

Modulhandbuch

des konsekutiven Master-Studiengangs

Allgemeine Informatik

Master of Science (M.Sc.)

Fb2: Informatik und Ingenieurwissenschaften

– Computer Science and Engineering

Inhaltsverzeichnis

| | | |
|----------|---|----|
| 1 | Qualifikationsziele | 3 |
| 2 | Beispielhafter Studienverlauf | 5 |
| 3 | ECTS-/Workload-Übersicht | 6 |
| 4 | Modulbeschreibungen | 8 |
| Modul 1 | Grundlagen adaptiver Wissenssysteme | 8 |
| Modul 2 | Maschinelles Lernen | 11 |
| Modul 3 | Mensch-Maschine Interaktion | 14 |
| Modul 4 | Spracherkennung und -synthese | 17 |
| Modul 5 | Bilderkennung | 20 |
| Modul 6 | Robotics | 23 |
| Modul 7 | Ontologiesprachen und Semantic Web | 27 |
| Modul 8 | Künstliche Intelligenz | 31 |
| Modul 9 | Learning from Data | 34 |
| Modul 10 | Internet of Things | 38 |
| Modul 11 | Advanced IT-Security | 40 |
| Modul 12 | Smart Sensor Network Systems | 43 |
| Modul 13 | Introductory Data Analysis | 45 |
| Modul 14 | Data Mining Methods | 49 |
| Modul 15 | Simulation Methods | 53 |
| Modul 16 | Advanced Data Structures and Algorithms | 56 |
| Modul 17 | Pattern Oriented Software Architecture | 60 |
| Modul 18 | Cloud Computing | 63 |
| Modul 19 | Advanced Distributed Systems | 66 |
| Modul 20 | Advanced Formal Modelling | 69 |
| Modul 21 | Formal Specification and Verification | 72 |
| Modul 22 | Mathematics Update | 75 |
| Modul 23 | Advanced Testing Methods | 78 |
| Modul 24 | Enterprise Architecture Engineering | 81 |
| Modul 25 | Current Topics in Computer Science | 85 |
| Modul 26 | Projekt Intelligente Systeme | 87 |
| Modul 27 | Projekt Digitalisierung | 89 |
| Modul 28 | Projekt Softwaretechnik | 91 |
| Modul 29 | Masterarbeit mit Kolloquium | 93 |

1 Qualifikationsziele

Die Absolventinnen und Absolventen sind qualifiziert, um kompetent, eigenverantwortlich und selbständig anspruchsvolle und innovative Funktionen in Projektierung, Projektdurchführung, Entwicklung, Beratung, Vertrieb in Unternehmen der Wirtschaft, Industrie sowie der öffentlichen Hand auszuüben bzw. sich in der Forschung weiter zu qualifizieren.

Die Absolventinnen und Absolventen werden durch das Studienfeld „Softwaretechnik“ in die Lage versetzt, die zeitgemäße Umsetzung von verteilten Geschäfts- und Entwicklungsprozess- und IT-Strategien eigenverantwortlich voranzutreiben, sei es durch Analyse und Konzeption in Forschungs- und Entwicklungsprojekten oder durch die ganzheitliche Herangehensweise im Projektmanagementumfeld. Dabei liegt ein besonderer Fokus auf den Herausforderungen durch das automatisierte Lernen aus (großen) Datenmengen unter besondere Berücksichtigung von Sicherheitsaspekten.

Im Studienfeld „Intelligente Systeme“ erwerben die Absolventinnen und Absolventen Kenntnisse in Techniken des maschinellen Lernens, der adaptiver Wissensgewinnung und der Interaktion von Mensch und Maschine kombiniert mit Robotik, Bild- oder Sprachverarbeitung. Dies befähigt die Absolventen aktuelle Schlüsseltechnologien für intelligente interagierende Systeme zu entwickeln, die in einer natürlichen Umwelt auf intuitive Weise mit ihren Nutzern kooperieren.

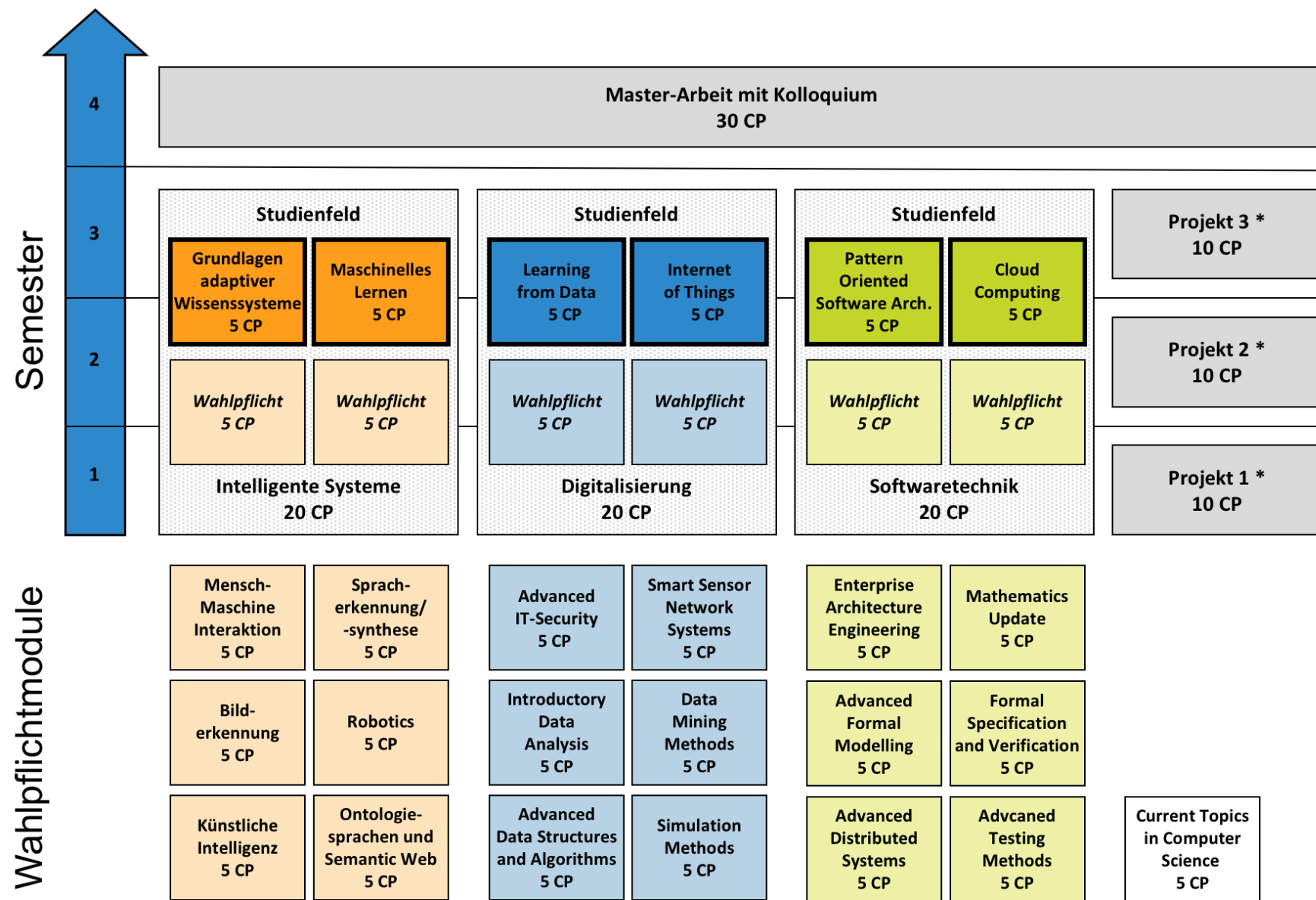
Das Studienfeld „Digitalisierung“ vermittelt die informationstheoretische Methodik und das Handwerkzeug zur Generierung (u. a. mithilfe von Sensoren) und technischen, sicheren Beherrschbarkeit der (großen) Datenmengen, die durch das Internet of Things (IoT) generiert werden. Es ermächtigt ergänzend daher dazu, die Industrie 4.0 mitzugestalten.

Aufbauend auf den mit dem Bachelorabschluss erworbenen Kenntnissen werden die Absolventinnen und Absolventen befähigt, komplexere Probleme und Aufgaben in der unternehmerischen Praxis (z. B. „Industrie 4.0“, „Internet der Dinge“, Assistenzsysteme, R&D Projekte) im Team erfolgreich zu bearbeiten. Diese Probleme und Aufgaben erfordern einen ganzheitlichen und grundlagenbasierten Analyse- und Konzeptionsansatz, für den oft noch keine standardisierten Vorgehensmodelle und/oder widerstreitende Lösungsansätze existieren.

Im Rahmen des Studiengangs werden nicht nur die nötigen methodischen und technischen Grundlagen gelehrt, sondern auch relevante soziale und Querschnittskompetenzen, unter anderem:

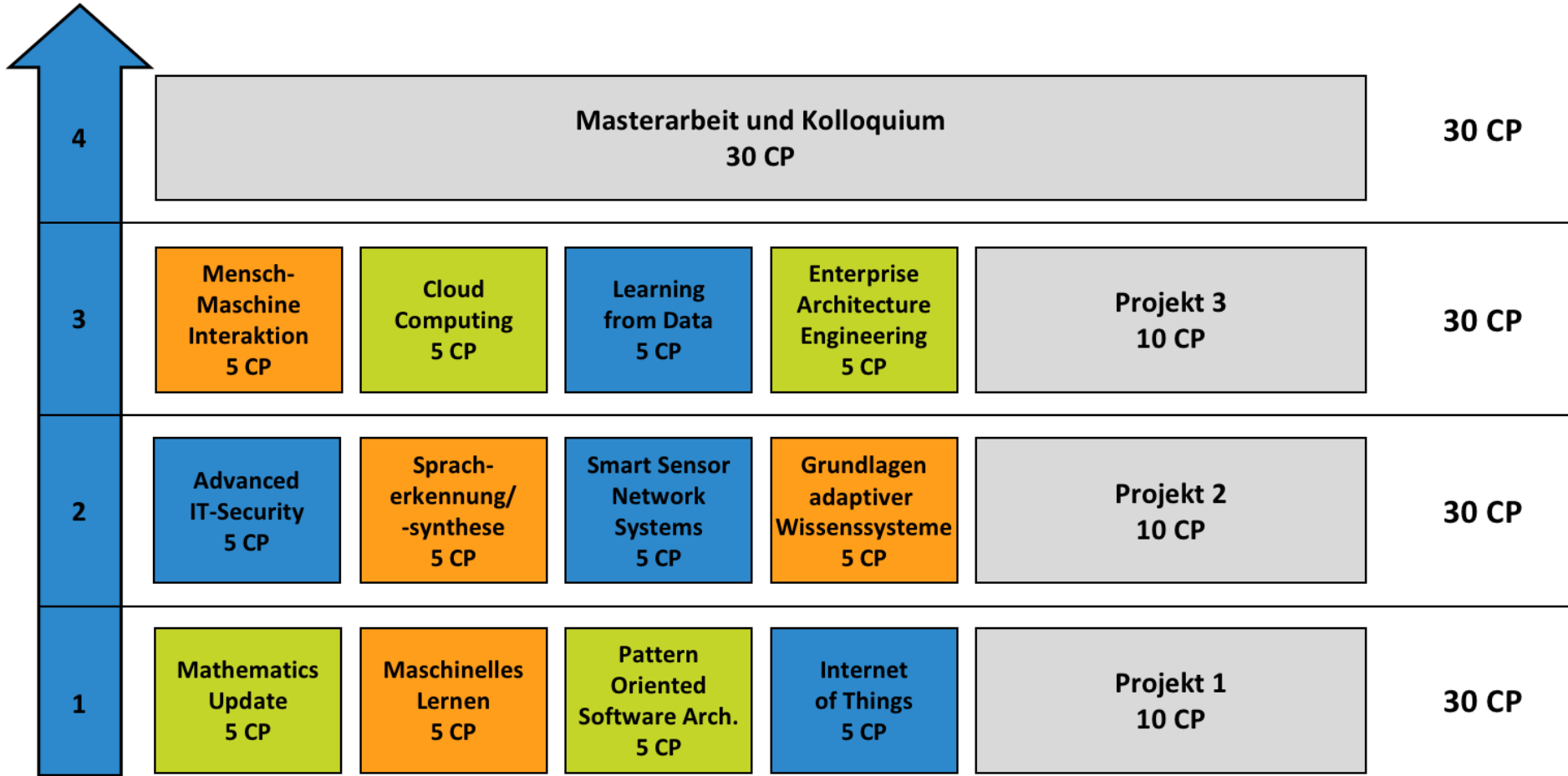
- selbständiges Bearbeiten wissenschaftlicher Fragestellung aus Fachgebieten der Informatik nach wissenschaftlichen Methoden, d. h. Untersuchen und Strukturieren des Fachgebiets, Entwickeln von Lösungen für die gegebene Fragestellung und Bewertung von Lösungsalternativen anhand geeigneter Kriterien
- sachgerechtes Darstellen und Diskussion der erarbeiteten Ergebnisse in Vorträgen oder Workshops unter Einsatz geeigneter Medien
- kooperatives und verantwortliches Arbeiten in Gruppen
- Einschätzen der eigenen Stärken und Schwächen im Hinblick auf eine führende Rolle, wie z. B. Softwarearchitekt oder Qualitätsmanager
- Entwickeln von Perspektiven für die eigene berufliche Zukunft.

Die Absolventinnen und Absolventen sind weiter befähigt, Implikationen ihres Handelns in Form zukünftiger Probleme, Technologien und Entwicklungen zu antizipieren.



* Jeweils ein Projekt ist thematisch einem der drei Studienfelder zugeordnet!

