

Modulhandbuch

Masterstudiengang

»Informatik«

SPO 2017



**Hochschule
Augsburg** University of
Applied Sciences

Fakultät für
Informatik

Stand: 27. April 2022

Die Modulbeschreibungen dienen der inhaltlichen Orientierung in Ihrem Studium.

Rechtlich verbindlich ist nur die jeweils geltende Studien- und Prüfungsordnung.

Contents

1	Modul Wissenschaftliche Grundlagen	3
1.1	Angewandte Mathematik	3
1.2	Theoretische Informatik	5
2	Modul Systemarchitekturen	8
2.1	Softwaresysteme	8
2.2	Hardwaresysteme	10
3	Modul Vertiefung	12
3.1	Fachbezogene Wahlpflichtfächer	12
4	Modul Wissenschaftliches Arbeiten	13
4.1	Projektarbeit	13
4.2	Workshop	15
4.3	Master's Seminar	17
5	Modul Masterarbeit	18
5.1	Master Thesis	18

1 Modul Wissenschaftliche Grundlagen

1.1 Angewandte Mathematik

Modulbezeichnung	Angewandte Mathematik
Titel in Englisch	Applied Mathematics
Prüfungsnummer	8900010
Modulkürzel	AMATH
Modulbereich	Modul A: Wissenschaftliche Grundlagen
Modulverantwortlicher	Prof. Dr. Stefan Glasauer
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik Fakultät für angewandte Geistes- und Naturwissenschaften
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Wintersemester angeboten.
Lehrveranstaltungen in den Modulen	Angewandte Mathematik (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Inhalte des Moduls	<ul style="list-style-type: none"> • Diskrete Fourier-Transformation <ul style="list-style-type: none"> – Fourier-Reihen – Diskrete Fourier-Transformation (DFT) – Schnelle Fourier-Transformation (FFT) – Anwendung: Digitale Filter • Algebraische Strukturen <ul style="list-style-type: none"> – Gruppentheorie – Ringe und Körper – Zahlentheorie – Polynomringe – Endliche Körper – Anwendung: Public Key-Kryptographie – Anwendung: Fehlerkorrigierende Codes
Qualifikationsziele des Moduls Lernziel/Kompetenzen	Die Studierenden erwerben Kenntnisse über algebraische Konzepte und Methoden, die als zeitloses methodisches Rüstzeug für die Beschreibung und Lösung informations- und kommunikationstechnischer Probleme dienen. Die Anwendung dieser Konzepte wird jeweils exemplarisch illustriert in den Bereichen Kryptographie, Codierungstheorie und digitale Filter.
Lehr- und Lernmethoden des Moduls	Seminaristischer Unterricht, Seminar, Übungen, Teilnehmeraktive Lehrveranstaltung
Voraussetzungen für die Teilnahme	Kenntnisse über komplexe Zahlen und lineare Algebra
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	Klausur, 60 Minuten, Hilfsmittel: Taschenrechner, 2 DIN-A4-Seiten handgeschriebene Formelsammlung

Literaturliste	<p>Burger, Burge: Digitale Bildverarbeitung, Springer 2015</p> <p>Manz: Fehlerkorrigierende Codes, Springer 2017</p> <p>Neubauer: Diskrete Fouriertransformation, Vieweg + Teubner 2012</p> <p>Socher: Algebra für Informatiker, Hanser 2012</p> <p>Wätjen: Kryptographie, Springer 2018</p> <p>Witt: Algebraische und zahlentheoretische Grundlagen der Informatik, Springer 2014</p>
----------------	--

1.2 Theoretische Informatik

Modulbezeichnung	Theoretische Informatik
Titel in Englisch	Theoretical Computer Science
Prüfungsnummer	8900020
Modulkürzel	THEM
Modulbereich	Modul A: Wissenschaftliche Grundlagen
Modulverantwortlicher	Prof. Dr. Thomas Rist
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Sommersemester angeboten.
Lehrveranstaltungen in den Modulen	Theoretische Informatik (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.

