-style-type: none"> <li>▪ XIB in Bachelor Elektrotechnik 2020</li> <li>▪ XGA in Bachelor Elektrotechnik und Informationstechnik 2024</li> <li>▪ XIB in Bachelor Technische Informatik 2020</li> <li>▪ XIB2 in Bachelor Technische Informatik 2020</li> <li>▪ XGA in Bachelor Informatik und Systems-Engineering 2024</li> </ul>
<b>Besonderheiten und Hinweise</b>	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.
<b>Letzte Aktualisierung</b>	16.3.2025, 17:36:54
<b>Web-Modulhandbuch- Editor-Links</b>	Modul Lehrveranstaltung



## 7. Wahlbereiche

Im Folgenden wird dargestellt, welche Module in einem bestimmten Wahlbereich gewählt werden können. Für alle Wahlbereiche gelten folgende Hinweise und Regularien:

- In welchem Semester Wahlpflichtmodule eines Wahlbereichs typischerweise belegt werden können, kann den Studienverlaufsplänen entnommen werden.
- Module werden in der Regel nur entweder im Sommer- oder Wintersemester angeboten. Das heißt, dass eine eventuell erforderliche begleitende Prüfung nur im Sommer- oder Wintersemester abgelegt werden kann. Die summarischen Prüfungen werden bei Modulen der Fakultät 07 für Medien-, Informations- und Elektrotechnik in der Regel in der Prüfungszeit nach jedem Semester angeboten.
- Ein absolviertes Modul wird für maximal einen Wahlbereich anerkannt, auch wenn es in mehreren Wahlbereichen aufgelistet ist.
- Bei manchen Modulen gibt es eine Aufnahmebegrenzung. Näheres hierzu ist in den Bekanntmachungen zu den Aufnahmebegrenzungen zu finden.
- Die Anmeldung an und die Aufnahme in fakultätsexterne Module unterliegen Fristen und anderen Bedingungen der anbietenden Fakultät oder Hochschule. Eine Aufnahme kann nicht garantiert werden. Studierende müssen sich frühzeitig bei der jeweiligen externen Lehrperson informieren, ob Sie an einem externen Modul teilnehmen dürfen und was für eine Anmeldung und Teilnahme zu beachten ist.
- Auf Antrag kann der Wahlbereich um weitere passende Module ergänzt werden. Ein solcher Antrag ist bis spätestens sechs Monate vor einer geplanten Teilnahme an einem zu ergänzenden Modul formlos an die Studiengangsleitung zu richten. Über die Annahme des Antrags befindet der Prüfungsausschuss im Benehmen mit der Studiengangsleitung und fachlich geeigneten Lehrpersonen.

### 7.1 AUS - Auslandsphase während des Studiums

Gemäß § 4 Abs. 2 der Prüfungsordnung besteht die Möglichkeit, einen Auslandsaufenthalt in das Studium zu integrieren.

Studierende, die einen Auslandsaufenthalt in ihr Studium integriert haben und dabei Studienleistungen an einer ausländischen Hochschule erbracht haben, können sich diese auf Antrag und mit Zustimmung des Prüfungsausschusses anerkennen lassen.

Die Leistungspunkte der anerkannten Studienleistungen können von den im regulären Studienverlaufplan vorgesehenen Leistungspunkten abweichen.

Näheres regeln die vom Fakultätsrat beschlossenen „Richtlinien zur Anerkennung von im Ausland an einer Hochschule erbrachten Leistungen“ in der jeweils gültigen Fassung.

Vor Antritt des Auslandsaufenthaltes ist mit dem Auslandsbeauftragten der Fakultät ein Learning Agreement abzuschließen.

## 7.2 VMT - Vertiefung Mathematik

Für dieses Wahlmodul stehen folgende Module zur Verfügung.

Aus diesem Wahlbereich müssen Module im Umfang von mindestens 5 ECTS-Kreditpunkten belegt werden.

