

Modulhandbuch

des Bachelor-Studiengangs

Informatik – Mobile Anwendungen

Bachelor of Science (B.Sc.)
Fb2: Informatik und Ingenieurwissenschaften
– Computer Science and Engineering

Inhalt

1. Qualifikationsprofil	S. 3
2. Empfohlener Studienverlauf (Modulübersicht)	S. 4
3. ECTS-/Workload-Übersicht	S. 5
4. Modulbeschreibungen	S. 6
Modul 1: „Algebra“	S. 7
Modul 2: „Analysis“	S. 11
Modul 3: „Einführung in Ubiquitous Computing“	S. 14
Modul 4: „Einführung in die Programmierung“	S. 18
Modul 5: „Betriebswirtschaftslehre“	S. 24
Modul 6: „Englisch“	S. 26
Modul 7: „Diskrete Mathematik“	S. 28
Modul 8: „Mikrocomputer Technik“	S. 33
Modul 9: „Algorithmen und Datenstrukturen“	S. 37
Modul 10: „Recht und Datenschutz“	S. 42
Modul 11: „Software Engineering - Analysis“	S. 45
Modul 12: „Embedded Systems“	S. 49
Modul 13: „Object-oriented Programming“	S. 53
Modul 14: „Databases“	S. 57
Modul 15: „Computer Networks“	S. 61
Modul 16: „Operating Systems“	S. 65
Modul 17: „Software Engineering“	S. 68
Modul 18: „Mobile Devices“	S. 71
Modul 19: „IT-Security“	S. 74
Modul 20: „Distributed Systems“	S. 77
Modul 21: „Mobile Application Exercises“	S. 81
Modul 22: „Serviceorientierte Architekturen“	S. 83
Modul 23: „Human Machine Interfaces“	S. 86
Modul 24: „Software-Projekt Mobile und Verteilte Anwendungen“	S. 90
Modul 25: „Wahlpflicht“	S. 93
Modul 26: „Interdisziplinäre Studium Generale“	S. 104
Modul 27: „Praxisphase“	S. 105
Modul 28: „Bachelor-Arbeit mit Kolloquium“	S. 109

1. Allgemeines Qualifikationsprofil des Studiengangs

Qualifikationsziel

Mit absolvieren des Bachelor-Studiengangs Informatik – Mobile Anwendungen erwerben die Studierenden einen ersten berufsqualifizierenden Abschluss, der sie befähigt, wissenschaftliche Theorien, Methoden und Techniken der Informatik –mit Vertiefung im Bereich mobile Anwendungen- zu kombinieren und erfolgreich in die berufliche Praxis zu übertragen und anzuwenden. Absolvent/-innen des Studiengangs sind qualifiziert eine wissenschaftliche ausgerichtete Berufstätigkeit im informationstechnischen Bereich in Wirtschaft, Technik und Verwaltung aufzunehmen sowie sich auch wissenschaftlich weiterführend mit einem Master-Studium zu qualifizieren.

Kompetenzen

Absolventinnen und Absolventen des Studiengangs sind in der Lage Fragestellungen der Informatik methodisch-grundlagenorientiert zu analysieren, zu erklären sowie Lösungsansätze zu entwickeln, aufzuzeigen und kritisch zu reflektieren.

Mit der Ausrichtung im Bereich mobiler und verteilter Anwendungen beherrschen die Absolvent/-innen informationswissenschaftliche Aspekte des Ubiquitous Computing und sind befähigt software-technische Systeme neu zu entwerfen, anzupassen und weiter zu entwickeln.