Inhalte des Moduls	<p>Theoretische Informatik befasst sich mit den grundsätzlichen Fragestellungen, die bei der Konstruktion und Programmierung von Rechenmaschinen auftreten. Welche Problemstellungen lassen sich überhaupt vernünftig mit einer Rechenmaschine behandeln, welche prinzipiell nicht? Welche Problemstellungen sind inhärent schwierig und bringen - trotz prinzipieller Lösbarkeit - reale Rechenmaschinen schnell an ihre Grenzen? Welche Aussagen über Programme lassen sich automatisch ableiten, welche nicht? Welchen Unterschied macht es, wie solche Maschinen konzipiert sind und welche Ausdrucksmächtigkeit die darauf ausführbaren Programme haben?</p> <p>Aufbauend auf den im Bachelorstudiengang erworbenen Grundlagenwissen führt der Kurs - den klassischen Weg folgend - kompakt über die Themen „Formale Sprachen“, Grammatiken und Automaten hin zur Turing Maschine und ihren Varianten. Mit der vertieften Diskussion wichtiger Konzepte wie Äquivalenzen zwischen formalen Sprachen und Berechnungsmechanismen, Akzeptanz- und Entscheidbarkeit werden die Teilnehmer*innen an Denkweisen und formale Beweistechniken der Theoretischen Informatik herangeführt und lernen repräsentative Instanzen un- / semi- / entscheidbarer Probleme kennen.</p> <p>Über eine Auffrischung der aus dem Bachelorstudium als bekannt vorausgesetzten O-Notation zur Einordnung des Ressourcenverbrauchs von Berechnungen, wird der Bogen zur Komplexitätstheorie gespannt. Mit der Einführung wichtiger Komplexitätsklassen und deren Beziehungen zueinander werden die Begriffe Vollständigkeit und Reduktion geklärt. In die Betrachtungen miteinbezogen werden auch neuere Konzeptionen von Rechenmaschinen und Algorithmen (u.a. Parallelrechner, probabilistische Algorithmen, Quantum-Computer).</p> <p>Anhand formaler Logiksprachen werden die Themen Expressivität, Modell und Kalkül eingeführt und mit Blick auf Berechenbarkeit, Komplexität und deren Bedeutung für andere Teilgebiete der Informatik (u.a. Programmverifikation und symbolische KI / Wissensverarbeitung) diskutiert.</p> <p>Die im Modul behandelten Themen umfassen:</p> <ul style="list-style-type: none"> • Beweistechniken für induktiv definierte Mengen • Eigenschaften von Sprachklassen der Chomsky-Hierarchie • Nachweis der Äquivalenzen verschiedener Verarbeitungsmechanismen • Berechenbarkeit, Konzepte und grundlegende Theoreme (u.a. Halteproblem, PKP, Satz von Church, Satz von Rice) • Komplexitätstheorie, O-Notation, Komplexitätsklassen, Reduktion • Komplexität und weitere Komplexitätsklassen im Kontext von Parallelrechner, probabilistische Algorithmen und Quanten-Computer • Formale Logiksprachen, Definition, Expressivität, Kalküle, Berechenbarkeit, Komplexität und Anwendungsfelder, u.a. Programmverifikation und symbolische Künstliche Intelligenz und Wissensverarbeitung.
--------------------	--

Qualifikationsziele des Moduls Lernziel/Kompetenzen	<p>Kenntnisse: Die Studierenden:</p> <ul style="list-style-type: none"> • verschaffen sich einen Überblick zu den zentralen Fragestellungen, Herangehensweisen und Erkenntnissen der Theoretischen Informatik sowie deren grundlegende Bedeutung für anwendungsorientierte Informatik, • kennen für die Gebiete Berechenbarkeit und Komplexität repräsentative Problemstellungen. <p>Fertigkeiten: Die Studierenden sind in der Lage:</p> <ul style="list-style-type: none"> • grundlegende Argumentationen / Beweisansätze der Theoretischen nachzuvollziehen und an einfachen Problemstellungen selbst durchzuführen, • Algorithmen und Softwaresysteme hinsichtlich ihrer Komplexität zu analysieren und in passende Komplexitätsklassen einordnen, • Sachverhalte in einer Logiksprache zu formalisieren <p>Kompetenzen: Die Studierenden können:</p> <ul style="list-style-type: none"> • erworbenes Wissen und Fähigkeiten auch auf neue Problemstellungen übertragen und anwenden, • Argumentationen zu Eigenschaften von Algorithmen und Programmen formal untermauern.
Lehr- und Lernmethoden des Moduls	Seminaristischer Unterricht mit verzahnten Übungseinheiten und aktiver Beteiligung der Teilnehmer*innen.
Voraussetzungen für die Teilnahme	Grundkenntnisse zu den Themen Formalen Sprachen, endlichen Automaten, Algorithmen und Datenstrukturen, wie sie typischerweise im Grundlagenbereich eines Informatik-Bachelorstudiums vermittelt werden.
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	Klausur, 60 Minuten, keine Hilfsmittel
Literaturliste	<p>D.W. Hoffmann: Theoretische Informatik. 4. Auflage, Hanser (2018)</p> <p>R. Socher: Theoretische Grundlagen der Informatik. 2.Auflage, Fachbuchverlag Leipzig (2005)</p> <p>M. Kreuzer & S. Kühling: Logik für Informatiker. Pearson Studium (2006)</p> <p>Aktuelle Literaturempfehlungen und Online-Quellen zu behandelten Themen werden im Kurs bekannt gegeben.</p>