Module der Fakultät:

<b>Modulkürzel</b>	<b>Modulbezeichnung</b>	<b>ECTS</b>
HIM	Advanced Mathematics	5
KOGA	Kombinatorische Optimierung und Graphenalgorithmen	5
KRY	Cryptography	5
MLWR	Maschinelles Lernen und wissenschaftliches Rechnen	5

Module anderer Fakultäten oder Hochschulen:

<b>Zugehörigkeit</b>	<b>Modulbezeichnung</b>	<b>ECTS</b>
TH Köln (Fakultät 10)	Multivariate Statistik	6

### 7.3 VTI - Vertiefung Technische Informatik

Aus diesem Wahlbereich müssen Module im Umfang von mindestens 15 ECTS-Kreditpunkten belegt werden.

Module der Fakultät:

Modulkürzel	Modulbezeichnung	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

Module anderer Fakultäten oder Hochschulen:

Zugehörigkeit	Modulbezeichnung	ECTS
TH Köln (Fakultät 10)	Advanced Machine Learning	6
Universität Köln	Quantum Information Theory	6

## 7.4 W - Allgemeiner Wahlbereich

Aus diesem Wahlbereich müssen Module im Umfang von mindestens 20 ECTS-Kreditpunkten belegt werden.

Dieser Wahlbereich umfasst insbesondere alle Module aus folgenden anderen Bereichen:

- Wahlbereich VTI - Vertiefung Technische Informatik
- Wahlbereich VMT - Vertiefung Mathematik

Module der Fakultät:

Modulkürzel	Modulbezeichnung	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 in Master Medientechnologie 2024	5
CI	Computational Intelligence	5
CSO	Computersimulation in der Optik in Master Elektrotechnik und Informationstechnik 2024	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 in Master Elektrotechnik und Informationstechnik 2024	5
RFSD	RF System Design	5
SIM	Simulation in der Ingenieurwissenschaft in Master Elektrotechnik und Informationstechnik 2024	5
VAE	Virtual Acoustic Environments	5
VER	Virtuelle und erweiterte Realität	5

Module anderer Fakultäten oder Hochschulen:

<b>Zugehörigkeit</b>	<b>Modulbezeichnung</b>	<b>ECTS</b>
Hochschule Bonn-Rhein-Sieg	Virtuelle Private Netze	6
Hochschule Bonn-Rhein-Sieg	Service Management in Netzen	6
TH Köln (Fakultät 10)	Data Driven Modeling	6
TH Köln (Fakultät 10)	Domain-Driven Design of Large Software Systems	6
Hochschule Bonn-Rhein-Sieg	Embedded Systems	6
TH Köln (Fakultät 10)	Data Science and Ethics	6
TH Köln (Fakultät 09)	Modellierung von Energiesystemen	5
TH Köln (Fakultät 10)	Advanced Machine Learning	6
Universität Köln	Quantum Information Theory	6
TH Köln (Fakultät 10)	Multivariate Statistik	6

### 7.5 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.

Aus diesem Wahlbereich müssen Module im Umfang von mindestens 5 ECTS-Kreditpunkten belegt werden.

Module der Fakultät:

Modulkürzel	Modulbezeichnung	ECTS
ERMK	Entrepreneurship, Gewerblicher Rechtsschutz, Market Knowledge in Master Elektrotechnik und Informationstechnik 2024	5
ETH	Ethik	5
PLET	Projektleitung	5
PLSYP	Projektleitung Systementwicklungs-Projekt	5
XGA	Gremienarbeit	0

Module anderer Fakultäten oder Hochschulen:

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



## 8. Studienschwerpunkte

Im Masterstudiengang Informatik und Systems-Engineering gibt es keine Studienrichtungen oder -schwerpunkte

## 9. Prüfungsformen

Im Folgenden werden die in den Modulbeschreibungen referenzierten Prüfungsformen näher erläutert. Die Erläuterungen stammen aus der Prüfungsordnung, §19ff. Bei Abweichungen gilt der Text der Prüfungsordnung.

### **(elektronische) Klausur**

Schriftliche, in Papierform oder digital unterstützt abgelegte Prüfung. Genauerer regelt §19 der Prüfungsordnung.