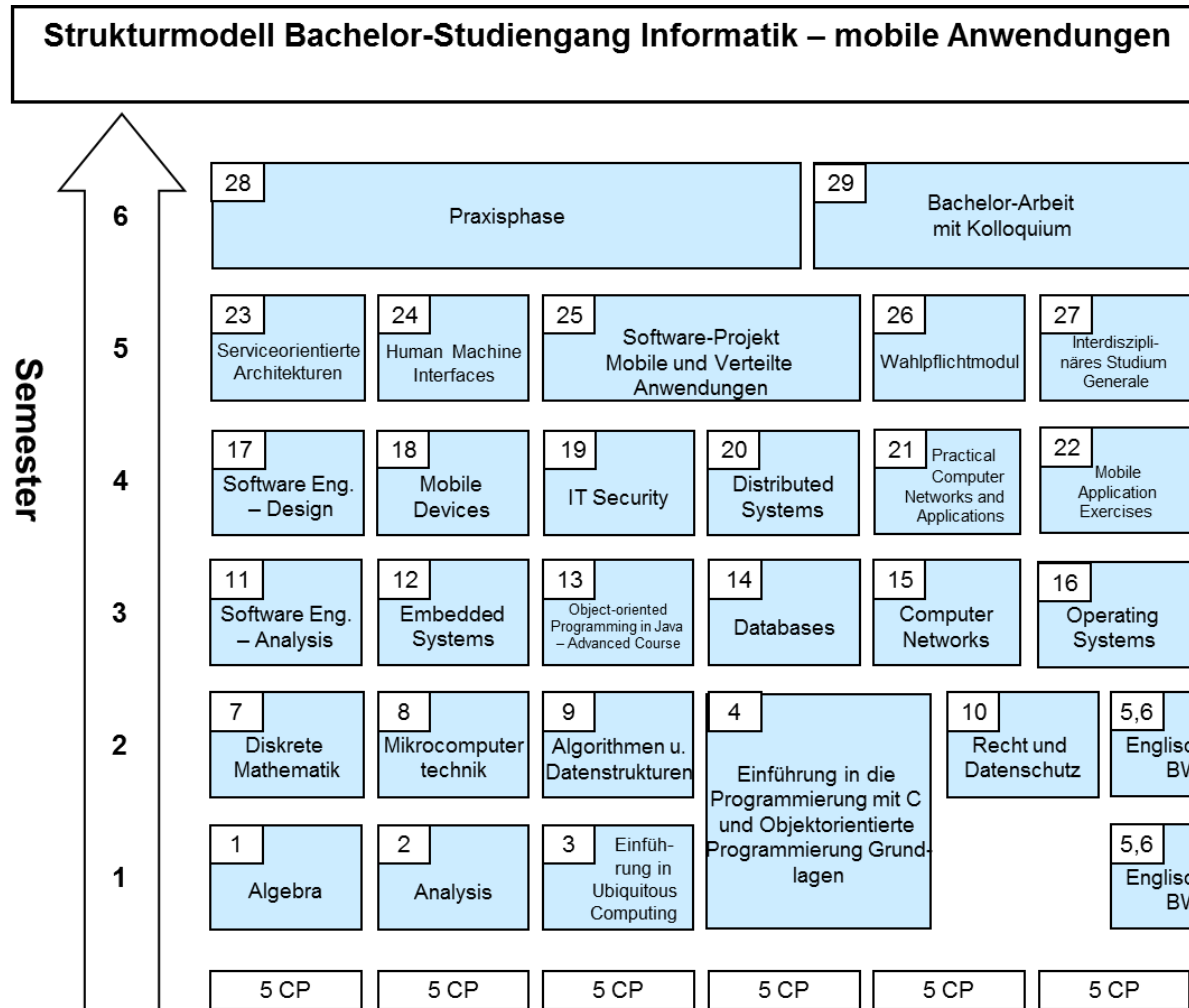
Sie verfügen insbesondere über folgende berufsbezogene Kompetenzen:

- abstrakte Methoden, Strukturen, und Pattern, die die Grundlage der Informatik darstellen anzuwenden,
- die Gesetzmäßigkeiten der Informatik und deren zugrunde liegende Prinzipien zu verstehen,
- die geringen Ressourcen von mobilen Systemen zu managen,
- moderne Verfahren der Informatik und deren Umsetzung in Theorie und Praxis mit geeigneten Werkzeugen anzuwenden,
- unterschiedliche Frameworks, insbesondere solche für eingebettete Systeme und Mobil-Devices zu verwenden,
- umfangreiche Informationssysteme zu gestalten und zu realisieren,
- „Embedded Systeme“ zu konzeptionieren und zu implementieren,
- Mobil-Devices zu implementieren,
- Applikationen für Smartphones und Tablet-PCs (“Apps“) zu entwickeln und zu programmieren,
- bestehende Softwaresysteme zu analysieren und zu verstehen,
- Beratung und Unterstützung in informationstechnischen Fragen zu leisten
- Geschäftsprozesse aus betriebswirtschaftlicher Perspektive bis zu deren Umsetzung ganzheitlich begleiten.