2 Modul Systemarchitekturen

2.1 Softwaresysteme

Modulbezeichnung	Softwaresysteme
Titel in Englisch	Software Systems
Prüfungsnummer	8900030
Modulkürzel	SWS
Modulbereich	Modul B: Systemarchitekturen
Modulverantwortlicher	Prof. Dr. Philip Heidegger Prof. Dr.-Ing. Dominik Merli
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Sommersemester angeboten.
Lehrveranstaltungen in den Modulen	Softwaresysteme (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Inhalte des Moduls	<p>Im Rahmen dieser Veranstaltung präsentieren zwei Dozenten Softwaresysteme aus ihren Arbeitsgebieten und diskutieren zentrale Aspekte mit den Studierenden.</p> <p>1. Systemprogrammierung mit Fokus auf Sicherheit (Prof. Dr.-Ing. Dominik Merli) Systemprogrammiersprachen werden verwendet, um Betriebssysteme, Firmware oder performanceorientierte Applikationen wie Web-Browser zu entwickeln. Die Paradebeispiele C und C++ weisen jedoch seit Jahrzehnten Mängel bzgl. Memory Safety auf, die zu unzähligen Schwachstellen geführt haben. In diesem Kurs wird die Sprache Rust motiviert und eingeführt. Neben der Programmierung in Rust werden auch die Vorteile des umfangreichen Ökosystems diskutiert. Studierende wenden die neu erworbenen Kenntnisse in einem anwendungsnahen Projekt an, um kreative Ideen technisch sauber und sicher umsetzen zu können.</p> <p>2. Webanwendungen (Prof. Dr. Phillip Heidegger) Viele Anwendungen sollen heutzutage auf einer Vielzahl von Geräten, u.a. auf Handys, PCs, Tablets verwendet werden. Um nicht für jede Plattform eine eigene native Anwendung entwickeln zu müssen, wird häufig auf die Entwicklung von Webanwendungen zurückgegriffen. Da Webanwendungen in Browsern ablaufen, und sich deren Ökosystem stark von dem der traditionellen Betriebssysteme unterscheidet, sind Webanwendungen häufig andersartig strukturiert und Ihre Architektur unterscheidet sich von denen normaler nativer Anwendungen. Nach einer kurzen Einführung in die Architektur traditioneller GUI basierter Anwendungen werden die Unterschiede und die damit verbundenen Herausforderungen von Webanwendungen systematisch analysiert und mögliche Lösungen evaluiert. Wichtige Technologien, die vorgestellt werden, sind TypeScript, JQuery und Angular.</p>

Qualifikationsziele des Moduls Lernziel/Kompetenzen	<p>Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> • Besonderheiten der für die vorgestellten Softwaresysteme verwendeten Technologien zu verstehen. • Einschlägige Programmiertechniken für die Erstellung von Softwaresystemen anzuwenden. • Anforderungen und Herausforderungen an Softwaresysteme systematisch zu analysieren. • Vorgegebene Beispiele systematisch bezüglich Effektivität und Effektivität zu beurteilen. • Softwaresysteme aus existierenden Komponenten zu konstruieren.
Lehr- und Lernmethoden des Moduls	Seminaristischer Unterricht, Seminar, Laborpraktikum, Studienarbeit, Übung, Teilnehmeraktive Lehrveranstaltung
Voraussetzungen für die Teilnahme	Fortgeschrittene Kenntnisse aus den Bereichen Software-Entwicklung und Software-Engineering.
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	<p>Portfolioprüfung:</p> <ul style="list-style-type: none"> • Studienarbeit A, 5-10 Seiten, 50% • Studienarbeit B, 5-10 Seiten, 50%
Literaturliste	<p>W. Burger, M. J. Burge: Digitale Bildverarbeitung, Springer (2015)</p> <p>Pacheco, P.: Introduction to Parallel Programming, Morgan Kaufman, 2011</p> <p>I. Ristic: Bulletproof SSL and TLS, Feisty Duck (2015)</p> <p>Insight Segmentation and Registration Toolkit (ITK), www.itk.org</p> <p>https://angular.io</p> <p>https://www.typescriptlang.org</p>