### **Mündliche Prüfung**

Mündlich abzulegende Prüfung. Genauerer regelt §21 der Prüfungsordnung.

### **Mündlicher Beitrag**

Siehe §22, Abs. 5 der Prüfungsordnung: Ein mündlicher Beitrag (z. B. Referat, Präsentation, Verhandlung, Moderation) dient der Feststellung, ob die Studierenden befähigt sind, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe nach wissenschaftlichen und fachpraktischen Methoden selbstständig zu bearbeiten und mittels verbaler Kommunikation fachlich angemessen darzustellen. Dies beinhaltet auch, Fragen des Auditoriums zur mündlichen Darstellung zu beantworten. Die Dauer des mündlichen Beitrags wird von der Prüferin beziehungsweise dem Prüfer zu Beginn des Semesters festgelegt. Die für die Benotung des mündlichen Beitrags maßgeblichen Tatsachen sind in einem Protokoll festzuhalten, zur Dokumentation sollen die Studierenden ebenfalls die schriftlichen Unterlagen zum mündlichen Beitrag einreichen. Die Note ist den Studierenden spätestens eine Woche nach dem mündlichen Beitrag bekanntzugeben.

### **Fachgespräch**

Siehe §22, Abs. 8 der Prüfungsordnung: Ein Fachgespräch dient der Feststellung der Fachkompetenz, des Verständnisses komplexer fachlicher Zusammenhänge und der Fähigkeit zur analytischen Problemlösung. Im Fachgespräch haben die Studierenden und die Prüfenden in etwa gleiche Redeanteile, um einen diskursiven fachlichen Austausch zu ermöglichen. Semesterbegleitend oder summarisch werden ein oder mehrere Gespräche mit einer Prüferin oder einem Prüfer geführt. Dabei sollen die Studierenden praxisbezogene technische Aufgaben, Problemstellungen oder Projektvorhaben aus dem Studiengang vorstellen und erläutern sowie die relevanten fachlichen Hintergründe, theoretischen Konzepte und methodischen Ansätze zur Bearbeitung der Aufgaben darlegen. Mögliche Lösungsansätze, Vorgehensweisen und Überlegungen zur Problemlösung sind zu diskutieren und zu begründen. Die für die Benotung des Fachgesprächs maßgeblichen Tatsachen sind in einem Protokoll festzuhalten.

### **Projektarbeit**

Siehe §22, Abs. 6 der Prüfungsordnung: Die Projektarbeit ist eine Prüfungsleistung, die in der selbstständigen Bearbeitung einer spezifischen Fragestellung unter Anleitung mit wissenschaftlicher Methodik und einer Dokumentation der Ergebnisse besteht. Bewertungsrelevant sind neben der Qualität der Antwort auf die Fragestellung auch die organisatorische und kommunikative Qualität der Durchführung, wie z.B. Slides, Präsentationen, Meilensteine, Projektpläne, Meetingprotokolle usw.

### **Praktikumsbericht**

Siehe §22, Abs. 10 der Prüfungsordnung: Ein Praktikumsbericht (z. B. Versuchsprotokoll) dient der Feststellung, ob die Studierenden befähigt sind, innerhalb einer vorgegebenen Frist eine laborpraktische Aufgabe selbstständig sowohl praktisch zu bearbeiten als auch Bearbeitungsprozess und Ergebnis schriftlich zu dokumentieren, zu bewerten und zu reflektieren. Vor der eigentlichen Versuchsdurchführung können vorbereitende Hausarbeiten erforderlich sein. Während oder nach der Versuchsdurchführung können Fachgespräche stattfinden. Praktikumsberichte können auch in Form einer Gruppenarbeit zur Prüfung zugelassen werden. Die Bewertung des Praktikumsberichts ist den Studierenden spätestens sechs Wochen nach Abgabe des Berichts bekanntzugeben.