Neben den fachlichen Kompetenzen haben sie die Fähigkeit und Bereitschaft zur Kommunikation und Zusammenarbeit in nationalen und internationalen Teams sowohl mit Fachkollegen als auch mit Anwendern von informationstechnischen Systemen erworben und können Inhalte und Probleme der Informatik im Austausch mit Fachexperten und Laien in deutscher und englischer Sprache fachlich vertreten und präsentieren. Mit Vorträgen unter Nutzung moderner Präsentationstechniken sowie dem Verfassen wissenschaftlicher Berichte und Stellungnahmen sind sie vertraut.

Hierbei sind sie sensibilisiert die Gesetzmäßigkeiten und deren zugrunde liegenden Prinzipien der Informatik zu erkennen und zu berücksichtigen und haben ein Bewusstsein für die Auswirkungen ihrer Tätigkeit auf die Gesellschaft entwickelt. Mit ethischen Grundsätzen ihrer Tätigkeit sind sie vertraut.

2. Empfohlener Studienverlauf



3. ECTS-/Workload-Übersicht

Nr.	Modultitel	Cp ECTS	Dauer [Sem.]	Prüfungsform	Sprache	Gew.
1	Algebra	5	1	Klausur (90 Minuten)	Deutsch	1/48
2	Analysis	5	1	Klausur (90 Minuten)	Deutsch	1/48
3	Einführung in Ubiquitous Computing	5	1	Klausur (90 Minuten)	Deutsch	1/48
4	Einführung in die Programmierung mit C und Objekt-orientierte Programmierung Grundlagen	15	2	TPL 1: Klausur am Rechner (120 Minuten), Gewichtung 50% TPL 2: Klausur am Rechner (120 Minuten), Gewichtung 50 %	Deutsch	1/16
5	Betriebswirtschaftslehre	5	1	Klausur (90 Minuten)	Deutsch	1/48
6	Englisch	5	1	Klausur (90 Minuten)	Englisch	1/48
7	Diskrete Mathematik	5	1	Klausur (90 Minuten)	Deutsch	1/48
8	Mikrocomputer Technik	5	1	Klausur (90 Minuten)	Deutsch	1/48
9	Algorithmen und Datenstrukturen	5	1	Klausur (90 Minuten)	Deutsch	1/48
10	Recht und Datenschutz	5	1	Klausur (90 Minuten)	Deutsch	1/48
11	Software Engineering - Analysis	5	1	Computer-based project with documentation (submission period 9 weeks, processing time 60 hours) Pass/fail	Englisch	Keine
12	Embedded Systems	5	1	Klausur (90 Minuten)	Englisch	1/36
13	Object-oriented Programming	5	1	Projektarbeit	Englisch	1/36
14	Databases	5	1	Klausur (120 Minuten)	Englisch	1/36
15	Computer Networks	5	1	Klausur (90 Minuten)	Englisch	1/36