2.2 Hardwaresysteme

Modulbezeichnung	Hardwaresysteme
Titel in Englisch	Hardware Systems
Prüfungsnummer	8900040
Modulkürzel	HWS
Modulbereich	Modul B: Systemarchitekturen
Modulverantwortlicher	Prof. Dr. Gundolf Kiefer
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Sommersemester angeboten.
Lehrveranstaltungen in den Modulen	Hardwaresysteme (2 SWS) Praktikum Hardwaresysteme (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Inhalte des Moduls	Hintergrundwissen: <ul style="list-style-type: none"> • Technologische Grundlagen & Trends • Modellierungssprachen (VHDL, SystemC) • Aufbau & Funktionsweise aktueller System-on-Chip-Plattformen • Systementwicklung mit FPGA-Technologie • Konfigurierbare Prozessoren • Hardware-orientierte Software-Entwicklung
Qualifikationsziele des Moduls Lernziel/Kompetenzen	Die Studierenden sind in der Lage, <ul style="list-style-type: none"> • digitale Systeme mit FPGAs in VHDL oder SystemC zu implementieren. • die Einsatzmöglichkeiten von Standard-Prozessoren, Spezialprozessoren sowie FPGA-Technologie zu charakterisieren. • die Effizienz alternativer Hardware-Architekturen zu beurteilen. • einfache Hardwaresysteme selber zu entwerfen.
Lehr- und Lernmethoden des Moduls	Seminaristischer Unterricht, Laborpraktikum und Studienarbeit zur Anwendung und Vertiefung der erworbenen Kenntnisse, Teilnehmeraktive Lehrveranstaltung
Voraussetzungen für die Teilnahme	Keine
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

<p>Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten</p>	<p>Portfolioprüfung:</p> <ul style="list-style-type: none"> • Projektarbeit, 5-15 Seiten, 80% • Präsentation, 5-25 Minuten, 20%
<p>Literaturliste</p>	<p>J. L. Hennessy, D. A. Patterson: "Computer Architecture – A Quantitative Approach", 6th Edition, Morgan Kaufmann Publishers, 2017</p> <p>F. Kesel, R. Bartholomä: "Entwurf von digitalen Schaltungen und Systemen mit HDLs und FPGAs", 3. Auflage, De Gruyter Oldenburg, 2013</p> <p>J. Reichardt, B. Schwarz: "VHDL-Synthese: Entwurf digitaler Schaltungen und Systeme", 7. Auflage, De Gruyter Oldenburg, 2015</p> <p>J. Bhasker: "A SystemC Primer", 2nd Edition, Star Galaxy Publishing, Allentown, 2004</p>

3 Modul Vertiefung

3.1 Fachbezogene Wahlpflichtfächer

Modulbezeichnung	Fachbezogene Wahlpflichtfächer
Titel in Englisch	Optional required Subjects
Prüfungsnummer	
Modulkürzel	
Modulbereich	Modul C: Vertiefung
Modulverantwortlicher	Professorinnen und Professoren der Fakultät für Informatik
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Wahlpflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird sowohl im Sommersemester als auch im Wintersemester angeboten.
Lehrveranstaltungen in den Modulen	Die FWP-Fächer können aus dem Angebot der Fakultät für Informatik ausgewählt werden.
Lehrsprache	Siehe Angaben des jeweiligen FWP-Fachs
Inhalte des Moduls	Spezifische Fachkompetenz in den einzelnen Fächern.
Qualifikationsziele des Moduls Lernziel/Kompetenzen	Spezifische Fachkompetenz in den jeweiligen Fächern.
Lehr- und Lernmethoden des Moduls	Directed Reading, Seminaristischer Unterricht, Seminar, Übungen, Praktikum, Teilnehmeraktive Lehrveranstaltung
Voraussetzungen für die Teilnahme	Informationen zur Teilnahme an Wahlpflichtfächern können der Studien- und Prüfungsordnung entnommen werden.
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 10 - 20 (*), ECTS: 25 Präsenzzeit: 112,5 h - 225 h Selbststudium: 637,5 h - 525 h Gesamtaufwand: 750 h (*) Die Anzahl der SWS des Vertiefungsmoduls kann je nach Art der gewählten Lehrveranstaltungen zwischen 10 und 20 SWS betragen.
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	Weitere Informationen zu den fachbezogenen Wahlpflichtfächern finden Sie auf der Webseite des Studiengangs unter Studienrelevante Downloads.
Gewichtung der Einzelleistung in der Gesamtnote	Weitere Informationen zu den fachbezogenen Wahlpflichtfächern finden Sie auf der Webseite des Studiengangs unter Studienrelevante Downloads.
Literaturliste	Literatur wird im Seminar bekannt gegeben.