2 Beispielhafter Studienverlauf



3 ECTS-/Workload-Übersicht

| Nr. | Modultitel | ECTS | Prüfungsform | Sprache | Gewicht |
|-----|---|------|--|----------|---------|
| 1 | Grundlagen adaptiver Wissenssysteme | 5 cp | Mündliche Prüfung min. 20 Min. und max. 30 Min. | Deutsch | 1 |
| 2 | Maschinelles Lernen | 5 cp | Klausur, Dauer 90 Min. | Deutsch | 1 |
| 3 | Human Machine Interaction | 5 cp | Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes) | Englisch | 1 |
| 4 | Spracherkennung und -synthese | 5 cp | Mündliche Prüfung min. 20 Min. und max. 30 Min. | Deutsch | 1 |
| 5 | Bildererkennung | 5 cp | Mündliche Prüfung min. 20 Min. und max. 30 Min. | Deutsch | 1 |
| 6 | Robotics | 5 cp | Written documentation of project result (processing time 6 weeks), presentation of at least 15 min. and max. 30 min. | Englisch | 1 |
| 7 | Ontologiesprachen und Semantic Web | 5 cp | Klausur am Rechner, Dauer 90 Min. | Deutsch | 1 |
| 8 | Künstliche Intelligenz | 5 cp | Klausur, Dauer 90 Min. | Deutsch | 1 |
| 9 | Learning from Data | 5 cp | Written project report (submission period 6 weeks) with presentation (min. 15, max. 45 minutes) | Englisch | 1 |
| 10 | Internet of Things | 5 cp | Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes) | Englisch | 1 |
| 11 | Advanced IT-Security | 5 cp | Written examination (120 minutes) | Englisch | 1 |
| 12 | Smart Sensor Network Systems | 5 cp | Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes) | Englisch | 1 |
| 13 | Introductory Data Analysis | 5 cp | Written (computer) examination of 90 minutes duration | Englisch | 1 |
| 14 | Data Mining Methods | 5 cp | Written computer-based examination (90 minutes) | Englisch | 1 |
| 15 | Simulation Methods | 5 cp | Written examination of 90 minutes duration | Englisch | 1 |
| 16 | Advanced Data Structures and Algorithms | 5 cp | Written report in the form of a scientific contribution (processing time 6 weeks) and oral presentation of the results in the form of an event talk according to the rules of a scientific society, i.e., Springer, ACM, IEEE (min. 10 and max. 20 minutes). | Englisch | 1 |

| | | | | | |
|----|--|-------|---|----------|---|
| 17 | Pattern Oriented Software Architecture | 5 cp | Oral examination of at least 15 and maximum 45 minutes duration | Englisch | 1 |
| 18 | Cloud Computing | 5 cp | Written assignment (processing time 6 weeks) with presentation (min. 15, max. 30 minutes) | Englisch | 1 |
| 19 | Advanced Distributed Systems | 5 cp | Written examination of 90 minutes duration | Englisch | 1 |
| 20 | Advanced Formal Modelling | 5 cp | Written computer-based examination (90 minutes) | Englisch | 1 |
| 21 | Formal Specification and Verification | 5 cp | Written computer-based examination (90 minutes) | Englisch | 1 |
| 22 | Mathematics Update | 5 cp | Written examination of 90 minutes duration | Englisch | 1 |
| 23 | Advanced Testing Methods | 5 cp | Written examination of 90 minutes duration | Englisch | 1 |
| 24 | Enterprise Architecture Engineering | 5 cp | Portfolio mit folgenden Werkstücken: 1. Projektarbeit (Bearbeitungszeit 8 Wochen) mit Präsentation (mindestens 20, maximal 30 Minuten). In diesem Werkstück sind maximal 50 Punkte erreichbar. 2. Schriftliches Testat (Prüfungsdauer 60 Minuten). In diesem Werkstück sind maximal 50 Punkte erreichbar. Die Note ergibt sich aus der Summe der erreichten Punktzahlen. Zum Bestehen reichen 50% der erreichbaren Punkte aus. | Deutsch | 1 |
| 25 | Current Topics in Computer Science | 5 cp | Paper written according to international scientific journal standards (processing time 6 weeks) and oral presentation (min. 15, max. 30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation. | Englisch | 1 |
| 26 | Projekt Intelligente Systeme | 10 cp | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | Deutsch | 2 |
| 27 | Projekt Digitalisierung | 10 cp | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | Deutsch | 2 |
| 28 | Projekt Softwaretechnik | 10 cp | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | Deutsch | 2 |
| 29 | Masterarbeit mit Kolloquium | 30 cp | Master-Arbeit (Bearbeitungszeit 22 Wochen) mit Kolloquium (mindestens 30, maximal 60 Minuten) | Deutsch | 6 |

4 Modulbeschreibungen

Modul 1 Grundlagen adaptiver Wissenssysteme

| | | |
|---|---|----|
| Modultitel | Grundlagen adaptiver Wissenssysteme | PO |
| Modulnummer | 1 | PO |
| Modulcode | Wissen | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | BaSys – Intelligente Systeme (M.Sc.) | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Gute Grundlagenkenntnisse in Mathematik empfohlen | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Mündliche Prüfung mind. 20 Min. und max. 30 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach Abschluss des Moduls über ein grundlegendes Verständnis über adaptive-lernende Systeme mit Schwerpunkt auf den Paradigmen „genetische Algorithmen“, „genetische Programmierung“, sowie „Classifier Systeme“.</p> <p>Die Studierenden sind in der Lage, zu Problemstellungen selbständig geeignete adaptive Systeme zu entwerfen, zu implementieren und zu evaluieren.</p> <p>Personale Kompetenzen: Mit der Arbeit in kleinen Teams sind sie vertraut und hierdurch befähigt Kritik und Konflikten im Team reflektiert zu begegnen.</p> <p>Ihre Lösungsansätze können sie gegenüber Fachvertretern und auch Laien präsentieren und argumentativ vertreten.</p> <p>Darüber hinaus haben sie ein besonderes Verständnis für die speziellen Bedingungen von adaptiven Prozessen entwickelt und sind für gesellschaftsrelevante Fragestellungen mit solchen Phänomenen sensibilisiert.</p> | PO |
| Inhalte des Moduls | Grundlagen adaptiver Wissenssysteme | PO |

| | | |
|-------------------------------------|---|----|
| Lehrformen des Moduls | Seminaristische Vorlesung mit Anwendungsstudien | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Sommersemester | PO |
| Modulkoordination | Prof. Dr. Thomas Gabel | |
| Hinweise | Zur Verfügung stehen eine eLearning-Plattform, ein öffentliches Wiki des Studiengangs, eine Softwareversionsverwaltung, eine webbasierte Plattform für die benutzte Software, ein Onlineskript. | |

Unitbeschreibung 1: Grundlagen adaptiver Wissenssysteme

| | |
|--|---|
| Name der Unit | Grundlagen adaptiver Wissenssysteme |
| Code | |
| Name des Moduls | Grundlagen adaptiver Wissenssysteme (1) |
| Inhalte der Unit | <p>Die Vorlesung behandelt die Klasse adaptiver lernender Systeme, die in der Lage sind, ihr Wissen und ihre Fähigkeiten zum Handeln durch Interaktion mit ihrer Umgebung zu optimieren. Es werden Verfahren und Algorithmen des optimierenden Lernens besprochen, bei denen das System seine Handlungsstrategie rein auf Basis der Information über die Güte seiner getroffenen Entscheidungen adaptiert.</p> <ul style="list-style-type: none"> • Einführung: Wissen und Adaptivität im Bereich intelligenter Systeme • Formalisierung von Wissen und von sequenziellen Entscheidungsproblemen • Markov'sche Entscheidungsprozesse, Actor-Critic-Architektur • wertfunktionsbasierte Verfahren (dynamisches Programmieren, Wertiteration, zeitliche Differenzmethoden, modellfreie Verfahren) • direkte Strategiesuche (gradienten- oder fitnessbasiert) • ausgewählte fortgeschrittene Verfahren (Neuroevolution, Batch-Mode) <p>Der Anwendungsbezug wird in der Vorlesung einerseits durch viele Praxisbeispiele sowie durch praktische Übungsaufgaben hergestellt.</p> |
| Lehrformen | Seminaristische Vorlesung mit Anwendungsstudien |
| SWS der Unit | 4 h |
| Workload (h) | 150 h |
| Anteil der Präsenzzeit (h) | 60 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 15 h |
| Anteil Selbststudium (h) | 35 h |
| Anteil Praxiszeit (h) | 40 h |

| | |
|---|--|
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Thomas Gabel |
| Basis-Literatur | A. Barto, R. Sutton: Reinforcement Learning: An Introduction D. Bertsekas, J. Tsitsiklis: Neuro-Dynamic Programming M. Wiering: Reinforcement Learning State-of-the-Art Zu Semesterbeginn werden zusätzlich aktuelle Literaturhinweise gegeben. |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | Zur Verfügung stehen eine eLearning-Plattform, ein öffentliches Wiki des Studiengangs, eine Softwareversionsverwaltung, eine webbasierte Plattform für die benutzte Software, ein Onlineskript. |

Modul 2 Maschinelles Lernen

| | | |
|---|---|----|
| Modultitel | Maschinelles Lernen | PO |
| Modulnummer | 2 | PO |
| Modulcode | ML | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | BaSys – Intelligente Systeme (M.Sc.) | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 1. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Gute Grundlagenkenntnisse in Mathematik (Algebra, Analysis, Statistik) empfohlen | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Klausur, Dauer 90 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach diesem Modul über ein grundlegendes Verständnis eines Standardmodells aus dem Bereich Maschinelles Lernen, sowie der generellen Terminologie und Prinzipien des Feldes. Dazu verfügen sie über ein hinreichendes Verständnis der anwendungsrelevanten mathematischen, statistischen und numerischen Aspekte der Thematik</p> <p>Sie sind in der Lage, dieses Wissen</p> <ul style="list-style-type: none"> • in verschiedenen Anwendungskontexten selbständig auf Problemstellungen anzuwenden. • problemorientiert auf einer entsprechenden Softwareplattform umsetzen. <p>Personale Kompetenzen: Die Studierenden sind in der Lage sich eigenständig komplexe theoretische Modelle zu erarbeiten und dem Stand der Forschung zu folgen.</p> <p>Sie können erarbeitete Lösungsansätze sowohl gegenüber Fachkollegen als auch Fachfremden angemessen präsentieren. Aufgrund der Komplexität der Anforderungen sind sie in der Lage im Team explorativ und effizient an einer Aufgabenstellung zu arbeiten und das Ziel nicht aus den Augen zu verlieren.</p> | PO |

| | | |
|-------------------------------------|---|----|
| Inhalte des Moduls | Maschinelles Lernen | PO |
| Lehrformen des Moduls | Seminaristischer Unterricht mit Übungen | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Wintersemester | PO |
| Modulkoordination | Prof. Dr. Ute Bauer-Wersing | |
| Hinweise | Der Kurs wird zusätzlich unterstützt durch eine eLearning-Plattform | |

Unitbeschreibung 2: Maschinelles Lernen

| | |
|--|--|
| Name der Unit | Maschinelles Lernen |
| Code | |
| Name des Moduls | Maschinelles Lernen (2) |
| Inhalte der Unit | <p>In der Vorlesung wird eine Einführung in die grundlegenden und weiterführenden Sichtweisen, Probleme, Methoden und Techniken des Maschinellen Lernens vermittelt.</p> <p>Es werden u. a. folgende Themen behandelt:</p> <ul style="list-style-type: none"> • Einführung und historische Entwicklung des Maschinellen Lernens • Gehirn als Vorbild, biologische Motivation • Formalisierung und Modellierung am Beispiel Neuronaler Netze • Überblick über Problemstellungen (Klassifikation, Regression, Clustering) • Supervised (überwachtes) und unsupervised (unüberwachtes) Lernen • Perzeptron, Lineare Trennbarkeit, Bezug zur Bayes-Klassifikation • Lineare Netze, Online PCA und Dimensionsreduktion, Merkmalsauswahl • Multilagen-Perzeptron • Error-Backpropagation, Optimierung • Strukturelle Überlegungen (Generalisierungsfähigkeit, Abschätzung Approximationsfehler, MLE) • Überlegungen zum Algorithmus Gradientbasiertes Lernen, Online vs. Batch-Lernen, Crossvalidation • Deep Learning, Convolutional Networks etc. <p>Jeweils ergänzt durch praktische Übungen zu den vermittelten Theorien und Konzepten unter Verwendung einer geeigneten Software.</p> |
| Lehrformen | Seminaristischer Unterricht mit Übungen |
| SWS der Unit | 4 h |
| Workload (h) | 150 h |
| Anteil der Präsenzzeit (h) | 60 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 15 h |

| | |
|---|--|
| Anteil Selbststudium (h) | 35 h |
| Anteil Praxiszeit (h) | 40 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Bauer-Wersing |
| Basis-Literatur | <p>C. M. Bishop (2006). Pattern Recognition and Machine Learning. Springer Verlag.</p> <p>T. Hastie, R. Tibshiranie, J. Friedmann (2001). The Elements of Statistical Learning. Springer Verlag.</p> <p>E. Alpaydin (2008). Maschinelles Lernen. Oldenbourg Wissenschaftsverlag.</p> <p>S. Sharma (1996). Applied Multivariate Techniques. Wiley & Sons.</p> <p>Ggf. werden zu Semesterbeginn zusätzlich aktuelle Literaturhinweise ergänzt.</p> |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | Der Kurs wird zusätzlich unterstützt durch eine eLearning-Plattform |

Modul 3 Human Machine Interaction

| | | |
|---------------------------------|---|----|
| Module title | Human Machine Interaction | PO |
| Module number | 3 | PO |
| Module code | MMI | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | Barrierefreie Systeme (M.Sc.), High Integrity Systems (M.Sc.) | PO |
| Module duration | One Semester | PO |
| Recommended semester | 3. Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | Basic knowledge of software engineering or system engineering | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • Overall goal is to gain basic knowledge about HMI as part of a systems engineering process, psychological conditions of a user, how to describe the behavior of the user, how to derive requirements for an interface, and how to test the usability of an interface • Transferring the gained knowledge into a theoretical model solving a concrete problem • Transferring the theoretical model into a working demonstrator • Working with others toward a same goal, and contributing to the capability of the teamwork. • Creating an atmosphere of respect, helpfulness and cooperation. • Validating the demonstrator with the aid of usability tests • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper | PO |
| Module contents | Human Machine Interaction – Project | PO |

| | | |
|-------------------------|---|----|
| Module teaching methods | After an introduction the student teams work in a project. They have to use official textbooks and/ or scientific papers to back up their knowledge. The professor can be interviewed on demand | PO |
| Module language | English | PO |
| Module availability | Summer semester | PO |
| Module coordination | n/a | |
| Comments | | |