### **Übungspraktikum**

Siehe §22, Abs. 11 der Prüfungsordnung: Mit der Prüfungsform "Übungspraktikum" wird die fachliche Kompetenzen bei der Anwendung der in der Vorlesung erlernten Theorien und Konzepte sowie praktische Fertigkeiten geprüft, beispielsweise der Umgang mit Entwicklungswerkzeugen und Technologien. Dazu werden semesterbegleitend mehrere Aufgaben gestellt, die entweder alleine oder in Gruppenarbeit, vor Ort oder auch als Hausarbeit bis zu einem jeweils vorgegebenen Termin zu lösen sind. Die Lösungen der Aufgaben sind durch die Studierenden in (digitaler) schriftlicher Form einzureichen. Die genauen Kriterien zum Bestehen der Prüfung wird zu Beginn der entsprechenden Lehrveranstaltung bekannt gegeben.

### **Übungspraktikum unter Klausurbedingungen**

Siehe §22, Abs. 11, Satz 5 der Prüfungsordnung: Ein "Übungspraktikum unter Klausurbedingungen" ist ein Übungspraktikum, bei dem die Aufgaben im zeitlichen Rahmen und den Eigenständigkeitsbedingungen einer Klausur zu bearbeiten sind.

### **Hausarbeit**

Siehe §22, Abs. 3 der Prüfungsordnung: Eine Hausarbeit (z.B. Fallstudie, Recherche) dient der Feststellung, ob die Studierenden befähigt sind, innerhalb einer vorgegebenen Frist eine Fachaufgabe nach wissenschaftlichen und fachpraktischen Methoden selbstständig in schriftlicher oder elektronischer Form zu bearbeiten. Das Thema und der Umfang (z. B. Seitenzahl des Textteils) der Hausarbeit werden von der Prüferin beziehungsweise dem Prüfer zu Beginn des Semesters festgelegt. Eine Eigenständigkeitserklärung muss vom Prüfling unterzeichnet und abgegeben werden. Zusätzlich können Fachgespräche geführt werden.

### **Lernportfolio**

Ein Lernportfolio dokumentiert den studentischen Kompetenzentwicklungsprozess anhand von Präsentationen, Essays, Ausschnitten aus Praktikumsberichten, Inhaltsverzeichnissen von Hausarbeiten, Mitschriften, To-Do-Listen, Forschungsberichten und anderen Leistungsdarstellungen und Lernproduktionen, zusammengefasst als sogenannte „Artefakte“. Nur in Verbindung mit der studentischen Reflexion (schriftlich, mündlich oder auch in einem Video) der Verwendung dieser Artefakte für das Erreichen des zuvor durch die Prüferin oder den Prüfer transparent gemachten Lernziels wird das Lernportfolio zum Prüfungsgegenstand. Während der Erstellung des Lernportfolios wird im Semesterverlauf Feedback auf Entwicklungsschritte und/oder Artefakte gegeben. Als Prüfungsleistung wird eine nach dem Feedback überarbeitete Form des Lernportfolios - in handschriftlicher oder elektronischer Form - eingereicht.

### **Schriftliche Prüfung im Antwortwahlverfahren**

Siehe §20 der Prüfungsordnung.

### **Zugangskolloquium**

Siehe §22, Abs. 12 der Prüfungsordnung: Ein Zugangskolloquium dient der Feststellung, ob die Studierenden die versuchsspezifischen Voraussetzungen erfüllen, eine definierte laborpraktische Aufgabe nach wissenschaftlichen und fachpraktischen Methoden selbstständig und sicher bearbeiten zu können.

### **Testat / Zwischentestat**

Siehe §22, Abs. 7 der Prüfungsordnung: Mit einem Testat/Zwischentestat wird bescheinigt, dass die oder der Studierende eine Studienarbeit (z.B. Entwurf) im geforderten Umfang erstellt hat. Der zu erbringende Leistungsumfang sowie die geforderten Inhalte und Anforderungen ergeben sich aus der jeweiligen Modulbeschreibung im Modulhandbuch sowie aus der Aufgabenstellung.