4 Modul Wissenschaftliches Arbeiten

4.1 Projektarbeit

Modulbezeichnung	Projektarbeit
Titel in Englisch	Project Work
Prüfungsnummer	8900090
Modulkürzel	PA
Modulbereich	Modul D: Wissenschaftliches Arbeiten
Modulverantwortlicher	Studiengangsleiter
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Das Modul wird regelmäßig im Sommersemester angeboten und hat eine Laufzeit von zwei Semestern.
Lehrveranstaltungen in den Modulen	Projektarbeit (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Inhalte des Moduls	Die Studierenden führen in kleinen Teams IT-Projekte durch. Zu den Aufgaben der Studierenden zählen Projektmanagement, sowie je nach Projekt die klassische oder agile Softwareentwicklung, Dokumentation und Präsentation der Ergebnisse.
Qualifikationsziele des Moduls Lernziel/Kompetenzen	Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage: <ul style="list-style-type: none">• Teamprozesse zu verstehen und typische Teamkonflikte zu lösen.• Softwareprojekte im Team hinsichtlich Zeit, Aufwände und Ressourcen eigenverantwortlich zu planen und durchzuführen.• Agile oder klassische Softwareentwicklungsmethoden praktisch anzuwenden.• Selbstständiges Erlernen von Methoden und Verfahren.• Analysieren und Bewerten von Methoden im Hinblick auf die Projektziele.• Projektergebnisse verständlich zu dokumentieren und ansprechend zu präsentieren.
Lehr- und Lernmethoden des Moduls	Studierende erarbeiten in Kleingruppen weitestgehend eigenständig Lösungen zu einem praxisorientierten Thema aus dem IT-Bereich. Ziel ist es einen Projektablauf möglichst realitätsnah mit allen Facetten abzubilden. Die Projektthemen werden von Prüfungsberechtigten der Fakultät für Informatik vergeben und umfassen einen praktischen Teil (Software/Hardware), eine Dokumentation (Studienarbeit) und eine Präsentation. Die Präsentation findet in der Regel im Rahmen eines Projekttagess statt. Die Abstimmung mit dem Projektsteller erfolgt in regelmäßigen persönlichen Treffen und über elektronische Kanäle. Die Bearbeitung ist nicht notwendigerweise an die Vorlesungszeit gebunden.

Voraussetzungen für die Teilnahme	Solide Kenntnisse aus den wichtigsten Themenbereichen der Informatik, wie z.B. Algorithmen und Datenstrukturen, Programmieren, Datenbanken, Datenkommunikation, Rechnerstrukturen und Betriebssysteme.
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 4, CPs: 10, Präsenzzeit: 60 h, Selbststudium: 240 h, Gesamtaufwand: 300 h
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	Portfolioprüfung: <ul style="list-style-type: none"> • Projektarbeit, 20-40 Seiten, 80% • Präsentation, 20-40 Minuten, 20%
Literaturliste	Projektspezifische Literatur wird vom Betreuer vor Beginn des Projektes bekanntgegeben.