Unitbeschreibung 3: Human Machine Interaction Project

| | |
|---|---|
| Unit title | Human Machine Interaction – Project |
| Code | |
| Module title | Human Machine Interaction |
| Unit contents | <p>HMI as part of a systems engineering process</p> <ul style="list-style-type: none"> • Psychological aspects of human - computer interaction • Behavior modeling and interface design • HMI prototyping <ul style="list-style-type: none"> • Usability tests |
| Teaching methods | R&D project with small groups (4 students max.) |
| Semester periods (hours) per week | 4 |
| Workload (h) | 150 h |
| Class hours | 60h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 80h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Matthias Deegener, Prof. Dr. Gerd Döben-Henisch |
| Recommended reading | Current literature, e.g. research papers, will be announced at the beginning of the semester |
| Assessment type and form | None |
| Assessment grading | None |

| | |
|----------|--|
| Comments | |
|----------|--|

Modul 4 Spracherkennung und -synthese

| | | |
|---|--|----|
| Modultitel | Spracherkennung und -synthese | PO |
| Modulnummer | 4 | PO |
| Modulcode | Sprache | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | BaSys – Intelligente Systeme (M.Sc.) | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Wahlpflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Empfohlen: Kenntnisse aus dem Modul Maschinelles Lernen, Programmierkenntnisse, gute Grundlagen in Mathematik (Lineare Algebra, Analysis, Wahrscheinlichkeitsrechnung) | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Mündliche Prüfung mind. 20 Min. und max. 30 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach diesem Modul über ein grundlegendes Verständnis von Sprache als menschlichem Kommunikationsmittel, sowie den neurologischen und neuromotorischen Grundlagen der Perzeption und Sprachproduktion; auch krankhafte und altersbedingte Einschränkungen der Sprachwahrnehmung und -produktion werden erörtert.</p> <p>Sie sind in der Lage</p> <ul style="list-style-type: none"> • selbständig aktuelle mathematische Modelle zur Spracherkennung und -synthese anzuwenden • selbständig Techniken zur Realisierung von Spracherkennungs- und Sprachsynthesystemen anzuwenden. • sich eigenständig in komplexe Anwendungspakete und Toolkits zur Spracherkennung und Synthese einzuarbeiten. <p>Personale Kompetenzen: Sie haben die Bereitschaft, sich mit den Inhalten auseinander zu setzen und Risiken und Folgen Ihrer Lösungen zu antizipieren.</p> | PO |

| | | |
|-------------------------------------|--|----|
| | <p>Sie sind offen für neue Ansätze und probieren das erlernte Wissen tatkräftig aus, auch mit der Bereitschaft neue Ansätze vorzuschlagen. Sie sind in der Lage flexibel eigene Initiativen zu entwickeln.</p> <p>Mit der Arbeit in kleinen Teams sind sie vertraut und hierdurch befähigt Kritik und Konflikten im Team reflektiert zu begegnen.</p> <p>Ihre Lösungsansätze können sie gegenüber Fachvertretern und auch Laien präsentieren und argumentativ vertreten.</p> <p>Darüber hinaus haben sie ein besonderes Verständnis für gesellschaftsrelevante Fragestellungen entwickelt.</p> | |
| Inhalte des Moduls | Spracherkennung und -synthese | PO |
| Lehrformen des Moduls | Seminaristischer Unterricht mit Übungen | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Sommersemester | PO |
| Modulkoordination | Prof. Dr. Ute Bauer-Wersing | |
| Hinweise | Zur Verfügung stehen eine eLearning-Plattform und ein Lab | |

Unitbeschreibung 4: Spracherkennung und -synthese

| | |
|--|---|
| Name der Unit | Spracherkennung und -synthese |
| Code | |
| Name des Moduls | Spracherkennung und -synthese (4) |
| Inhalte der Unit | <ol style="list-style-type: none"> 1. Einführung in die Physik des Schalls, Phonetik, Fourieranalyse und Zeit-Frequenzdarstellung 2. Text-to-Speech-Systeme: Textverarbeitung und -normalisierung, POS-Tagging, Prosodiesteuerung, konkatenative Synthese, Unit-Selection 3. Spracherkennung: Hidden-Markov-Modelle, Baum_Welch-Algorithmus, Viterbi-Dekodierung, Akustische Modellierung, Merkmalsgenerierung – MFCCs, Sprachmodellierung |
| Lehrformen | Seminaristischer Unterricht mit Übungen |
| SWS der Unit | 4 h |
| Workload (h) | 150 h |
| Anteil der Präsenzzeit (h) | 60 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 15 h |
| Anteil Selbststudium (h) | 35 h |
| Anteil Praxiszeit (h) | 40 h |
| Sprache der Unit | Deutsch |

| | |
|---|--|
| Lehrende/-r | Prof. Dr. Bauer-Wersing |
| Basis-Literatur | <p>X.Huang, A. Acero, H. Hon (2001). Spoken Language Processing. Prentice Hall PTR.</p> <p>S. Euler (2006). Grundkurs Spracherkennung. Vieweg & Sohn Verlag.</p> <p>Benesty, Jacob, M. Mohan Sondhi, and Yiteng Huang, eds (2007). Springer handbook of speech processing. Springer Science & Business Media.</p> <p>Zusätzlich wird diese zu Semesterbeginn durch aktuelle Literaturhinweise ergänzt.</p> |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | Zur Verfügung stehen eine eLearning-Plattform und ein Lab |

Modul 5 Bilderkennung

| | | |
|---|--|----|
| Modultitel | Bilderkennung | PO |
| Modulnummer | 5 | PO |
| Modulcode | Bild | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | BaSys – Intelligente Systeme (M.Sc.) | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Wahlpflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Keine | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Mündliche Prüfung mind. 20 Min. und max. 30 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach diesem Modul über ein grundlegendes Verständnis der Bildverarbeitung bei künstlichen intelligenten Systemen insbesondere bei jenen, die mit Menschen interagieren sollen. Sie kennen die Struktur und Funktionsweise von Software für die Modellierung von bilderkennenden Strukturen. Dieses Wissen können sie selbständig auf Problemstellungen anwenden, durch Analyse, des Problems, und durch Transfer von Wissen.</p> <p>Die Studierenden sind in der Lage, Probleme angemessen zu analysieren, technische Lösungen zu erarbeiten und zu evaluieren.</p> <p>Personale Kompetenzen: Mit der Arbeit in kleinen Teams sind sie vertraut und hierdurch befähigt Kritik und Konflikten im Team reflektiert zu begegnen.</p> <p>Ihre Lösungsansätze können sie gegenüber Fachvertretern und auch Laien präsentieren und argumentativ vertreten.</p> <p>Darüber hinaus haben sie ein besonderes Verständnis für gesellschaftsrelevante Fragestellungen entwickelt.</p> | PO |
| Inhalte des Moduls | Bilderkennung | PO |

| | | |
|-------------------------------------|---|----|
| Lehrformen des Moduls | Seminaristischer Unterricht mit Anwendungsstudien | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Sommersemester | PO |
| Modulkoordination | Prof. Dr. Peter Nauth | |
| Hinweise | | |

Unitbeschreibung 5: Bilderkennung

| | |
|---|---|
| Name der Unit | Bilderkennung |
| Code | |
| Name des Moduls | Bilderkennung (5) |
| Inhalte der Unit | <ul style="list-style-type: none"> • Texte und exemplarische Problemstellungen zur <i>Rolle von Bildverarbeitung als Voraussetzung zur Bilderkennung</i> in Anwendungskontexten, insbesondere bei technischen Systemen, die mit Menschen interagieren sollen. • Texte und Übungen zu <i>theoretischen Modellen</i> der Bildverarbeitung im Kontext der Bilderkennung. • Texte und Projekte zu <i>Softwarestrukturen</i> für die Realisierung von Bildverarbeitung und Bilderkennung. |
| Lehrformen | Seminaristischer Unterricht mit Anwendungsstudien |
| SWS der Unit | 4 h |
| Workload (h) | 150 h |
| Anteil der Präsenzzeit (h) | 60 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 15 h |
| Anteil Selbststudium (h) | 35 h |
| Anteil Praxiszeit (h) | 40 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Peter Nauth, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Andreas Pech |
| Basis-Literatur | Es gibt ein Onlineskript mit sehr ausführlichen Literaturhinweisen. Dieses wird zusätzlich zu Semesterbeginn durch aktuelle Literaturhinweise ergänzt. |
| Art und Form des Leistungsnachweises der Unit | Keine |

| | |
|--|---|
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | Zur Verfügung stehen eine eLearning-Plattform und ein Labor |

Modul 6 Robotics

| | | |
|---------------------------------|---|----|
| Module title | Robotics | PO |
| Module number | 6 | PO |
| Module code | Robot | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | BaSys – Intelligente Systeme (M.Sc.), Information Technology (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 3rd Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | None | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written documentation of project result (processing time 6 weeks), presentation of at least 15 min. and max. 30 min. | PO |
| Learning outcomes and skills | <p>Professional competence: The student will have a thorough knowledge regarding the architecture, hardware and software of robotic systems. He/she is familiar with intelligent algorithms and their application in intelligent sensors, action planning and decision making, especially with respect autonomous mobile robots.</p> <p>The students will be able</p> <ul style="list-style-type: none"> • to develop mechanics and hardware of robots by means of respective modules. • to develop and program algorithms for sensing, autonomous behavior and control of actuators. • to deal with new, unfamiliar problems and to gather knowledge in new areas. <p>Personal Competence: The students can work in small teams with diverging roles, and they can work out technical solutions mediated by discourse.</p> <p>They are able to communicate his results in an argumentative way with regard to professionals as well as non-professionals</p> <p>They can work in teams as well as alone, showing responsibility and reflectiveness.</p> | PO |

| | | |
|-------------------------|--|----|
| | They can analyze problems from different points of views. | |
| Module contents | Robotics – Lectures, Robotics – Project | PO |
| Module teaching methods | Lectures: Interactive teaching Project: Teamwork in small development groups | PO |
| Module language | English | PO |
| Module availability | Winter semester | PO |
| Module coordination | Prof. Dr. Peter Nauth | |
| Comments | | |

Unit description of module 6: Robotics – Lectures

| | | |
|---|---|--|
| Unit title | Robotics – Lectures | |
| Code | | |
| Module title | Robotics (6) | |
| Unit contents | <ul style="list-style-type: none"> • Theory of the architecture, hardware and software of robotic systems. • Theory of intelligent algorithms for robots and their application in intelligent sensors, action planning and decision making. | |
| Teaching methods | Lectures: Interactive teaching | |
| Semester periods (hours) per week | 2 h | |
| Workload (h) | 80 h | |
| Class hours | 30 h | |
| Total time of examination incl. preparation (h) | 15 h | |
| Total time of individual study (h) | 35 h | |
| Total time of practical training (h) | 0 h | |
| Unit language | English | |
| Lecturer | Prof. Dr. Peter Nauth | |
| Recommended reading | Bajd,T., Mihelj,M., Lenarčič,J., Stanovnik,A., Munič,M. Robotics Series: Intelligent Systems, Control and Automation: Science and Engineering, Vol. 43, Springer Verlag, 2010; ISBN: 978-1-84628-641-4 | |

| | |
|-----------------------------|---|
| | Siciliano, Bruno; Khatib, Oussama (Eds.) Springer Handbook of Robotics, Springer Verlag, 2008; ISBN: 978-3-540-23957-4 Peter Cork: Robotics, Vision and Control; ISBN 978-3-642-20143-1 Current literature will be announced at the beginning of each semester. |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | |

Unit description of module 6: Robotics – Project

| | |
|---|--|
| Unit title | Robotics – Project |
| Code | |
| Module title | Robotics (6) |
| Unit contents | <ul style="list-style-type: none"> • Planning a small project to apply the learned theoretical concepts. • Implement parts of the theory in a running version. |
| Teaching methods | Project: Teamwork in small development groups |
| Semester periods (hours) per week | 2 h |
| Workload (h) | 70 h |
| Class hours | 15 h |
| Total time of examination incl. preparation (h) | 0 h |
| Total time of individual study (h) | 40 h |
| Total time of practical training (h) | 15 h |
| Unit language | English |
| Lecturer | Prof. Dr. Peter Nauth |
| Recommended reading | Current literature will be announced at the beginning of each semester. |
| Assessment type and form of | None |

| | |
|--------------------|------|
| Assessment grading | None |
| Comments | |

Modul 7 Ontologiesprachen und Semantic Web

| | | |
|---|--|----|
| Modultitel | Ontologiesprachen und Semantic Web | PO |
| Modulnummer | 7 | PO |
| Modulcode | OWL | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Wahlpflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Kenntnisse in XML und in Logik | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Klausur am Rechner, Dauer 90 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach Abschluss dieses Moduls über einen grundlegendes Verständnis des Semantic Webs und der darunter liegenden Technologien. Sie verstehen die formale Semantik und die logischen Grundlagen der Ontologiesprachen RDFS und OWL. Sie kennen die theoretischen Grenzen der logikbasierten Wissensrepräsentation und die Probleme, die in der Praxis auftauchen.</p> <p>Sie sind in der Lage</p> <ul style="list-style-type: none"> komplexes Wissen aus einem beliebigen Anwendungsgebiet mit Hilfe der Ontologiesprache OWL zu modellieren und dieses Modell in einem Ontologieeditor umzusetzen. <p>Personale Kompetenzen: Die Studierenden sind in der Lage sich eigenständig komplexe theoretische Modelle zu erarbeiten und dem Stand der Forschung zu folgen. Sie können erarbeitete Lösungsansätze sowohl gegenüber Fachkollegen als auch Fachfremden präsentieren. Aufgrund der Komplexität der Anforderungen sind sie in der Lage in kleinen Teams explorativ und effizient an einer Aufgabenstellung zu arbeiten.</p> | PO |

| | | |
|-------------------------------------|--|----|
| Inhalte des Moduls | Ontologiesprachen und Semantic Web – Vorlesung, Ontologiesprachen und Semantic Web – Übungen | PO |
| Lehrformen des Moduls | Seminaristische Vorlesung und Übungen | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Sommersemester | PO |
| Modulkoordination | Prof. Dr. Barış Sertkaya | |
| Hinweise | | |

Unitbeschreibung 7: Ontologiesprachen und Semantic Web – Vorlesung

| | |
|--|--|
| Name der Unit | Ontologiesprachen und Semantic Web – Vorlesung |
| Code | |
| Name des Moduls | Ontologiesprachen und Semantic Web (7) |
| Inhalte der Unit | <ul style="list-style-type: none"> • Einführung: Die Vision des Semantic Web • Datenmodellierung mit XML • RDF und RDF Schema zur Darstellung von Metadaten und einfachen Ontologien • Formale Semantik von RDF(S) • Die Web Ontology Language (OWL) und Ontologien in OWL • Formale Semantik von OWL <ul style="list-style-type: none"> ○ Beschreibungslogiken und Schlussfolgerung • Die SPARQL-Anfragesprache • Praktische Anwendungen <ul style="list-style-type: none"> ○ Linked Data ○ Ontology Engineering |
| Lehrformen | Seminaristische Vorlesung |
| SWS der Unit | 2 h |
| Workload (h) | 70 h |
| Anteil der Präsenzzeit (h) | 30 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 10 h |
| Anteil Selbststudium (h) | 30 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Barış Sertkaya |
| Basis-Literatur | G. Antoniou, P. E. Groth, F. Van Harmelen, R. Hoekstra. A Semantic Web Primer. The MIT Press, 2012. ISBN 978- 0262018289. |

| | |
|---|---|
| | <p>P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure. Semantic Web Grundlagen. Springer, 2008. ISBN 978-3-540-33993-9.</p> <p>P. Szeredi, G. Lukcsy, T. Benkö. The Semantic Web Explained: The Technology and Mathematics behind Web 3.0. Cambridge University Press, 2014. ISBN 978-0521700368.</p> <p>F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. F. Patel-Schneider. The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2003.</p> |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Unitbeschreibung 7: Ontologiesprachen und Semantic Web – Übungen

| | |
|---|--|
| Name der Unit | Ontologiesprachen und Semantic Web – Übungen |
| Code | |
| Name des Moduls | Ontologiesprachen und Semantic Web (7) |
| Inhalte der Unit | Ziel der Übungen ist es, die Studierenden anhand kleinen, selbst auszuarbeitenden Szenarien für die behandelten Themenkreise zu sensibilisieren. |
| Lehrformen | Übung |
| SWS der Unit | 2 h |
| Workload (h) | 80 h |
| Anteil der Präsenzzeit (h) | 30 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 50 h |
| Anteil Praxiszeit (h) | 30 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Barış Sertkaya |
| Basis-Literatur | siehe Unit Ontologiesprachen und Semantic Web – Vorlesung |
| Art und Form des Leistungsnachweises der Unit | Keine |