### **Open-Book-Ausarbeitung**

Die Open-Book-Ausarbeitung oder -Arbeit (OBA) ist eine Kurz-Hausarbeit und damit eine unbeaufsichtigte schriftliche oder elektronische Prüfung. Sie zeichnet sich dadurch aus, dass gemäß Hilfsmittelerklärung der Prüferin bzw. des Prüfers in der Regel alle Hilfsmittel zugelassen sind. Auf die Sicherung guter wissenschaftlicher Praxis durch ordnungsgemäßes Zitieren etc. und das Erfordernis der Eigenständigkeit der Erbringung jedweder Prüfungsleistung wird besonders hingewiesen.

### **Abschlussarbeit**

Bachelor- oder Masterarbeit im Sinne der Prüfungsordnung §25ff.: Die Masterarbeit ist eine schriftliche Hausarbeit. Sie soll zeigen, dass die oder der Studierende befähigt ist, innerhalb einer vorgegebenen Frist ein Thema aus ihrem oder seinem Fachgebiet sowohl in seinen fachlichen Einzelheiten als auch in den fachübergreifenden Zusammenhängen nach wissenschaftlichen und fachpraktischen Methoden selbstständig zu bearbeiten. Die interdisziplinäre Zusammenarbeit kann auch bei der Abschlussarbeit berücksichtigt werden.

### **Kolloquium**

Kolloquium zur Bachelor- oder Masterarbeit im Sinne der Prüfungsordnung §29: Das Kolloquium dient der Feststellung, ob die Studentin oder der Student befähigt ist, die Ergebnisse der Masterarbeit, ihre fachlichen und methodischen Grundlagen, fachübergreifende Zusammenhänge und außerfachlichen Bezüge mündlich darzustellen, selbstständig zu begründen und ihre Bedeutung für die Praxis einzuschätzen.

## 10. Profil-Modulmatrix

Im Folgenden wird dargestellt, inwieweit die Module des Studiengangs die Kompetenzen und Handlungsfelder des Studiengangs sowie hochschulweite Studiengangskriterien stützen bzw. ausbilden.

Kürzel	Modulbezeichnung	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 Kreativitä...	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	●			●	●	●	●	●	●				●	●		●	●				

Kürzel	Modulbezeichnung	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	●	●		●	●	●		●	●	●		●					●				
PLET	Projektleitung	●	●	●	●			●	●		●		●		●	●	●		●		●	
PLSYP	Projektleitung Systementwicklungs-Projekt			●			●		●						●		●		●			
QC	Quantum Computing	●	●																	●		
QEKS	Qualitätsgesteuerter Entwurf komplexer Softwaresysteme	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
RFSD	RF System Design		●		●	●	●	●		●				●				●		●	●	
THI	Theoretische Informatik	●	●		●	●	●		●	●	●						●	●				
VAE	Virtual Acoustic Environments	●	●		●	●	●	●	●						●		●				●	●
VER	Virtuelle und erweiterte Realität	●	●		●	●	●	●	●	●	●	●	●	●		●	●	●				

Kürzel	Modulbezeichnung
	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
XGA	Gremienarbeit

## 11. Versionsverlauf

In untenstehender Tabelle sind die verschiedenen Versionen des Lehrangebots aufgeführt. Die Versionen sind umgekehrt chronologisch sortiert mit der aktuell gültigen Version in der ersten Zeile. Die einzelnen Versionen können über den Link in der rechten Spalte aufgerufen werden.

Version	Datum	Änderungen	Link
1.1	2024-11-29-12-00-00	1. Version zur Abstimmung im Fakultätsrat	<a href="#">Link</a>
1.0	2024-11-29-12-00-00	1. Begutachtete Version für Reakkreditierung 2024 2. Neues Layout für sämtliche Modulhandbücher	<a href="#">Link</a>

Impressum

Datenschutzhinweis

Haftungshinweis

Bei Fehlern, bitte Mitteilung an  
die  
[modulhandbuchredaktion@f07.th-koeln.de](mailto:modulhandbuchredaktion@f07.th-koeln.de)