4.2 Workshop

Modulbezeichnung	Workshop
Titel in Englisch	Workshop
Prüfungsnummer	8900115
Modulkürzel	WSH
Modulbereich	Modul D: Wissenschaftliches Arbeiten
Modulverantwortlicher	Prof. Dr.-Ing. Dominik Merli Dipl. Ing. Oliver Pilscheur
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Sommersemester angeboten.
Lehrveranstaltungen in den Modulen	Workshop (2 SWS))
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Inhalte des Moduls	<ul style="list-style-type: none"> • Methoden und Tools zur Gestaltung von Workshops • Digitale Medien zur Nutzung in Workshops • Erstellung von Workshop-Konzepten • Durchführung von Workshops
Qualifikationsziele des Moduls Lernziel/Kompetenzen	<p>Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> • verschiedene Arten von Workshops zu erklären. • Methoden zu beschreiben, die eingesetzt werden können. • einen Workshop zu konzipieren. • Probleme zu benennen, die bei der Durchführung eines Workshops auftreten können, und mögliche Lösungen vorzuschlagen
Lehr- und Lernmethoden des Moduls	Seminaristischer Unterricht und begleitende Übungen und Präsentationen zur Anwendung und Vertiefung der erworbenen Kenntnisse.
Voraussetzungen für die Teilnahme	Keine
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	SWS: 2, CPs: 5, Präsenzzeit: 30 h, Selbststudium: 120 h, Gesamtaufwand: 150 h
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	<p>Portfolioprüfung:</p> <ul style="list-style-type: none"> • Studienarbeit, 10 Seiten • Präsentation, 30 Minuten
Gewichtung der Einzelleistung in der Gesamtnote	mit Erfolg abgelegt / ohne Erfolg abgelegt

Literaturliste	<p>U. Lipp, H. Will: "Das große Workshop-Buch: Konzeption, Inszenierung und Moderation von Klausuren, Besprechungen und Seminaren", Beltz, 2008</p> <p>A. Funcke, E. Havenith: "Moderations-Tools: Anschauliche, aktivierende und klärende Methoden für die Moderations-Praxis", managerSeminare Verlags GmbH, 2017</p> <p>M. Haussmann: "UZMO - Denken mit dem Stift: Visuell präsentieren, dokumentieren und erkunden", Redline Verlag, 2014</p>
----------------	---

4.3 Master's Seminar

Title	Masterseminar
Title in English	Master's Seminar
Examination number	8900110
Module code	MASEM
Modul area	Academic work
Module coordinator	Prof. Dr. Alessandra Zarcone
Name of university lecturer	You can find the name of the lecturer for the current semester in the timetable.
Faculty	Faculty of Computer Science
Module type	Compulsory module
Duration of module / frequency of module offer	1 semester, winter semester, summer semester
Courses that make up the module	Master's seminar (2 credit hours)
Teaching language	English
Content of the module	The aim of the master's seminar is to promote the autonomous development of in-depth knowledge of the field of specialisation at hand. The students summarise the results in an essay and then make a presentation on the results. As all of the content is thematically related, synergistic effects are produced over the course of the seminar. The prevailing issues surrounding the chosen field of specialisation within computer science are addressed.
Qualification aims for the module learning objectives/skills	The students acquire detailed knowledge of the chosen specialist-field, develop their presentation and elocution expertise and improve their English language skills.
Teaching and learning methods of the module	The topics are announced in good time so that any students who are interested in the module can sign up online and select topics. Students are expected to make a presentation (in English) lasting approx. one hour which is followed by a discussion (content of presentation, presentation style). The topic must be set out in writing 1-2 weeks before the presentation date (covering approx. 10-15 pages of text, i.e. without images) and submitted to your supervisor.
Prerequisites for participation	Basic knowledge of the chosen specialist field.
Possibility to use module within student's own study programme or other programmes	Master's degree Computer Science
Total workload and its constituent parts	Credit hours: 2, ECTS credits: 5, Contact hours: 30h, Independent study: 120h, Total workload: 150h
Type of examination / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Presentation, 20-30 minutes, 40% • Written assignment, 10-15 pages, 60%
Reading list	