| | |
|--|-------|
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 8 Künstliche Intelligenz

| | | |
|---|---|----|
| Modultitel | Künstliche Intelligenz | PO |
| Modulnummer | 8 | PO |
| Modulcode | KI | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 1. Semester | PO |
| Art des Moduls | Wahlpflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP/150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Keine | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Klausur, Dauer 90 Min. | PO |
| Lernergebnisse und Kompetenzen | <p>Fachkompetenzen: Die Studierenden verfügen nach diesem Modul über ein grundlegendes Verständnis von Problemen, Methoden und Techniken zur Entwicklung und Bewertung künstlicher intelligenter Systeme. Dazu gehören Kenntnisse von klassischen und aktuellen theoretischen Modellen der symbolischen künstlichen Intelligenz sowie von Softwaresystemen für die Implementierung kennengelernter Methoden und Algorithmen. Sie sind in der Lage, selbständig KI-basierte Lösungen zu gegebenen Problemstellungen zu konzipieren, die Funktionsweise ausgewählter Lernalgorithmen händisch nachzuvollziehen, diese eigenständig zu implementieren sowie gängige Softwarelösungen in der Umsetzung ihrer Lösungsideen zielorientiert einzusetzen.</p> <p>Personale Kompetenzen: Die Studierenden können Ihre Lösungsansätze sowohl gegenüber Fachvertretern als auch Fachfremden präsentieren und argumentativ vertreten. Sie verstehen die Grundzüge wissenschaftlichen Arbeitens, beherrschen Literaturrecherchen sowie die Kommunikation von Inhalten mittels Präsentationen und sind vertraut in der Nutzung der eLearning-Plattform.</p> | PO |
| Inhalte des Moduls | Künstliche Intelligenz | PO |

| | | |
|-------------------------------------|---|----|
| Lehrformen des Moduls | Seminaristischer Unterricht mit Übungen | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Wintersemester | PO |
| Modulkoordination | Prof. Dr. Thomas Gabel | |
| Hinweise | | |

Unitbeschreibung 8: Künstliche Intelligenz

| | |
|--|--|
| Name der Unit | Künstliche Intelligenz |
| Code | |
| Name des Moduls | Künstliche Intelligenz (8) |
| Inhalte der Unit | <p>Im ersten Teil der Vorlesung wird eine Einführung in die grundlegenden Sichtweisen, Probleme, Methoden und Techniken der künstlichen Intelligenz vermittelt. Es werden u. a. folgende Themen behandelt:</p> <ul style="list-style-type: none"> • Einführung und historische Entwicklung der KI • der Agentenbegriff in der KI • Problemlösen und Suche • Handlungsplanung, Logik, Schließen • symbolisches Lernen • Darstellung und Verarbeitung unsicheren Wissens <p>Darauf aufbauend werden im zweiten Teil spezielle ausgewählte, aktuelle Themen aus der KI vertieft, hierzu können u. a. gehören: Spieltheorie, Deep Learning, Multi-Agenten-Systeme, fallbasiertes Schließen, evolutionäre Algorithmen, Optimierung.</p> <p>Die theoretischen Konzepte und Inhalte werden jeweils ergänzt durch theoretische, anwendungsorientierte sowie praktische Übungen unter Verwendung geeigneter Software.</p> |
| Lehrformen | Seminaristischer Unterricht mit Übungen |
| SWS der Unit | 4 h |
| Workload (h) | 150 h |
| Anteil der Präsenzzeit | 60 h |
| Anteil Prüfungszeit inkl. Vorbereitung | 15 h |
| Anteil Selbststudium | 35 h |
| Anteil Praxiszeit | 40 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Thomas Gabel, Prof. Dr. Ute Bauer-Wersing |

| | |
|---|--|
| Basis – Literatur | S. Russell, P. Norvig: Artificial Intelligence: A Modern Approach Zu Semesterbeginn sowie im Laufe des Semesters werden zusätzliche aktuelle Literaturhinweise gegeben. |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 9 Learning from Data

| | | |
|---------------------------------|---|----|
| Module title | Learning from Data | PO |
| Module number | 9 | PO |
| Module code | LD | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M. Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 3rd semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | <ul style="list-style-type: none"> • Good knowledge in mathematics (Linear algebra, calculus of one and multiple variables, statistics) • Good programming capabilities | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written project report (processing time 6 weeks) with presentation (min. 15 minutes, max. 45 minutes) | PO |
| Learning outcomes and skills | <p>Professional expertise:</p> <p>The students gain a principal understanding of the mathematical and epistemological basics of statistical learning theory and machine learning. They are capable to independently apply their knowledge in various problem fields, e.g. robotics, big data etc. In addition they know the most important application fields of statistical learning theory and are able to assess the ethical and societal dimensions of applications. The students have the opportunity to test their findings in a prototype on a relevant platform and write a scientific paper.</p> <p>Personal expertise:</p> <p>The students are capable to independently work out complex theoretical models and to follow the state-of-the-art of current research. They are capable to write scientific publications and to present the results of their research for experts and lay persons alike.</p> | PO |
| Module contents | <p>Learning for Data – Lectures</p> <p>Learning from Data – Exercises</p> | PO |
| Module teaching methods | <p>Interactive lectures</p> <p>Exercises: Seminar</p> | PO |

| | | |
|---------------------|------------------------|----|
| Module language | English | PO |
| Module availability | Each semester | PO |
| Module coordination | Prof. Dr. Jörg Schäfer | |
| Comments | | |

Unitbeschreibung 9: Learning From Data - Lectures

| | |
|---------------|---|
| Unit title | Learning from Data– Lectures |
| Code | |
| Module title | Learning from Data |
| Unit contents | <ul style="list-style-type: none"> • Problem space • Is statistical or machine learning possible? (Outlook Vapnik-Chervonenkis theory, VC Dimension and Limit) • Statistical/Machine learning: Supervised, unsupervised, reinforcement learning, classification, regression • Probabilistic and non-probabilistic methods • Different models: <ul style="list-style-type: none"> o Linear model o Kernel methods o Support Vector Machines • Bayes methods <ul style="list-style-type: none"> o Gaussian processes o MCMC and particle filters • Neural networks <ul style="list-style-type: none"> o Genetic algorithms • Model selection <ul style="list-style-type: none"> o Overfitting and regularization o Bias-Variance tradeoff, error and noise • Training, testing, validation <ul style="list-style-type: none"> o Curse of dimensionality • Vapnik-Chervonenki theory, Hoeffding’s lemma, VC dimension and VC inequality • Structural Risk Minimization (SRM), Occham’s Razor • Overview of selected application fields • consequences of statistical learning theory • Philosophic implications (Are data more important than theories? Where are the limits of statistical learning theory?) • Ethical implications for society, principal problems of algorithmic decisions (Information privacy, individual freedom, checks-and balances of citizens, corporations, states etc.) • Lectures are accompanied by practical exercises with respect to theories and concepts, supported by the use of appropriate software. |

| | |
|---|--|
| Teaching methods | Lectures |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Prof. Dr. Jörg Schäfer |
| Recommended reading | <p>Yaser S. Abu-Mostafa, Malik Magdon-Ismael, and Hsuan-Tien Lin. Learning From Data. AMLBook, 2012.</p> <p>Nello Cristianini and John Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, 1 edition, 2000.</p> <p>Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning: data mining, inference and prediction. Springer, 2 edition, 2008.</p> <p>G. James, D. Witten, T. Hastie, and R. Tibshirani. An Introduction to Statistical Learning: with Applications in R. Springer Texts in Statistics. Springer New York, 2014.</p> <p>Vladimir N. Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.</p> |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | n/a |

Unitbeschreibung 9: Learning From Data – Exercise and Seminar

| | |
|---|--|
| Unit title | Learning from Data – Exercises and Seminar |
| Code | |
| Module title | Learning from Data |
| Unit contents | Elaboration on individual aspects from the lectures in exercises and seminar contributions |
| Teaching methods | Seminar with accompanying exercises |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Prof. Dr. Jörg Schäfer |
| Recommended reading | See Unit Learning from Data – Lectures |
| Assessment type and form of | |
| Assessment grading | |
| Comments | n/a |

Modul 10 Internet of Things

| | | |
|---------------------------------|--|----|
| Module title | Internet of Things | PO |
| Module number | 10 | PO |
| Module code | IoT | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | None | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Project (submission period 8 weeks) with oral presentation (min. 20 , max. 30 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • understand the basic technologies for the Internet of Things, • asses emerging technologies concerning their suitability, • get acquainted quickly with new technologies, and • develop new application fields. • to search for, read, summarize and cite scientific literature on a large scale; • to read and interpret national and international standards; • to write a report as a scientific paper; • to give a scientific talk. | PO |
| Module contents | Internet of Things – Seminar | PO |
| Module teaching methods | Seminar | PO |
| Module language | English | PO |
| Module availability | Annually | PO |
| Module coordination | Prof. Dr. Matthias Wagner | |
| Comments | | |

Unit description of module 10: Internet of Things – Seminar

| | |
|---|---|
| Unit title | Internet of Things – Seminar |
| Code | |
| Module title | Internet of Things |
| Unit contents | <p>The course will cover selected subjects from the following areas. The depth of coverage might vary.</p> <ul style="list-style-type: none"> • Technological foundation of the Internet of Things • HW Basics • Field-Bus systems • Wireless sensor networks • Middleware and integration into the Internet • Example(s) of relevant algorithms • HMI <p>Application examples</p> |
| Teaching methods | Seminar |
| Semester periods (hours) per week | 2 |
| Workload (h) | 150 h |
| Class hours | 36 |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 114 h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Deegener, Prof. Dr. Matthias Wagner |
| Recommended reading | |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | n/a |

Modul 11 Advanced IT-Security

| | | |
|---------------------------------|--|----|
| Module title | Advanced IT-Security | PO |
| Module number | 11 | PO |
| Module code | IT-Sec | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | <ul style="list-style-type: none"> • Good knowledge in introductory computer science, • Programming skills in C or Java, • Basic System Administration Skills in Windows and Unix Theoretical foundations of computer science, networks, operating systems | |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written examination (120 minutes) | PO |
| Learning outcomes and skills | Upon completion of this course, the student is able to: <ul style="list-style-type: none"> • to identify, analyze, and perhaps solve network-related security problems in computer systems. • to understand security problems in the combination of the Internet with Intranets. • to comprehend the need to protect all architectural levels. • to get an understanding of how to coordinate hardware and software to provide data security against internal and external attacks. • to communicate in international teams | PO |
| Module contents | Advanced IT-Security – Lectures Advanced IT-Security – Exercises | PO |
| Module teaching methods | Lectures: Interactive group lecturing Exercises: Teamwork in small groups | PO |
| Module language | English | PO |
| Module availability | Summer semester | PO |

| | |
|---------------------|-------------------------|
| Module coordination | Prof. Dr. Martin Kappes |
| Comments | |

Unit description of module 11: Advanced IT-Security – Lectures

| | |
|---|---|
| Unit title | Advanced IT-Security – Lectures |
| Code | |
| Module title | Advanced IT-Security |
| Unit contents | <ul style="list-style-type: none"> • Introduction • Cryptography, Computational Complexity and Computability • Security threats in computer networks and countermeasures and security protocols on all layers of the reference model • Firewalls, VPNs • Anomaly Detection <ul style="list-style-type: none"> • Further Topics |
| Teaching methods | Interactive group lecturing |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10h |
| Total time of individual study (h) | 30h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Martin Kappes |
| Recommended reading | Current literature to be announced at the beginning of the semester |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | |

Unit description of module 11: Advanced IT-Security – Exercises

| | |
|---|---|
| Unit title | Advanced IT-Security - Exercises |
| Code | |
| Module title | Advanced IT-Security |
| Unit contents | <ul style="list-style-type: none"> • Lab exercises with software tools pertaining to the contents described in the unit Advanced IT-Security – lectures • practical teamwork on real world problems • Lessons learned session after group work |
| Teaching methods | Teamwork in small software development groups, exercises |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80 h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50 h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Martin Kappes |
| Recommended reading | See unit Advanced IT-Security lectures |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | |

Modul 12 Smart Sensor Network Systems

| | | |
|---------------------------------|--|----|
| Module title | Smart Sensor Network Systems | PO |
| Module number | 12 | PO |
| Module code | SSN | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | <ul style="list-style-type: none"> • Knowledge in software and systems engineering, • C/C++ - programming • Numerical analysis | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Project (submission period 8 weeks) with presentation (min. 10 minutes, max. 20 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • understand the interface between computer science and the physical environment, • assess the challenges of the measuring process and the possible errors, • set up and program a Wireless Sensor Network and interface it with a standard network and/or the Internet, • participate in the solution of measuring tasks by cooperation with specialists of other disciplines • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper | PO |
| Module contents | Smart Sensor Network Systems – Project | PO |
| Module teaching methods | Project | PO |
| Module language | English | PO |
| Module availability | Summer semester | PO |

| | |
|---------------------|---------------------------|
| Module coordination | Prof. Dr. Matthias Wagner |
| Comments | |

Unit description of module 12: Smart Sensor Network Systems – Project

| | |
|---|--|
| Unit title | Smart Sensor Network Systems – Project |
| Code | |
| Module title | Smart Sensor Network Systems |
| Unit contents | <ul style="list-style-type: none"> • Introduction to measuring technology for computer scientists • Data acquisition basics • The measuring chain • Data acquisition challenges and error propagation • Intelligent sensor concepts • Wireless sensor networks (WSN) • WSN operating systems • Real-time aspects of WSNs <ul style="list-style-type: none"> • Signal analysis basics |
| Teaching methods | R&D project with small groups (4 students max.) |
| Semester periods (hours) per week | 4 |
| Workload (h) | 150h |
| Class hours | 60h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 80h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Matthias Wagner |
| Recommended reading | <p>Holger Karl, Andreas Willig: Protocols and Architectures for Wireless Sensor Networks, Wiley, 2005</p> <p>Current literature, e.g. research papers, will be announced at the begin of the semester</p> |
| Assessment type and form of | None |

| | |
|--------------------|------|
| Assessment grading | None |
| Comments | |

Modul 13 Introductory Data Analysis

| | | |
|---------------------------------|---|----|
| Module title | Introductory Data Analysis | PO |
| Module number | 13 | PO |
| Module code | IDA | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | Basic skills in statistics as they are offered in the Bachelor program Informatik, i.e., students should be able to perform the most important methods of inferential statistics in line with some real-world problems, the students should be able to interpret and assess the results of basic statistical methods | PO |
| Module prerequisites | None | |
| Module examination requirements | Solutions to at least 40% of weekly exercises in unit Introductory Data Analysis – Exercises; short written exposé as stated in unit Introductory Data Analysis (total effort: 35 hours) | PO |
| Module examination | Written (computer) examination of 90 minutes duration | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • Confident assessment of the usage of the various methods of univariate and bivariate statistics in the application context. • Knowledge and understanding of different probability concepts (distributions, statistical models, testing procedures and principles) • Capacity to apply methods to selected real world situations • Capacity to use the computer to solve problems in real world situations • Capacity to understand and judge results of statistical analysis • Awareness of dangers of misuse and misinterpretation • Capacity to communicate using statistical language, i.e., explain procedures, results of an analysis and a critique of the results <p>Non specialist competencies:</p> <ul style="list-style-type: none"> • Scientific work style | PO |

| | | |
|-------------------------|--|----|
| Module contents | Introductory Data Analysis – Lectures, Introductory Data Analysis – Exercises | PO |
| Module teaching methods | Lectures using multimedia presentation techniques Exercises on PC using spreadsheets and statistical software tool | PO |
| Module language | English | PO |
| Module availability | Winter semester | PO |
| Module coordination | Prof. Dr. Egbert Falkenberg | |
| Comments | | |