Nr.	Modultitel	Cp ECTS	Dauer [Sem.]	Prüfungsform	Sprache	Gew.
16	Operating Systems	5	1	Klausur (90 Minuten)	Englisch	1/36
17	Software Engineering - Design	5	1	Klausur (120 Minuten)	Englisch	5/144
18	Mobile Devices	5	1	Projektarbeit	Englisch	5/144
19	IT-Security	5	1	Klausur (90 Minuten)	Englisch	5/144
20	Distributed Systems	5	1	Klausur (90 Minuten)	Englisch	5/144
21	Practical Computer Networks and Applications	5	1	Computer-based project (submission period 9 weeks) with documentation (processing time 60 hours) Pass/fail	English	Keine
22	Mobile Application Exercises	5	1	Projektarbeit	Englisch	5/144
23	Serviceorientierte Architekturen	5	1	Klausur (90 Minuten)	Deutsch	5/144
24	Human Machine Interfaces	5	1	Projektarbeit	Deutsch	5/144
25	Software-Projekt Mobile und Verteilte Anwendungen	10	1	Projektarbeit	Deutsch	1/12
26.1	WP Datenbankadministration	5	1	TPL1: Vortrag TPL2: Klausur (90 Min.)	Deutsch	5/144
26.2	WP Ausgewählte Probleme aus dem ACM Programming Contest	5	1	TPL1: Hausarbeit TPL2: Mündl. Prüfung	Deutsch	5/144
26.3	WP Fahrzeug zu Fahrzeug Kommunikation	5	1	Projektarbeit	Deutsch	5/144
27	Interdisziplinäre Studium Generale	5	1	Projektarbeit	Deutsch	4/144
28	Praxisphase	18	1	Bericht und Vortrag	Deutsch	10/144
29	Bachelor-Arbeit mit Kolloquium	12	1	Bachelor-Arbeit und Kolloquium	Deutsch	22/144

4. Modulbeschreibungen

Modulbeschreibung zum Modul 1: Algebra

Modultitel	Algebra
Modulnummer	M1
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	1. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Keine
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Nach Abschluss des Moduls können die Studierenden</p> <ul style="list-style-type: none"> - abstrakte mathematische Begriffe definieren - strukturelle und logische Grundlagen elektronischer Informationsverarbeitung erläutern und in Beziehung zueinander setzen - sich abstrakte Begriffe selbständig erarbeiten und grundlegende Techniken oder Verfahren der Algebra aneignen. - die wichtigsten Begriffe, Strukturen und Methoden der elementaren Algebra und linearen Algebra, insbesondere die algebraischen Grundstrukturen erläutern, in Berechnungen anwenden sowie deren Bedeutung als Grundlage formaler Strukturen der Informatik bewerten <p>Vektor- und Matrizenrechnung sowie Theorie und Anwendungen linearer Abbildungen samt deren Darstellung über verschiedene Klassen von Matrizen erläutern, in Berechnungen anwenden sowie Eigenwerte als wesentliches Charakteristikum von Matrizen anführen und einordnen</p>
Inhalte des Moduls	Vorlesung Algebra Übung Algebra
Lehrformen des Moduls	Vorlesungen und Übungen
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Wintersemester
Modulkoordination	Prof. Dr. Ruth Schorr

Unitbeschreibung zum Modul 1: Vorlesung Algebra

Name der Unit	Vorlesung Algebra
Code	VAlgebra
Name des Moduls	Algebra (1)
Inhalte der Unit	<p>Grundlagen</p> <ul style="list-style-type: none"> • Aussagen- und Prädikatenlogik, Beweismethoden • Mengen, Relationen inkl. Funktionen <p>Elementare Zahlentheorie</p> <ul style="list-style-type: none"> • Primzahlen und Teiler • Modulo Rechnung, Diophantische Gleichungen, Chinesischer Restsatz • Anwendung Kryptographie <p>Algebraische Strukturen</p> <ul style="list-style-type: none"> • Halbgruppen, Gruppen, Permutationsgruppen, Untergruppen, Satz von Lagrange • Ringe, Polynomringe, Nullteiler, Testklassenring • Körper, Primzahlkörper <p>Lineare Algebra</p> <ul style="list-style-type: none"> • Vektorräume • Lineare Gleichungssysteme • Matrizen: Rang, Matrixprodukt, Inverse, Eigenwerte und charakteristisches Polynom • Lineare Abbildungen: Kern, Basis, Basiswechsel <p>Spezielle lineare Abbildungen: Spiegelungen, Projektionen, Drehungen sowie Matrixdarstellung und Komposition von Abbildungen</p>
Lehrformen	Vorlesung
SWS der Unit	4 h
Workload (h)	100 h
Anteil der Präsenzzeit	60 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	30 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr, Thekla Unthan, Prof. Dr. Karsten Weronek