5 Modul Masterarbeit

5.1 Master Thesis

Modulbezeichnung	Master Thesis
Titel in Englisch	Master Thesis
Prüfungsnummer	9050
Modulkürzel	MA
Modulbereich	Modul E: Masterarbeit
Modulverantwortlicher	Professorinnen und Professoren der Fakultät für Informatik
Name des Hochschullehrers / der Hochschullehrerin	Den Namen des Lehrenden können Sie semesteraktuell dem Stundenplan entnehmen.
Fakultät	Fakultät für Informatik
Modulart	Pflichtmodul
Dauer des Moduls / Häufigkeit des Angebots des Moduls	Das Modul wird sowohl im Sommersemester als auch im Wintersemester angeboten.
Lehrveranstaltungen in den Modulen	Masterarbeit
Lehrsprache	Nach Absprache mit dem Betreuer
Inhalte des Moduls	<p>Das Studium beinhaltet eine Abschlussarbeit (Masterarbeit). Die Masterarbeit soll zeigen, dass die Studierenden in der Lage sind ein Problem aus dem Bereich der angewandten Wirtschaftsinformatik selbständig auf wissenschaftlicher Grundlage zu bearbeiten.</p> <p>Das Thema der Abschlussarbeit wird vom Studierenden eigenständig gesucht. Im Interesse einer raschen Praxiseingliederung der Studierenden soll die Masterarbeit vorwiegend im Rahmen eines Projekts mit einem Partner aus Industrie, Wirtschaft oder Verwaltung angefertigt werden. Natürlich können auch hochschulinterne Themen in Absprache mit dem Betreuenden ausgewählt werden.</p> <p>Das Thema der Masterarbeit soll so beschaffen sein, dass die Masterarbeit bei zusammenhängender ausschließlicher Bearbeitung in 4 Monaten abgeschlossen werden kann. Die Frist von der Ausgabe bis zur Abgabe der Arbeit kann höchstens 6 Monate betragen mit der Möglichkeit der einmaligen Verlängerung bis zu vier Monaten bei Glaubhaftmachung nicht selbst zu vertretender Verzögerungsgründe. Die Anforderungen an die Darstellung des Standes der wissenschaftlichen Erkenntnisse und an die Originalität/Neuartigkeit der Problemlösung liegen deutlich über denen an eine Bachelorarbeit (was sich auch im zeitlichen Aufwand ausdrückt).</p> <p>Durch die Masterarbeit sollen Kompetenzen auf den folgenden Gebieten nachgewiesen werden:</p> <ul style="list-style-type: none"> • Themenspezifisches Fachwissen aus dem Umfeld der Wirtschaftsinformatik, • Techniken, Methoden und Vorgehensweisen des wissenschaftlichen Arbeitens, • Projektmanagement bzgl. Zeit und Scope

Qualifikationsziele des Moduls Lernziel/Kompetenzen	<p>Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:</p> <ul style="list-style-type: none"> • ein Problem selbständig unter Verwendung wissenschaftlicher Techniken, Methoden und Vorgehensweisen eigenständig zu lösen • unterschiedliche Lösungsansätze zu identifizieren, zu bewerten, anzuwenden und in nachvollziehbarer und überzeugender Weise darzustellen • umfangreiche Aufgaben- und Problemstellungen eigenständig zu strukturieren, zu planen und in begrenzter Zeit und mit begrenztem Budget abzuschließen.
Lehr- und Lernmethoden des Moduls	Wissenschaftliches Arbeiten
Voraussetzungen für die Teilnahme	Informationen zur Masterarbeit können unter § 9 der Studien- und Prüfungsordnung entnommen werden.
Verwendbarkeit des Moduls innerhalb des eigenen, sowie für andere Studiengänge	Masterstudiengang Informatik
Gesamtarbeitsaufwand und seine Zusammensetzung	ECTS: 25, Arbeitsstunden: 750
Art der Prüfung / Voraussetzung für die Vergabe von Leistungspunkten	<p>Portfolioprüfung:</p> <ul style="list-style-type: none"> • Studienarbeit, 30-100 Seiten, 80% • Präsentation, 20-30 Minuten, 20%
Literaturliste	<p>Theisen, R. T.: Wissenschaftliches Arbeiten. Erfolgreich bei Bachelor- und Masterarbeit, Vahlen 2017</p> <p>Lück, W., Henke, M.: Technik des wissenschaftlichen Arbeitens. Seminararbeit, Diplomarbeit, Dissertation, Oldenbourg 2009</p> <p>Rössl, Dietmar: Die Diplomarbeit in der Betriebswirtschaftslehre, facultas.wuv 2008</p> <p>Eid, M., Gollwitzer, M., Schmitt, M.: Statistik und Forschungsmethoden, Beltz 2015</p>

Index

Angewandte Mathematik , 3

Fachbezogene Wahlpflichtfächer , 12

Hardwaresysteme , 10

Master Thesis , 18

Master's Seminar , 17

Projektarbeit , 13

Softwaresysteme , 8

Theoretische Informatik , 5

Workshop , 15