Unit description of module 13: Introductory Data Analysis – Lectures

| | |
|---|---|
| Unit title | Introductory Data Analysis – Lectures |
| Code | |
| Module title | Introductory Data Analysis (13) |
| Unit contents | <ul style="list-style-type: none"> • Descriptive statistics (characterics and plots, univariate and bivariate methods) • Probabilty concepts and theory (Baysian and frequentist approach, formulating of statistical models) • Inferential statistics (concepts and a selection of tests) • Some Test theory (assumptions, hypotheses, OC, alpha/beta error) • Performing Statistical Tests (Checking Assumptions. preparing the data, understanding results, discussing results) • Performing Statistical Tests (Examples of applications) • non-paramtric tools • Common Errors (how not to proceed) |
| Teaching methods | Interactive group lecturing |
| Semester periods (hours) per week | 2 h |
| Workload (h) | 70 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30 h |
| Total time of practical training (h) | 0 h |
| Unit language | English |

| | |
|-----------------------------|---|
| Lecturer | Prof. Dr. Christina Andersson, Prof. Dr. Behl, Prof. Dr. Egbert Falkenberg, Prof. Dr. Andreas Orth |
| Recommended reading | Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them) |
| Assessment type and form of | Prerequisite: Successful participation in Unit: Introductory Data Analysis – Exercises, Written (computer) examination of 90 minutes duration |
| Assessment grading | Graded according to published grading scheme |
| Comments | |

Unit description of module 13: Introductory Data Analysis – Exercises

| | |
|---|--|
| Unit title | Introductory Data Analysis – Exercises |
| Code | |
| Module title | Introductory Data Analysis (13) |
| Unit contents | <ul style="list-style-type: none"> • Computer Exercises pertaining to the contents described in the unit Introductory Data Analysis – Lectures • short written exposé of one real world problem, including reasoning on why which methods were selected, including interpretation and critique of results obtained • lesson’s learned session after written exposé • exam preparation session for the Module examination |
| Teaching methods | Using PC in Computer pool to solve problems |
| Semester periods (hours) per week | 2 h |
| Workload (h) | 80 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | 0 h |
| Total time of individual study (h) | 50 h |
| Total time of practical training (h) | 30 h |
| Unit language | English |
| Lecturer | Prof. Dr. Christina Andersson, Prof. Dr. Behl, Prof. Dr. Egbert Falkenberg, Prof. Dr. Andreas Orth |

| | |
|-----------------------------|--|
| Recommended reading | Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them) |
| Assessment type and form of | 50% Regular attendance at exercise groups, (unit Data Mining Methods – Exercises), solutions to 40% of weekly exercises in unit Data Mining Methods – Exercises, Compilation of a short written exposé |
| Assessment grading | Not graded |
| Comments | |

Modul 14 Data Mining Methods

| | | |
|---------------------------------|--|----|
| Module title | Data Mining Methods | PO |
| Module number | 14 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 2nd Semester | PO |
| Module type | Compulsory Elective Module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | Basic skills in statistics as they are offered in the Bachelor program Informatik, i.e., students should be able to perform the most important methods of inferential statistics in line with some real-world problems, the students should be able to interpret and assess the results of basic statistical methods | PO |
| Module prerequisites | None | |
| Module examination requirements | Software Exercises with documentation (processing time 80 hours) | PO |
| Module examination | Written computer-based examination (90 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • Awareness of different data types, data scales, data use as endogenous and exogenous • Skills in data recovery and data pre-processing • Theoretical understanding of statistical methods for information extraction • Capacity to use the computer to solve problems in real world data mining problems • Capacity to understand and judge results of statistical analysis in the context of data mining • Awareness of dangers of misuse and misinterpretation • Capacity to communicate using statistical language, i.e. explain procedures, results of an analysis and a critique of the results • Communication in international teams | PO |
| Module contents | Data Mining Methods – Lectures Data Mining Methods – Exercises | PO |
| Module teaching methods | Lectures using multimedia presentation techniques Exercises with a PC and statistical programming language in Computer pool to solve problems | PO |

| | | |
|---------------------|------------------------|----|
| Module language | English | PO |
| Module availability | Summer semester | PO |
| Module coordination | Prof. Dr. Andreas Orth | |
| Comments | | |

Unit description of module 14: Data Mining Methods – Lectures

| | |
|---|---|
| Unit title | Data Mining Methods – Lectures |
| Code | |
| Module title | Data Mining Methods |
| Unit contents | <ul style="list-style-type: none"> • Introduction to Data Mining (data types, data scales, roles of variables in an analysis, methods pertaining to different scales and types of variables, the data mining workflow, Loss functions) • Introduction to a statistical programming language (alternatively: R (or S-plus), SAS, SPSS, etc.) • Theory behind important methods data mining and inference selection out of <ul style="list-style-type: none"> ○ linear modelling – GLM, GLIM, mixed effects modeling, variable selection methods - , ○ Methods for Classification – prototype-methods, k-nearest neighbour classifiers, Linear Discriminant Analysis, logistic regression, separating hyperplanes, support vector machines etc. ○ Additive Models, Trees, Boosting Methods, Additive Trees ○ Neural nets ○ Unsupervised Learning • Variance Estimation and Validation methods (selection out of bootstrapping, jackknifing, cross-validation, Bayesian methods, EM-algorithm, MCMC) |
| Teaching methods | Interactive group lecturing |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30h |

| | |
|--------------------------------------|--|
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth |
| Recommended reading | <ul style="list-style-type: none"> • SAS – Online Documentation • R project-Dokumentation • Hastie, Tibshirani & Friedman: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer (2001) • Berthold & Hand: Intelligent Data Analysis: An Introduction. (1999) • John Fox: Applied Regression Analysis and Generalized Linear Models. Sage Publications (1998) • Efron, B; Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman&Hall/CRC (1993) <p>Christopher Bishop: Neural Networks for Pattern Recognition (1995)</p> |
| Assessment type and form of | None |
| Assessment grading | None |
| Comments | |

Unit description of module 14: Data Mining Methods – Exercises

| | |
|---|---|
| Unit title | Data Mining Methods – Exercises |
| Code | |
| Module title | Data Mining Methods |
| Unit contents | <ul style="list-style-type: none"> • Computer Exercises pertaining to the contents described in the unit Data Mining Methods – lectures • short written exposé of one real world problem, including reasoning on why which methods were selected, including interpretation and critique of results obtained • lesson's learned session after written exposé <ul style="list-style-type: none"> • exam preparation session for the Module examination |
| Teaching methods | Using PC and statistical programming language in Computer pool to solve problems |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |

| | |
|--------------------------------------|--|
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth |
| Recommended reading | <ul style="list-style-type: none"> • Literature as in Data Mining Methods – Lectures, in addition: • Davison, A.C.; Hinkley, D.V.: Bootstrap Methods and their Applications. Cambridge University Press (1997) • C. R. Robert, G. Casella: Introducing Monte Carlo Methods with R. Springer (2010) • John Fox, Stanford Weisberg: An R companion to Applied Regression. Sage Publications (2011) <p>Data Mining Group (2011): http://www.dmg.org/ (Zugriff 11.8. 2011)</p> |
| Assessment type and form | Software Exercises with documentation (processing time 80 hours) |
| Assessment grading | Pass/fail |
| Comments | n/a |

Modul 15 Simulation Methods

| | | |
|---------------------------------|---|----|
| Module title | Simulation Methods | PO |
| Module number | 15 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 3rd Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | Good knowledge in discrete mathematics, calculus, numerical methods, contents of module System Theory and Modeling | PO |
| Module prerequisites | None | |
| Module examination requirements | Laboratory exercises (total processing time 80 hours) | PO |
| Module examination | Written examination (90 minutes) | PO |
| Learning outcomes and skills | <p>Upon completion of this course, the student is able to:</p> <ul style="list-style-type: none"> • assess the growing importance of simulation for high-integrity systems, • understand the interaction between simulation and experimental verification, • get an overview over simulation methods, • get experience in using simulation tools, recognize the limitations of simulation work. | PO |
| Module contents | <p>Simulation Methods – Lectures</p> <p>Simulation Methods – Exercises</p> | PO |
| Module teaching methods | <p>Interactive lectures using multimedia presentation techniques</p> <p>Exercises: Teamwork</p> | PO |
| Module language | English | PO |
| Module availability | Summer semester | PO |
| Module coordination | Prof. Dr. Doina Logofatu | |
| Comments | | |

Unit description of module 15: Simulation Methods – Lectures

| | |
|---|--|
| Unit title | Simulation Methods – Lectures |
| Code | |
| Module title | Simulation Methods |
| Unit contents | <ul style="list-style-type: none"> • Methods <ul style="list-style-type: none"> Approximation techniques Types and categories Software tools Numerical methods Visualization • Validation <ul style="list-style-type: none"> Simulation and Measurement • Applications |
| Teaching methods | Interactive group learning |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10h |
| Total time of individual study (h) | 30h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner |
| Recommended reading | <ul style="list-style-type: none"> • L.v. Bertalanffy: General System Theory, George Braziller Inc., New York, 1968 • H. Anton, Calculus, A new horizon, Sixth Edition, John Wiley and Sons, New York, 1999 • B. P. Zeigler et al.: Theory of Modeling and Simulation, 2nd ed. Academic Press, 2000 • A. B. Shiflet; G. W. Shiflet: Introduction to Computational Science. Princeton University Press, 2006 <p>Current literature, e.g. journal papers, conference proceedings etc., will be announced at the beginning of the semester</p> |
| Assessment type and form | None |

| | |
|--------------------|------|
| Assessment grading | None |
| Comments | |

Unit description of module 15: Simulation Methods – Exercises

| | |
|---|--|
| Unit title | Simulation Methods – Exercises |
| Code | |
| Module title | Simulation Methods |
| Unit contents | <ul style="list-style-type: none"> • Lab exercises with software tools pertaining to the contents described in the unit Simulation Methods – lectures <ul style="list-style-type: none"> • Lessons learned session after group work |
| Teaching methods | Teamwork in small R&D groups |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner |
| Recommended reading | See Unit Simulation Methods – Lectures |
| Assessment type and form | Lab exercises (processing time 80 hours) |
| Assessment grading | Pass/fail |
| Comments | |

Modul 16 Advanced Data Structures and Algorithms

| | | |
|---------------------------------|--|----|
| Module title | Advanced Data Structures and Algorithms | PO |
| Module number | 16 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 3rd semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP/150 h | PO |
| Recommended previous knowledge | Good knowledge of fundamentals of data structures and algorithms, programming (e.g. C, C++ or Java) and mathematics are recommended | PO |
| Module prerequisites | None | |
| Module examination requirements | Solutions to at least 40% of weekly exercises in unit Advanced Data Structures and Algorithms – Exercises; short written exposé as stated in unit Advanced Data Structures and Algorithms – Exercises (total effort: 35 hours) | PO |
| Module examination | Written report in the form of a scientific contribution (processing time 6 weeks) and oral presentation of the results in the form of an event talk according to the rules of a scientific society, i.e., Springer, ACM, IEEE (min. 10 and max. 20 minutes). | PO |
| Learning outcomes and skills | <p>Upon the completion of this module, the student is able to:</p> <ul style="list-style-type: none"> • analyze the complexity of data structures • recognize, choose and appropriate use advanced data structures • analyze the complexity of algorithms • deal with selected advanced algorithms, especially from the area of nature- and bio-inspired algorithms • compare the efficiency/optimality of different algorithms • implement and compare different approaches for a given real application • deliver practical oriented solutions • perform statistical tests <p>Training for non-specialist competencies. Students:</p> <ul style="list-style-type: none"> • practice the scientific project management • communicate and work in team • research and write a scientific text • present their results in a scientific colloquium | PO |
| Module contents | Advanced Data Structures and Algorithms – Lectures, Advanced Data Structures and Algorithms – Exercises | PO |

| | | |
|-------------------------|--|----|
| Module teaching methods | Interactive group lecturing with exercises | PO |
| Module language | English | PO |
| Module availability | Summer semester | PO |
| Module coordination | Prof. Dr. Doina Logofatu | |
| Comments | | |

Unit description of module 16: Advanced Data Structures and Algorithms – Lectures

| | |
|-----------------------------------|---|
| Unit title | Advanced Data Structures and Algorithms – Lectures |
| Code | |
| Module title | Advanced Data Structures and Algorithms (16) |
| Unit contents | <p>The course will cover selected topics form the following areas. The depth of coverage might vary.</p> <p>Advanced Data Structures and their Complexity:</p> <ul style="list-style-type: none"> • Dynamic Graphs • Double-Ended Priority Queues • Binomial, Pairing and Fibonacci Heaps • Leftist, AVL, Red-Black, B+,B*, Splay, Priority Search, Segment, Multidimensional, Quad Trees • Compressed Binary, High Order Tries <p>Advanced Algorithms and related innovative Applications:</p> <ul style="list-style-type: none"> • Genetic Algorithms • Ant Colony and Swarm Intelligence Optimization • Genetic Programming • Quantum Computing and Quantum Annealing • Distributed/Parallel Evolutionary Algorithms • Other Nature- and Bio-Inspired Methods • Hybrid Evolutionary Algorithms • Memetic and Cloud Computing • Fuzzy Logic • Chaos Theory • Machine Learning <p>The emphasis must be on examples for systems and applications in practice.</p> |
| Teaching methods | Interactive group lecturing |
| Semester periods (hours) per week | 2 h |
| Workload (h) | 70 h |
| Class hours | 30 h |

| | |
|---|--|
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30 h |
| Total time of practical training (h) | 0 h |
| Unit language | English |
| Lecturer | Prof. Dr. Doina Logofatu |
| Recommended reading | <p>David E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison.Wesley, 1989</p> <p>John R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems), A. Bradford Book, 1992</p> <p>Dario Floreano, Claudio Matussi, Bio-Inspired Intelligence: Theories, Methods, and Technologies, The MIT Press, 2008</p> <p>Steven H. Strogatz, Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering, Westview Press, 2nd edition, 2014</p> <p>El-Ghazali Talbi, Metaheuristics: From Design to Implementation, Wiley, 2009</p> <p>Vijay V. Vazirani, Approximation Algorithms, Springer, 2004</p> <p>Stephen Boyd, Lieven Vandenbeghe, Convex Optimization, Cambridge University Press, 2004</p> <p>Wan Fokkink, Distributed Algorithms: An intuitive Approach, The MIT Press, 2013</p> <p>Calos A. Varela, Programming Distributed Computing Systems: A Foundational Approach, The MIT Press, 2013</p> |
| Assessment type and form of | Unit Advanced Data Structures and Algorithms – Exercises, 50% of end grade, Written examination of 90 minutes duration at the end of the semester – 50 % of the end grade |
| Assessment grading | Graded according to published grading scheme |
| Comments | Prerequisite: Successful participation in unit Advanced Data Structures and Algorithms – Exercises, 50% of end grade |