Basis – Literatur	<ul style="list-style-type: none"> • Brill, Manfred: Mathematik für Informatiker, Hanser, 2005 • Denecke, Klaus: Algebra und Diskrete Mathematik für Informatiker, Vieweg + Teubner, 2003 • Teschl, Gerold und Susanne: Mathematik für Informatiker, Band 1 Diskrete Mathematik und Lineare Algebra, Springer 2008 • Witt, Kurt-Ulrich: Algebraische Grundlagen der Informatik, Vieweg, 2007 <p>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 1: Übung Algebra

Name der Unit	Übung Algebra
Code	ÜAlgebra
Name des Moduls	Algebra (1)
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen.
Lehrformen	Übung
SWS der Unit	2 h
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr, Thekla Unthan, Prof. Dr. Karsten Weronek
Basis – Literatur	Siehe Unit Vorlesung Algebra
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Modulbeschreibung zum Modul 2: Analysis

Modultitel	Analysis
Modulnummer	M2
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Informatik (B. Sc.)
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	1. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	keine
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Nach Abschluss des Moduls können die Studierenden</p> <ul style="list-style-type: none"> - die wichtigsten Begriffe, Verfahren und Techniken der Differential- und Integralrechnung einschließlich Potenzreihen sowie komplexe Zahlen benennen und in Berechnungen anwenden - die typischen Methoden der Analysis unterscheiden - einfache Anwendungsprobleme in mathematische Aufgabenstellungen umsetzen und diese lösen - die Voraussetzungen und Grenzen der Methoden der Differential- und Integralrechnung erörtern - die zentralen Aussagen zur Konvergenz von Folgen, Reihen und Funktionen erläutern und damit Beispiele behandeln - die wichtigsten Eigenschaften der elementaren Funktionen wiedergeben - das Konzept der Stetigkeit und Differenzierbarkeit beschreiben und daraus Eigenschaften von Funktionen herleiten <p>eindimensionale (auch unentgeltliche) Integrale berechnen und die zugrunde liegende Theorie erläutern</p>
Inhalte des Moduls	Vorlesung Analysis Übung Analysis
Lehrformen des Moduls	Vorlesungen und Übungen
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Wintersemester
Modulkoordination	Prof. Dr. Egbert Falkenberg

Unitbeschreibung zum Modul 2: Vorlesung Analysis

Name der Unit	Vorlesung Analysis
Code	
Name des Moduls	Analysis (2)
Inhalte der Unit	<ul style="list-style-type: none"> • Reelle Zahlen: Folgen, Reihen, Konvergenz • Reelle Funktionen: Grundbegriffe, Grenzwerte, Stetigkeit, Zwischenwertsatz • Differenzialrechnung einer Veränderlichen: Differenzierbarkeit, Ableitungsregeln, Mittelwertsatz, Regeln von de l'Hospital, Taylor'scher Satz, Geometrische Bedeutung der Ableitungen, Kurvendiskussion • Integralrechnung: Integrierbarkeit, Stammfunktionen und Hauptsatz der Differenzial- und Integralrechnung, Elementare Integrationstechniken, Uneigentliche Integrale, Länge, Flächeninhalt, Volumen • Komplexe Zahlen: Definition und Darstellung, Rechenregeln, Potenzen, Wurzeln, Polynome, Fundamentalsatz der Algebra • Potenzreihen: Unendliche Reihen, Potenzreihen und Konvergenz, Taylor-Reihen, Eigenschaften • Näherungsverfahren zur Lösung nichtlinearer Gleichungen und Integralen: Heron'sches-, Bisektion-, Newton-, Sekanten-Verfahren, Trapezregel
Lehrformen	Vorlesung
SWS der Unit	4 h
Workload (h)	100 h
Anteil der Präsenzzeit	60 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	30 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Prof. Dr. Doina Logofatu, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr
Basis – Literatur	<p>Mathematik für das Ingenieurstudium, Jürgen Koch, Martin Stämpfle, Hanser, 4. neu bearbeitete Auflage, 2018</p> <p>Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 2: Übung Analysis

Name der Unit	Übung Analysis
Code	
Name des Moduls	Analysis (2)
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen.
Lehrformen	Übung
SWS der Unit	2 h
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Prof. Dr. Doina Logofatu, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr
Basis – Literatur	Siehe Unit Vorlesung Analysis
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Modulbeschreibung zum Modul 3: Einführung in Ubiquitous Computing