Unit description of module 16: Advanced Data Structures and Algorithms – Exercises

| | |
|---------------|--|
| Unit title | Advanced Data Structures and Algorithms – Exercises |
| Code | |
| Module title | Advanced Data Structures and Algorithms (16) |
| Unit contents | Realization, analyze and enhancement of applications and systems |

| | |
|---|---|
| Teaching methods | Teamwork in small groups (max. 4 students) |
| Semester periods (hours) per week | 2 h |
| Workload (h) | 80 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | 0 h |
| Total time of individual study (h) | 50 h |
| Total time of practical training (h) | 30 h |
| Unit language | English |
| Lecturer | Prof. Dr. Doina Logofatu |
| Recommended reading | See unit Advanced Data Structures and Algorithms – Lectures |
| Assessment type and form of | Written report in the form of a scientific contribution and oral presentation of the results in the form of an event talk according to the rules of a scientific society, i.e., Springer, ACM, IEEE (max 20 minutes). |
| Assessment grading | Graded according to published grading scheme |
| Comments | |

Modul 17 Pattern Oriented Software Architecture

| | | |
|---------------------------------|---|----|
| Module title | Pattern Oriented Software Architecture | PO |
| Module number | 17 | PO |
| Module code | POS | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | one semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory elective Module | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | Good knowledge in principles and procedures of software engineering, programming skills in object-oriented programming languages | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Oral examination (min. 15, max. 45 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • understand the motives of the pattern community. • distinguish between different types of patterns. • apply patterns in the design of SCS. • assess new developments of pattern catalogs and languages. • team work: Students acquire the ability to work with others toward a same goal, participating actively, sharing responsibility and rewards, and contributing to the capability of the teamwork. • communication in international teams | PO |
| Module contents | Pattern Oriented Software Architecture – Lectures Pattern Oriented Software Architecture – Exercises | PO |
| Module teaching methods | Interactive lectures Laboratory exercises in teamwork | PO |
| Module language | English | PO |
| Module availability | Winter semester | PO |
| Module coordination | Prof. Dr. Eicke Godehardt | |

| | |
|----------|--|
| Comments | |
|----------|--|

Unit description of module 17: Pattern Oriented Software Architecture – Lectures

| | |
|---|---|
| Unit title | Pattern Oriented Software Architecture - Lectures |
| Code | |
| Module title | Pattern Oriented Software Architecture |
| Unit contents | <ul style="list-style-type: none"> • Software architecture • Origins of the pattern movement • Principles and Practices of Modern Software Development and the prominent Role of Patterns • Pattern-oriented software architecture: Architectural patterns, Design patterns, • Idioms • Application-specific pattern systems • Patterns for software testing • Pattern languages <ul style="list-style-type: none"> • Alternatives, e.g. Frameworks |
| Teaching methods | Interactive Group Lecturing |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30 h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer |
| Recommended reading | <ul style="list-style-type: none"> • E. Gamma et. Al: Design Patterns Addison-Wesley, 1998 Buschmann et al: Pattern Oriented Software Architecture. Addison Wesley 1996 |
| Assessment type and form | None |
| Assessment grading | None |
| Comments | n/a |

Unit description of module 17: Pattern Oriented Software Architecture – Exercises

| | |
|------------|--|
| Unit title | Pattern Oriented Software Architecture - Exercises |
| Code | |

| | |
|---|---|
| Module title | Pattern Oriented Software Architecture |
| Unit contents | <ul style="list-style-type: none"> • Lab exercises pertaining to the contents described in the unit Pattern Oriented Software Architecture – lectures <ul style="list-style-type: none"> • Lessons learned session after solved problems |
| Teaching methods | Teamwork in small R&D groups, exercise tasks |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer |
| Recommended reading | See Unit Pattern-Oriented Software Architecture - Lectures |
| Assessment type and form | None |
| Assessment grading | None |
| Comments | |

Modul 18 Cloud Computing

| | | |
|---------------------------------|---|----|
| Module title | Cloud Computing | PO |
| Module number | 18 | PO |
| Module code | Cloud | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP /150 h | PO |
| Recommended previous knowledge | Good knowledge in software engineering, computer networks, databases and distributed applications and one high-level programming language. | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written assignment (processing time 6 weeks) with presentation (min. 15, max. 30 minutes) | PO |
| Learning outcomes and skills | <p>Upon completion of this course, the students</p> <ul style="list-style-type: none"> • understand the concepts and technologies fundamental for Cloud Computing • understand the economical and operational impact of Cloud Computing for providing IT-resources within the enterprise • is able to apply a structured, scientific process to evaluate architecture alternatives for Cloud Computing • are able to architect and implement Cloud Computing solutions. | PO |
| Module contents | <p>Cloud Computing – Lectures</p> <p>Cloud Computing – Exercises</p> | PO |
| Module teaching methods | <p>Lectures</p> <p>Exercises</p> | PO |
| Module language | English | PO |
| Module availability | Each semester | PO |

| | |
|---------------------|--------------------------|
| Module coordination | Prof. Dr. Christian Baun |
| Comments | |

Unit description of module 18: Cloud Computing – Lectures

| | |
|---|---|
| Unit title | Cloud Computing – Lectures |
| Code | |
| Module title | Cloud Computing |
| Unit contents | <p>The course will cover selected subjects from the following areas. The depth of coverage might vary:</p> <ul style="list-style-type: none"> • Definitions of Cloud Computing and Core Foundations of Cloud Computing • Virtualization technologies • SOA and Web-Services • Different Cloud Computing architectures (SaaS, PaaS, IaaS) • Different Cloud Computing vendor stacks including open source • Service Management for the cloud • Algorithms for Cloud Computing (e.g. MapReduce) • Security aspects of Cloud Computing <p>Operational aspects of Cloud Computing Economical aspects of Cloud Computing</p> |
| Teaching methods | Lectures |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30 h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes, Prof. Dr. Jörg Schäfer |
| Recommended reading | <p>Christian Baun, Marcel Kunze, Jens Nimis and Stefan Tai, Cloud Computing: Web-Based Dynamic IT Services, Springer (2011).</p> <p>Tom White, Hadoop: The Definitive Guide (Englisch) Taschenbuch, O'Reilly, 2015.</p> <p>Thomas Erl, Zaigham Mahmood, Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, Prentice Hall, 2013.</p> |

| | |
|--------------------------|---|
| | Nick Antonopoulos and Lee Gillam: Cloud Computing: Principles, Systems and Applications, Springer, 2010. Current literature recommendations will be given at the semester start. |
| Assessment type and form | None |
| Assessment grading | None |
| Comments | n/a |

Unit description of module 18: Cloud Computing – Exercises

| | |
|---|--|
| Unit title | Cloud Computing – Exercises |
| Code | |
| Module title | Cloud Computing |
| Unit contents | While the lectures provide the theoretical background, the exercises will enable students to apply the concepts. The students will read current research literature and vendor documentation and configure examples accordingly. In addition, simple prototypes will be developed. Henceforth, the students will receive continuous feedback, which will support the educational objectives. |
| Teaching methods | Exercises |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50 h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | See lecturers of Unit Cloud Computing – Lectures |
| Recommended reading | n/a |
| Assessment type and form | |
| Assessment grading | |
| Comments | n/a |

Modul 19 Advanced Distributed Systems

| | | |
|---------------------------------|---|----|
| Module title | Advanced Distributed Systems | PO |
| Module number | 19 | PO |
| Module code | ADS | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | <ul style="list-style-type: none"> • Knowledge of networking principles, basic knowledge of distributed applications, as well as programming skills in object-oriented programming languages. • This corresponds to the following modules of the Bachelor program Computer Science (Informatik): <ul style="list-style-type: none"> • Rechnernetze • Verteilte Anwendungen • Objektorientierte Programmierung • OOP-Vertiefung | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written examination (90 minutes) | PO |
| Learning outcomes and skills | <ul style="list-style-type: none"> • Understand the advantages and problems of distributed systems. • Knowledge of different distributed architectures and algorithms. • Ability to analyze distributed systems, in particular with respect to robustness. | PO |
| Module contents | Advanced Distributed Systems – Lectures Advanced Distributed Systems – Exercises | PO |
| Module teaching methods | Lectures: Interactive group lecturing Exercises: Teamwork in small groups | PO |
| Module language | English | PO |

| | | |
|---------------------|-----------------------------|----|
| Module availability | Summer semester | PO |
| Module coordination | Prof. Dr. Justus Klingemann | |
| Comments | | |

Unit description of module 19: Advanced Distributed Systems – Lectures

| | | |
|---|--|--|
| Unit title | Advanced Distributed Systems – Lectures | |
| Code | | |
| Module title | Advanced Distributed Systems | |
| Unit contents | <ul style="list-style-type: none"> • Properties of distributed systems • Time and synchronization • Distributed algorithms • Middleware for distributed systems • Consistency and replication | |
| Teaching methods | Interactive Group Lecturing | |
| Semester periods (hours) per week | 2 | |
| Workload (h) | 70h | |
| Class hours | 30h | |
| Total time of examination incl. preparation (h) | 10h | |
| Total time of individual study (h) | 30h | |
| Total time of practical training (h) | | |
| Unit language | English | |
| Lecturer | Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer | |
| Recommended reading | <p>G. Coulouris, J. Dollimore, T. Kindberg: Distributed Systems: Concepts and Design, Addison-Wesley</p> <p>A. Tanenbaum, M. van Steen: Distributed Systems: Principles and Paradigms, Prentice-Hall</p> | |
| Assessment type and form | | |
| Assessment grading | | |
| Comments | | |

Unit description of module 19: Advanced Distributed Systems – Exercises

| | |
|---|---|
| Unit title | Advanced Distributed Systems - Exercises |
| Code | |
| Module title | Advanced Distributed Systems |
| Unit contents | <ul style="list-style-type: none"> • Lab exercises with software tools pertaining to the contents described in the unit Distributed Systems – lectures • Practical teamwork on real world problems Lessons learned session after group work |
| Teaching methods | Teamwork in small software development groups |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer |
| Recommended reading | See Unit Advanced Distributed Systems – Lectures |
| Assessment type and form | |
| Assessment grading | |
| Comments | |

Modul 20 Advanced Formal Modelling

| | | |
|---------------------------------|---|----|
| Module title | Advanced Formal Modelling | PO |
| Module number | 20 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | Other Computer Science Master programs | PO |
| Module duration | One semester | PO |
| Recommended semester | 1st Semester | PO |
| Module type | Compulsory module | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | Basic knowledge of propositional and first order logic | PO |
| Module prerequisites | None | |
| Module examination requirements | Laboratory exercises with written assignment (processing time: 80 hours) | PO |
| Module examination | Written computer-based examination (90 minutes) | PO |
| Learning outcomes and skills | <p>The module provides an insight into the theory and practice of formal modeling. It focuses on the role of logic in deductive software verification and in knowledge representation. In this module the students learn how to model formal properties of software and prove its correctness using logic. They learn how to model knowledge using logic-based knowledge representation formalisms. Main goals are:</p> <ul style="list-style-type: none"> • Understanding the use of different logics in formal modeling. • Understanding the logical foundations of formal methods and logic-based knowledge representation formalisms. • Ability to prove correctness of simple code fragments. • Ability to formalize and reason using logic. • Obtaining practical skills in using a theorem prover and a formal verification tool. • Understanding the limitations of logic. • Non specialist competencies: Scientific working style | PO |
| Module contents | Advanced Formal Modeling – Lectures Advanced Formal Modeling – Exercises | PO |
| Module teaching methods | Lectures Exercises | PO |

| | | |
|---------------------|--------------------------|----|
| Module language | English | PO |
| Module availability | Winter semester | PO |
| Module coordination | Prof. Dr. Barjs Sertkaya | |
| Comments | | |

Unit description of module 20: Advanced Formal Modeling – Lectures

| | |
|---|---|
| Unit title | Advanced Formal Modeling - Lectures |
| Code | |
| Module title | Advanced Formal Modeling |
| Unit contents | <p>The lectures cover the theoretical aspects and provide examples mainly on the following topics:</p> <ul style="list-style-type: none"> - Mathematical and logical foundations – Sets, relations and functions - Propositional Logic, First Order Logic - Verification of sequential programs Verification of Java programs using Java Modeling Language - Logic-based knowledge representation and reasoning - Description Logics, Web Ontology Language OWL |
| Teaching methods | Lecture |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Ruth Schorr, Prof. Dr. Barjs Sertkaya |
| Recommended reading | <ul style="list-style-type: none"> • Logic in Computer Science Modelling and Reasoning about Systems. M. Huth, M. Ryan. 2004 (2nd Edition). ISBN: 9780521543101 • Understanding Formal Methods. J.-F. Monin. Springer, 2003. ISBN-10: 1852332476 • Deductive Software Verification The KeY Book. From Theory to Practice. W. |

| | |
|--------------------------|--|
| | Ahrendt, B. Beckert, R. Bubel, R. Hhnle, P. H. Schmitt, M. Ulbrich. ISBN 978-3-319-49811-9 The Description Logic Handbook: Theory, Implementation, and Applications. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. F. Patel-Schneider. Cambridge University Press, 2003. |
| Assessment type and form | None |
| Assessment grading | None |
| Comments | |

Unit description of module 20: Advanced Formal Modeling – Exercises

| | |
|---|--|
| Unit title | Advanced Formal Modeling - Exercises |
| Code | |
| Module title | Advanced Formal Modeling |
| Unit contents | The exercises <ul style="list-style-type: none"> • deepen the knowledge for the theoretical foundations of modeling and proving with logic, and • help to gain practical experience with theorem provers and formal verification tools. |
| Teaching methods | Exercises |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | n/a |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Ruth Schorr, Prof. Dr. Barjs Sertkaya |
| Recommended reading | See unit Advanced Formal Modeling – Lectures |
| Assessment type and form | Lab exercises with written assignment (processing time 80 hours) |
| Assessment grading | Pass/fail |

| | |
|----------|--|
| Comments | |
|----------|--|

Modul 21 Formal Specification and Verification

| | | |
|---------------------------------|--|----|
| Module title | Formal Specification and Verification | PO |
| Module number | 21 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 3rd Semester | PO |
| Module type | Compulsory elective module | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | <ul style="list-style-type: none"> • Basic knowledge of propositional and predicate logic • Basic knowledge of algorithm design and analysis • Basic knowledge of automata theory | PO |
| Module prerequisites | None | |
| Module examination requirements | Laboratory exercises with written assignment (Processing time: 80 hours) | PO |
| Module examination | Written computer-based examination (90 minutes) | PO |
| Learning outcomes and skills | <p>Understanding the principles of formal specification and verification.</p> <p>Understanding the theory (models and logics) used in model checking.</p> <p>Reasoning about safety, liveness and fairness properties in concurrent systems.</p> <p>Specifying safety and liveness properties of concurrent systems using temporal logic and/or computational tree logic.</p> <p>Verifying that a concurrent system satisfies certain safety and liveness properties using model checking algorithms.</p> <p>Obtaining practical skills in using a Model Checking Tool .</p> <p>Understanding the limitations of model checking.</p> <p>Non specialist competencies:</p> <p style="padding-left: 40px;">Communication in international teams</p> | PO |
| Module contents | <p>Formal Specification and Verification – Lectures</p> <p>Formal Specification and Verification – Exercises</p> | PO |

| | | |
|-------------------------|-----------------------|----|
| Module teaching methods | Lectures Exercises | PO |
| Module language | English | PO |
| Module availability | Each semester | PO |
| Module coordination | Prof. Dr. Ruth Schorr | |
| Comments | | |

Unit description of module 21: Formal Specification and Verification – Lectures

| | | |
|---|---|--|
| Unit title | Formal Specification and Verification – Lectures | |
| Code | | |
| Module title | Formal Specification and Verification | |
| Unit contents | <p>The lectures provide an introduction to the main principles of model checking:</p> <ul style="list-style-type: none"> • Modeling reactive systems by transition systems • Linear time properties and Büchi automata • Linear temporal logic and automata-based model checking <p>Computation tree logic</p> | |
| Teaching methods | Lectures | |
| Semester periods (hours) per week | 2 | |
| Workload (h) | 70 h | |
| Class hours | 30 h | |
| Total time of examination incl. preparation (h) | 10 h | |
| Total time of individual study (h) | 30 h | |
| Total time of practical training (h) | | |
| Unit language | English | |
| Lecturer | Prof. Dr. Ruth Schorr | |
| Recommended reading | <p>Baier, Christel and Katoen, Joost-Pieter: Principles of Model Checking MIT Press, 2008.</p> <p>Ben-Ari, Mordechai: <i>Principles of the Spin Model Checker</i>. Springer, 2008.</p> | |
| Assessment type and form | | |
| Assessment grading | | |

| | |
|----------|--|
| Comments | |
|----------|--|

Unit description of module 21: Formal Specification and Verification – Exercises

| | |
|---|---|
| Unit title | Formal Specification and Verification –Exercises |
| Code | |
| Module title | Formal Specification and Verification |
| Unit contents | <p>The exercises</p> <ul style="list-style-type: none"> • deepen the knowledge for the theoretical foundations of model checking and • provide an introduction to model checkers, i.e. SPIN, NuSMV or nuXmv, with practical exercises |
| Teaching methods | Exercises |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80 h |
| Class hours | 30 h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50 h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Ruth Schorr |
| Recommended reading | see Unit Formal Specification and Verification – Lectures |
| Assessment type and form | Laboratory exercises with written assignment (Processing time: 80 hours) |
| Assessment grading | |
| Comments | |

Modul 22 Mathematics Update

| | | |
|---------------------------------|---|----|
| Module title | Mathematics Update | PO |
| Module number | 22 | PO |
| Module code | Math | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 2nd Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | Undergraduate level in mathematics | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written examination (90 minutes) | PO |
| Learning outcomes and skills | Analyze mathematical problems in a software project's list of requirements. To familiarize with new mathematical fields. Assess the suitability and usability of mathematical software tools. | PO |
| Module contents | Mathematics Update – Lectures Mathematics Update – Exercises | PO |
| Module teaching methods | Interactive lectures Exercises with teamwork in small groups | PO |
| Module language | English | PO |
| Module availability | Each semester | PO |
| Module coordination | Prof. Dr. Doina Logofatu | |
| Comments | | |

Unit description of module 22: Mathematics Update – Lectures

| | |
|------------|-------------------------------|
| Unit title | Mathematics Update - Lectures |
|------------|-------------------------------|

| | |
|---|--|
| Code | |
| Module title | Mathematics Update |
| Unit contents | <ul style="list-style-type: none"> • Linear Algebra • Geometry • Discrete Mathematics • Calculus • Scientific Computing |
| Teaching methods | Interactive group lecturing |
| Semester periods (hours) per week | 2 |
| Workload (h) | 70h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | 10 h |
| Total time of individual study (h) | 30 h |
| Total time of practical training (h) | |
| Unit language | English |
| Lecturer | Prof. Dr. Jörg Schäfer, Prof. Dr. Doina Logofatu, Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr |
| Recommended reading | <ul style="list-style-type: none"> • H. Anton, Calculus, A new horizon, Sixth Edition, John Wiley and Sons, New York, 1999; • H. Anton, Elementary Linear Algebra, John Wiley and Sons, New York, 1994; • J. Stewart, Calculus, Cengage Learning Emea; Auflage: 7th Revised edition, 2011; • Scilab/Matlab on-line literature • Press et al.: Numerical Recipes. Cambridge University Press, 2007 |
| Assessment type and form | |
| Assessment grading | |
| Comments | n/a |

Unit description of module 22: Mathematics Update – Exercises

| | |
|---------------|---|
| Unit title | Mathematics Update – Exercises |
| Code | |
| Module title | Mathematics Update |
| Unit contents | <ul style="list-style-type: none"> • Computer Exercises pertaining to the contents described in the unit Mathematics Update – Lectures |

| | |
|---|--|
| | <ul style="list-style-type: none"> Lessons learned session |
| Teaching methods | Teamwork in small R&D groups, exercise tasks |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Prof. Dr. Jörg Schäfer, Prof. Dr. Doina Logofatu, Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr |
| Recommended reading | see Unit Mathematics Update – Lectures |
| Assessment type and form | None |
| Assessment grading | None |
| Comments | |

Modul 23 Advanced Testing Methods

| | | |
|---------------------------------|--|----|
| Module title | Advanced Testing Methods | PO |
| Module number | 23 | PO |
| Module code | Test | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | High Integrity Systems (M.Sc.) | PO |
| Module duration | One semester | PO |
| Recommended semester | 2nd Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | Extended knowledge in software engineering, very good programming skills in procedural and object-oriented programming languages | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Written examination (90 minutes) | PO |
| Learning outcomes and skills | <p>Upon completion of this course, the student is able to:</p> <ul style="list-style-type: none"> • assess different testing methodologies, • master various powerful testing procedures, • differentiate between the testing of procedural and object oriented software, • estimate the importance of safety criteria for test case design, • recognize the limits of testing capabilities, • use gained experience to select valuable automated tests, • recognize tests not to be automated. <p>This module facilitates communication structures used in business like Wikis and Discussion boards to show challenges working in global teams.</p> | PO |
| Module contents | <p>Advanced Testing Methods – Lectures</p> <p>Advanced Testing Methods – Exercises</p> | PO |
| Module teaching methods | <p>Lectures: Interactive group lecturing</p> <p>Exercises: Teamwork in small groups</p> | PO |
| Module language | English | PO |

| | | |
|---------------------|--|----|
| Module availability | Summer semester | PO |
| Module coordination | Dr. Torsten Schönfelder (Deutsche Lufthansa) | |
| Comments | | |

Unit description of module 23: Advanced Testing Methods – Lectures

| | | |
|---|--|--|
| Unit title | Advanced Testing Methods – Lectures | |
| Code | | |
| Module title | Advanced Testing Methods – Lectures | |
| Unit contents | <ul style="list-style-type: none"> • Planning for verification and validation • Design for testability • Testing strategies • Testing procedures • Testing of object-oriented systems • Testing patterns • Testing of and with safety criteria • Environment simulation • Testing tools | |
| Teaching methods | Interactive group lecturing | |
| Semester periods (hours) per week | 2 | |
| Workload (h) | 70h | |
| Class hours | 30h | |
| Total time of examination incl. preparation (h) | 10 h | |
| Total time of individual study (h) | 30h | |
| Total time of practical training (h) | | |
| Unit language | English | |
| Lecturer | Dr. Torsten Schönfelder, Deutsche Lufthansa | |
| Recommended reading | Current Software Engineering literature announced at the beginning of the semester | |
| Assessment type and form | | |
| Assessment grading | | |
| Comments | | |

Unit description of module 23: Advanced Testing Methods – Exercises

| | |
|---|---|
| Unit title | Advanced Testing Methods – Exercises |
| Code | |
| Module title | Advanced Testing Methods |
| Unit contents | <ul style="list-style-type: none"> • Lab exercises with software tools pertaining to the contents described in the unit Advanced Testing Methods – lectures • practical teamwork on real world problems • Lessons learned session after group work |
| Teaching methods | Teamwork in small software development groups |
| Semester periods (hours) per week | 2 |
| Workload (h) | 80h |
| Class hours | 30h |
| Total time of examination incl. preparation (h) | |
| Total time of individual study (h) | 50h |
| Total time of practical training (h) | n/a |
| Unit language | English |
| Lecturer | Dr. Torsten Schönfelder, Deutsche Lufthansa |
| Recommended reading | Current Software Engineering literature announced at the beginning of the semester |
| Assessment type and form | |
| Assessment grading | |
| Comments | |

Modul 24 Enterprise Architecture Engineering

| | | |
|---|---|----|
| Modultitel | Enterprise Architecture Engineering | PO |
| Modulnummer | 24 | PO |
| Modulcode | EAE | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | Wirtschaftsinformatik (M.Sc.) | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Wahlpflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 5 CP / 150 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Keine | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | <p>Portfolio mit folgenden Werkstücken:</p> <ul style="list-style-type: none"> • Projektarbeit (Bearbeitungszeit 8 Wochen) mit Präsentation (mindestens 20, maximal 30 Minuten). In diesem Werkstück sind maximal 50 Punkte erreichbar. • Schriftliches Testat (Prüfungsdauer 60 Minuten). In diesem Werkstück sind maximal 50 Punkte erreichbar. <p>Die Note ergibt sich aus der Summe der erreichten Punktzahlen. Zum Bestehen reichen 50% der erreichbaren Punkte aus.</p> | PO |
| Lernergebnisse und Kompetenzen | <p>Lernergebnisse</p> <p>Die Absolventinnen und Absolventen verfügen über ein vertieftes anwendungsbezogenes Wissen über Entwurfsprinzipien und Frameworks zur Entwicklung einer Unternehmensarchitektur, die ganzheitlich am Unternehmen und dessen Strategie ausgerichtet ist. Dazu kennen sie einschlägige Entwurfsprinzipien und Frameworks (z. B. OMG TOGAF).</p> <p>Fachkompetenzen</p> <p>Die Studierenden können als „Enterprise Architect“ Unternehmensarchitekturen ganzheitlich entwerfen und entwickeln, die an der Unternehmensstrategie ausgerichtet sind. In den Übungen präsentieren die Studierenden einzeln oder gemeinsam erarbeitete Entwürfe und Implementierungen, die sie gegenüber fachlicher Kritik vertreten.</p> | PO |

| | | |
|-------------------------------------|--|----|
| Inhalte des Moduls | Enterprise Architecture Engineering – Vorlesung, Enterprise Architecture Engineering – Übungen | PO |
| Lehrformen des Moduls | Vorlesung, Übung, selbstbestimmtes Lernen, Projektbearbeitung | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Sommersemester | PO |
| Modulkoordination | Prof. Dr. Rainer Buhr | |
| Hinweise | | |

Unitbeschreibung 24: Enterprise Architecture Engineering – Vorlesung

| | |
|--|--|
| Name der Unit | Enterprise Architecture Engineering – Vorlesung |
| Code | |
| Name des Moduls | Enterprise Architecture Engineering (24) |
| Inhalte der Unit | <p>Im Rahmen dieser Lehrveranstaltung werden grundlegende Inhalte und Methoden zu folgenden Themenkreisen behandelt:</p> <ul style="list-style-type: none"> • Unternehmensarchitektur und Unternehmensstrategie • Analyse von Geschäftsstrukturen und Prozessen • Strukturierung und Standardisierung von Prozessen • Enterprise IT-Design • Enterprise IT-Weiterentwicklung • Einsatz von Informationssystemen zur Informationsgewinnung und -analytik • Architekturprinzipien und Praktiken • Effektivität, Effizienz, Agilität und Nachhaltigkeit einer Unternehmensarchitektur |
| Lehrformen | Vorlesung |
| SWS der Unit | 2 h |
| Workload (h) | 75 h |
| Anteil der Präsenzzeit (h) | 30 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 45 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Ruth Schorr, Prof. Dr. Rainer Buhr |
| Basis-Literatur | Ahlemann, F. et al. (Ed.) „Strategic Enterprise Architecture Management“, Springer, Berlin, aktuelle Auflage |

| | |
|---|---|
| | <p>Hanschke, I. „Enterprise Architecture Management“, Hanser, Wiesbaden, aktuelle Auflage</p> <p>Hanschke, I. et al. „Strategisches Prozess Management“, Hanser, Wiesbaden, aktuelle Auflage</p> <p>Perroud, T. et al. „Enterprise Architecture Patterns“, Springer, Berlin, aktuelle Auflage</p> <p>The Open Group „TOGAF“, Van Haren Publishing, aktuelle Auflage</p> |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Unitbeschreibung 24: Enterprise Architecture Engineering – Übungen

| | |
|---|--|
| Name der Unit | Enterprise Architecture Engineering – Übungen |
| Code | |
| Name des Moduls | Enterprise Architecture Engineering (24) |
| Inhalte der Unit | Ziel der Übungen ist es, die Studierenden anhand kleinen, selbst auszuarbeitenden Szenarien für die behandelten Themenkreise zu sensibilisieren. |
| Lehrformen | Übung |
| SWS der Unit | 2 h |
| Workload (h) | 75 h |
| Anteil der Präsenzzeit (h) | 30 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 45 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Prof. Dr. Ruth Schorr, Prof. Dr. Rainer Buhr |
| Basis-Literatur | Siehe Beschreibung Unit Enterprise Architecture Engineering – Vorlesung |
| Art und Form des Leistungsnachweises der Unit | Keine |

| | |
|--|-------|
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 25 Current Topics in Computer Science

| | | |
|---------------------------------|--|----|
| Module title | Current Topics in Computer Science | PO |
| Module number | 25 | PO |
| Module code | | |
| Study program | Allgemeine Informatik (M.Sc.) | PO |
| Module usability | Applicable in other M.Sc. Programs in computer science | PO |
| Module duration | One semester | PO |
| Recommended semester | 3rd Semester | PO |
| Module type | Elective | PO |
| ECTS (cp) / Workload (h) | 5 CP / 150 h | PO |
| Recommended previous knowledge | None | PO |
| Module prerequisites | None | |
| Module examination requirements | None | PO |
| Module examination | Paper written according to international scientific journal standards (processing time 6 weeks) and oral presentation (30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation. | PO |
| Learning outcomes and skills | Upon completion of this course, the student is able to: <ul style="list-style-type: none"> • recognize important developments in the field of Computer Science, • incorporate new methods into the software and systems development process • criticize new technology with respect to their usability in critical systems development. Training for non-specialist competencies. Students learn: <ul style="list-style-type: none"> • to search for, read, summarize and cite scientific literature on a large scale; • to read and interpret national and international publications; • to write a report as a scientific paper; • to give a scientific talk. | PO |
| Module contents | Current Topics in Computer Science – Seminar | PO |
| Module teaching methods | Seminar | PO |
| Module language | English | PO |

| | | |
|---------------------|---------------------------|----|
| Module availability | Annual | PO |
| Module coordination | Prof. Dr. Eicke Godehardt | |
| Comments | | |

Unit description of module 25: Current Topics in Computer Science – Seminar

| | |
|--|--|
| Unit title | Current Topics in Computer Science – Seminar |
| Code | |
| Module title | Current Topics in Computer Science (25) |
| Unit contents | Current topics in Computer Science. |
| Unit teaching methods | Seminar |
| Semester periods per week (h) | 2 h |
| Workload (h) | 150 h |
| Attendance time (h) | 36 h |
| Examination time including preparation (h) | 0 h |
| Private study (h) | 114 h |
| Practical study (h) | 0 h |
| Unit language | English |
| Tutor(s) | All professors of the Master's programs Allgemeine Informatik (M.Sc.) and High Integrity Systems (M.Sc.) |
| Literature | Current research literature |
| Type and form of unit's assessment | None |
| Evaluation of unit's assessment | None |
| Notes | |

Modul 26 Projekt Intelligente Systeme

| | | |
|---|--|----|
| Modultitel | Projekt Intelligente Systeme | PO |
| Modulnummer | 26 | PO |
| Modulcode | | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 3. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 10 CP / 300 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Vorkenntnisse aus dem Bereich Intelligente Systeme empfohlen | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | PO |
| Lernergebnisse und Kompetenzen | <p>Im Studienfeld „Intelligente Systeme“ erwerben die Absolventinnen und Absolventen Kenntnisse, die sie befähigen aktuelle Schlüsseltechnologien für intelligente interagierende Systeme zu entwickeln, die in einer natürlichen Umwelt auf intuitive Weise mit ihren Nutzern kooperieren.</p> <p>Insbesondere können die Studierenden</p> <ul style="list-style-type: none"> • komplexere Themen und Aufgaben aus dem Studienfeld Intelligente Systeme, deren Lösung nicht durch schematische Anwendung vorformulierter Muster erfolgen kann, analysieren und unter Zuhilfenahme von selbst recherchierter Fachliteratur bearbeiten, • Konzeptions- und Modellierungsaufgaben unter Berücksichtigung wissenschaftlicher, technischer, ökonomischer und gesellschaftlicher Rahmenbedingungen bzw. Standards mit etablierten Methoden, Techniken und Werkzeugen durchführen und • das im Studium erworbene Wissen und die darauf aufbauenden Fähigkeiten und Kompetenzen anwenden und erweitern bzw. aktualisieren. <p>Im Bericht und Vortrag präsentieren die Studierenden einzeln oder gemeinsam erarbeitete Modelle und vertreten diese gegenüber fachlicher Kritik.</p> | PO |

| | | |
|-------------------------------------|--|----|
| Inhalte des Moduls | Projekt Intelligente Systeme | PO |
| Lehrformen des Moduls | Projekt | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Mindestens einmal jährlich; angebotene Projekte werden zu Beginn eines jeden Semesters ausgeschrieben; zugleich wird eine Vorschau auf die Angebote des jeweils folgenden Semesters bekannt gegeben. | PO |
| Modulkoordination | Prof. Dr. Eicke Godehardt | |
| Hinweise | | |

Unitbeschreibung 26: Projekt 1

| | |
|---|---|
| Name der Unit | Projekt Intelligente Systeme |
| Code | |
| Name des Moduls | Projekt Intelligente Systeme (26) |
| Inhalte der Unit | Themen aus dem Bereich Intelligente Systeme |
| Lehrformen | Projekt |
| SWS der Unit | 10 h |
| Workload (h) | 300 h |
| Anteil der Präsenzzeit (h) | 75 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 225 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Alle Lehrenden des Studiengangs |
| Basis-Literatur | Themenabhängig, aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben. |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 27 Projekt Digitalisierung

| | | |
|---|--|----|
| Modultitel | Projekt Digitalisierung | PO |
| Modulnummer | 27 | PO |
| Modulcode | | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 2. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 10 CP / 300 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Keine | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | PO |
| Lernergebnisse und Kompetenzen | <p>Das Studienfeld „Digitalisierung“ vermittelt die informationstheoretische Methodik und das Handwerkzeug zur Generierung (u. a. mithilfe von Sensoren) und technischen, sicheren Beherrschbarkeit der (großen) Datenmengen, die durch das Internet of Things (IoT) generiert werden. Es ermächtigt ergänzend daher dazu, die Industrie 4.0 mitzugestalten.</p> <p>Insbesondere können die Studierenden</p> <ul style="list-style-type: none"> • komplexere Themen und Aufgaben aus dem Studienfeld Digitalisierung, deren Lösung nicht durch schematische Anwendung vorformulierter Muster erfolgen kann, analysieren und unter Zuhilfenahme von selbst recherchierter Fachliteratur bearbeiten, • Konzeptions- und Modellierungsaufgaben unter Berücksichtigung wissenschaftlicher, technischer, ökonomischer und gesellschaftlicher Rahmenbedingungen bzw. Standards mit etablierten Methoden, Techniken und Werkzeugen durchführen und • das im Studium erworbene Wissen und die darauf aufbauenden Fähigkeiten und Kompetenzen anwenden und erweitern bzw. aktualisieren. <p>Im Bericht und Vortrag präsentieren die Studierenden einzeln oder gemeinsam erarbeitete Modelle und vertreten diese gegenüber fachlicher Kritik.</p> | PO |

| | | |
|-------------------------------------|--|----|
| Inhalte des Moduls | Projekt Digitalisierung | PO |
| Lehrformen des Moduls | Projekt | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Mindestens einmal jährlich; angebotene Projekte werden zu Beginn eines jeden Semesters ausgeschrieben; zugleich wird eine Vorschau auf die Angebote des jeweils folgenden Semesters bekannt gegeben. | PO |
| Modulkoordination | Prof. Dr. Eicke Godehardt | |
| Hinweise | | |

Unitbeschreibung 27: Projekt Digitalisierung

| | |
|---|--|
| Name der Unit | Projekt Digitalisierung |
| Code | |
| Name des Moduls | Projekt Digitalisierung (27) |
| Inhalte der Unit | Themen aus dem Bereich Digitalisierung |
| Lehrformen | Projekt |
| SWS der Unit | 10 h |
| Workload (h) | 300 h |
| Anteil der Präsenzzeit (h) | 75 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 225 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Alle Lehrenden des Studiengangs |
| Basis-Literatur | Themenabhängig, aktuelle Literaturhinweise werden zu Beginn des Semesters bekannt gegeben. |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 28 Projekt Softwaretechnik

| | | |
|---|--|----|
| Modultitel | Projekt Softwaretechnik | PO |
| Modulnummer | 28 | PO |
| Modulcode | | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 1. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 10 CP / 300 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Keine | PO |
| Voraussetzungen für die Teilnahme am Modul | Keine | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 30, maximal 60 Minuten) | PO |
| Lernergebnisse und Kompetenzen | <p>Die Absolventinnen und Absolventen werden durch das Studienfeld „Softwaretechnik“ in die Lage versetzt, die zeitgemäße Umsetzung von verteilten Geschäfts- und Entwicklungsprozess- und IT-Strategien eigenverantwortlich voranzutreiben, sei es durch Analyse und Konzeption in Forschungs- und Entwicklungsprojekten oder durch die ganzheitliche Herangehensweise im Projektmanagementumfeld. Dabei liegt ein besonderer Fokus auf den Herausforderungen durch das automatisierte Lernen aus (großen) Datenmengen unter besondere Berücksichtigung von Sicherheitsaspekten.</p> <p>Insbesondere können die Studierenden</p> <ul style="list-style-type: none"> • komplexere Themen und Aufgaben aus dem Studienfeld Softwaretechnik, deren Lösung nicht durch schematische Anwendung vorformulierter Muster erfolgen kann, analysieren und unter Zuhilfenahme von selbst recherchierter Fachliteratur bearbeiten, • Konzeptions- und Modellierungsaufgaben unter Berücksichtigung wissenschaftlicher, technischer, ökonomischer und gesellschaftlicher Rahmenbedingungen bzw. Standards mit etablierten Methoden, Techniken und Werkzeugen durchführen und | PO |

| | | |
|-------------------------------------|---|----|
| | <ul style="list-style-type: none"> das im Studium erworbene Wissen und die darauf aufbauenden Fähigkeiten und Kompetenzen anwenden und erweitern bzw. aktualisieren. <p>Im Bericht und Vortrag präsentieren die Studierenden einzeln oder gemeinsam erarbeitete Modelle und vertreten diese gegenüber fachlicher Kritik.</p> | |
| Inhalte des Moduls | Projekt Softwaretechnik | PO |
| Lehrformen des Moduls | Projekt | PO |
| Sprache | Deutsch | PO |
| Häufigkeit des Angebots von Modulen | Mindestens einmal jährlich; angebotene Projekte werden zu Beginn eines jeden Semesters ausgeschrieben; zugleich wird eine Vorschau auf die Angebote des jeweils folgenden Semesters bekannt gegeben. | PO |
| Modulkoordination | Prof. Dr. Eicke Godehardt | |
| Hinweise | | |

Unitbeschreibung 28: Projekt Softwaretechnik

| | |
|---|--|
| Name der Unit | Projekt Softwaretechnik |
| Code | |
| Name des Moduls | Projekt Softwaretechnik (28) |
| Inhalte der Unit | Themen aus dem Bereich Softwaretechnik |
| Lehrformen | Projekt |
| SWS der Unit | 10 h |
| Workload (h) | 300 h |
| Anteil der Präsenzzeit (h) | 75 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 225 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Alle Lehrenden des Studiengangs |
| Basis-Literatur | Themenabhängig, aktuelle Literaturhinweise werden zu Beginn des Semesters bekannt gegeben. |
| Art und Form des Leistungsnachweises der Unit | Keine |
| Bewertung des Leistungsnachweises der Unit | Keine |
| Hinweise | |

Modul 29 Master-Arbeit mit Kolloquium

| | | |
|---|--|----|
| Modultitel | Master-Arbeit mit Kolloquium | PO |
| Modulnummer | 29 | PO |
| Modulcode | | |
| Studiengang | Allgemeine Informatik (M.Sc.) | PO |
| Verwendbarkeit des Moduls | | PO |
| Dauer des Moduls | ein Semester | PO |
| Empfohlenes Semester im Studienverlauf | 4. Semester | PO |
| Art des Moduls | Pflichtmodul | PO |
| ECTS-Punkte (CP) / Workload (h) | 30 CP / 900 h | PO |
| Empfohlene inhaltliche Vorkenntnisse | Erfolgreicher Abschluss der Module entsprechend der PO im Umfang von 90 ECTS | PO |
| Voraussetzungen für die Teilnahme am Modul | Erfolgreicher Abschluss zweier der drei Projektmodule sowie für jedes Studienfeld zweier Pflicht- und zweier Wahlpflichtmodule. Die Zulassung zum dritten Projektmodul muss ausgesprochen sein | |
| Voraussetzungen für die Teilnahme an der Modulprüfung | Keine | PO |
| Modulprüfung | Master-Arbeit (Bearbeitungszeit 22 Wochen) und Kolloquium (mindestens 30, maximal 60 Minuten) | PO |
| Lernergebnisse und Kompetenzen | <p>Die Absolventinnen und Absolventen sind qualifiziert, um kompetent, eigenverantwortlich und selbständig anspruchsvolle und innovative Funktionen in Projektierung, Projektdurchführung, Entwicklung, Beratung, Vertrieb in Unternehmen der Wirtschaft, Industrie sowie der öffentlichen Hand auszuüben bzw. sich in der Forschung weiter zu qualifizieren.</p> <p>Die Absolventinnen und Absolventen werden befähigt, komplexere Probleme und Aufgaben in der unternehmerischen Praxis (z. B. „Industrie 4.0“, „Internet der Dinge“, R&D Projekte) im Team erfolgreich zu bearbeiten. Diese Probleme und Aufgaben erfordern einen ganzheitlichen und grundlagenbasierten Analyse- und Konzeptionsansatz, für den oft noch keine standardisierten Vorgehensmodelle und/oder widerstreitende Lösungsansätze existieren. Die Absolventinnen und Absolventen sind weiter befähigt, Implikationen ihres Handelns in Form zukünftiger Probleme, Technologien und Entwicklungen zu antizipieren.</p> | PO |
| Inhalte des Moduls | Master-Arbeit mit Kolloquium | PO |
| Lehrformen des Moduls | Projekt | PO |
| Sprache | Deutsch | PO |

| | | |
|-------------------------------------|---------------------------|----|
| Häufigkeit des Angebots von Modulen | jedes Semester | PO |
| Modulkoordination | Prof. Dr. Eicke Godehardt | |
| Hinweise | | |

Unitbeschreibung 29: Masterarbeit mit Kolloquium

| | |
|---|--|
| Name der Unit | Masterarbeit mit Kolloquium |
| Code | |
| Name des Moduls | Masterarbeit mit Kolloquium (29) |
| Inhalte der Unit | Masterarbeit mit Kolloquium |
| Lehrformen | Projekt |
| SWS der Unit | 0 h |
| Workload (h) | 900 h |
| Anteil der Präsenzzeit (h) | 15 h |
| Anteil Prüfungszeit inkl. Vorbereitung (h) | 0 h |
| Anteil Selbststudium (h) | 885 h |
| Anteil Praxiszeit (h) | 0 h |
| Sprache der Unit | Deutsch |
| Lehrende/-r | Alle Lehrenden des Studiengangs |
| Basis-Literatur | Themenabhängig, aktuelle Literaturhinweise werden zu Beginn der Masterarbeit bekanntgegeben. |
| Art und Form des Leistungsnachweises der Unit | Master-Arbeit (Bearbeitungszeit 22 Wochen, Gewicht für Modulnote 70%) und Kolloquium (mindestens 30, maximal 60 Minuten, Gewicht für Modulnote 30%) |
| Bewertung des Leistungsnachweises der Unit | Differenziert (Note 1–4; gemäß der Allg. Bestimmungen für Prüfungsordnungen mit den Abschlüssen Bachelor und Master an der Frankfurt University of Applied Sciences) |
| Hinweise | |