Modultitel	Einführung in Ubiquitous Computing
Modulnummer	M3
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	1. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Vorausgesetzt werden die zum Erwerb der Fachhochschulreife erwarteten Kenntnisse der Mathematik
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Nach erfolgreichem Abschluss können die Studierenden :</p> <ul style="list-style-type: none"> • die unterschiedlichen Bereiche der Informatik beschreiben, • die Verarbeitung und Berechnung von Zahlen auf der Hardware-Ebene durchführen und systematisch beschreiben, • den Aufbau und die Wirkungsweise eines Mikroprozessors darlegen und erläutern, • die Vorgänge im Rechner bei Programmerstellung und Programmabläufen darstellen und begründen, • die Besonderheiten von mobilen Anwendungen nennen und erläutern, • die Fähigkeiten und Anwendungen von „Überall Rechner“ einordnen, • wissenschaftlichen Definitionen und Begriffen ggf. zu erläutern und zu verwenden. <p>Neben den fachlichen Aspekten sind sie in der Lage Prozesse und Technik unter Bedingungen der gesellschaftlichen Verantwortung zu erläutern.</p>
Inhalte des Moduls	<p>Vorlesung Einführung in Ubiquitous Computing</p> <p>Übung Einführung in Ubiquitous Computing</p>

Lehrformen des Moduls	Vorlesung und Übung
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Jährlich
Modulkoordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul 3: Vorlesung Einführung in Ubiquitous Computing

Name der Unit	Vorlesung Einführung in Ubiquitous Computing
Code	
Name des Moduls	Einführung in Ubiquitous Computing
Inhalte der Unit	<p>Das Modul liefert als Einführung in die Informatik und Ubiquitous Computing allgemeine und grundlegende Kenntnisse und Fähigkeiten für das gesamte Studium.</p> <ul style="list-style-type: none"> • Informationsdarstellung im Rechner (Speicherformate, Logische Größen, Zeichen, Zahlen, Zahlensysteme, Ganze Zahlen, Gleitkommazahlen) • Einführung in Rechnerarchitekturen • Verarbeitung von Zahlen • Betriebssysteme (Überblick, Aufgaben, Klassifikation, Hauptkomponenten) • Betriebssysteme für Embedded Systeme (Grundlagen) • Massenspeicher • Sensorik (Grundlagen) • Funktionsweise eines Mikroprozessors • Herausforderungen für mobile Anwendungen • Ressourcennutzung (Speicher, Rechenzeit, Stromverbrauch) • „Überall-“ Rechnersysteme
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	5 h
Anteil Selbststudium	15 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literatur	<ul style="list-style-type: none"> • Herold, Lurz, Wohrab: Grundlagen der Informatik: praktisch, technisch, theoretisch, Pearson Studium, 2012 • Schiffmann/Schmitz: Technische Informatik Bd. 1 und 2, Berlin, Springer 2001 • Weiser M. (1991) The Computer for the 21st Century. Scientific American 265(3): 94–104 • Friedemann Mattern (Herausgeber): Die Informatisierung des

	Alltags: Leben in smarten Umgebungen, Springer 2007 • John Krumm (Editor): Ubiquitous Computing Fundamentals, Chapman and Hall/CRC, 2009
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 3: Übung Einführung in Ubiquitous Computing

Name der Unit	Übung Einführung in Ubiquitous Computing
Code	
Name des Moduls	Einführung in Ubiquitous Computing
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen. Um dem besonderen Fokus des Studiengangs Informatik- mobile Anwendungen Rechnung zu tragen, werden Beispiele auch aus dem Bereich Mobile Endgeräte und Embedded Systems verwendet.
Lehrformen	Übung
SWS der Unit	2
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	
Anteil Selbststudium	20 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literatur	Siehe Unit Vorlesung Einführung in Ubiquitous Computing
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Modulbeschreibung zum Modul 4: Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen

Modultitel	Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen
Modulnummer	4
Modulcode	
Studiengang	Informatik – Mobile Anwendungen (B.Sc.)
Verwendbarkeit des Moduls	Informatik (B.Sc.)
Dauer des Moduls	zwei Semester
Empfohlenes Semester im Studienverlauf	1. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	15 cp / 450 h
Empfohlene inhaltliche Vorkenntnisse	Es werden keine Programmierkenntnisse vorausgesetzt
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Übungen am Rechner (Gesamtaufwand 30 Stunden) Voraussetzung für die Teilprüfungsleistung 2: bestandene Teilprüfungsleistung 1
Modulprüfung	Teilprüfungsleistung 1: Eigenständige Programmierung C in Form einer Klausur am Rechner (120 Minuten), Gewichtung: 50% Teilprüfungsleistung 2: Eigenständige Objektorientierte Programmierung in Form einer Klausur am Rechner (120 Minuten), Gewichtung: 50%
Lernergebnisse und Kompetenzen	Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"> • die wichtigsten Sprachelemente und Bibliotheksfunktionen benennen und unterscheiden • Lösungen für einfache Aufgabenstellungen als strukturierten Entwurf formulieren und in C umsetzen • Methoden zur Fehlererkennung und Fehlerbeseitigung fallbezogen einsetzen • Denk- und Herangehensweisen der objektorientierten Programmierung skizzieren und kritisch vergleichen Begriffe wie Datenkapselung, Wiederverwendung von Code, Klassen, Vererbung und Polymorphie erläutern und einordnen
Inhalte des Moduls	Vorlesung Einführung in die Programmierung mit C Übung Einführung in die Programmierung mit C Vorlesung Objektorientierte Programmierung Grundlagen Übung Objektorientierte Programmierung Grundlagen

Lehrformen des Moduls	Vorlesung und Übungen
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Wintersemester
Modulkoordination	Prof. Dr. Baris Sertkaya
Hinweise	Die Modulteilprüfung C erfolgt im 1. Semester des/der Studierenden. Wiederholungsprüfungen sind gegebenenfalls in dem auf das Nichtbestehen folgenden Semester abzulegen

Unitbeschreibung zum Modul 4: Vorlesung Einführung in die Programmierung mit C

Name der Unit	Vorlesung Einführung in die Programmierung mit C
Code	
Name des Moduls	Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen (4)
Inhalte der Unit	<ul style="list-style-type: none"> • Einführendes Beispiel mit grundlegenden Sprachelementen • Analyse einfacher Aufgabenstellungen und Erstellen eines strukturierten Lösungsvorschlags • Editieren, Übersetzen, Ausführen von Programmen • Elementare Datentypen, Variablen und Arithmetik • Ein-/Ausgabe Verzweigung und Schleifen • Felder, Zeichenkette • Fehlersuche und Fehlerbeseitigung • Zeiger, dynamische Speicherverwaltung • Unterprogramme (Funktionen) und Parameter, modularer Programmaufbau, • Bibliotheksfunktionen Dateien Strukturierte Datentypen
Lehrformen	Vorlesung
SWS der Unit	2 h
Workload (h)	100 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	60 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Baris Sertkaya, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Carsten Biemann
Basis – Literatur	<ul style="list-style-type: none"> • Brian W. Kernighan, Dennis Ritchie, The C Programming Language, Prentice Hall Software, 2000 • Erlenkötter, H., C Programmieren von Anfang an, Rowohlt, 2008 • Mittelbach, H., Einführung in C, Fachbuchverlag Leipzig, 2002 • Die Programmiersprache C. Ein Nachschlagewerk, Regionales Rechenzentrum für Niedersachsen/Universität Hannover, 1RRZN. • Das Handbuch zu den zum GNU-C-Compiler gehörenden Standardfunktionen (GNU C Library) kann man einsehen unter http://www.gnu.org/software/libc/manual/ <p>• Weitere Literaturhinweise können zu Beginn des Semesters</p>

	bekanntgegeben werden.
Art und Form des Leistungsnachweises der Unit	TPL1: Eigenständige Programmierung C in Form einer Klausur am Rechner (120 Minuten) , Gewichtung 50%
Bewertung des Leistungsnachweises der Unit	differenziert

Unitbeschreibung zum Modul 4: Übung Programmieren in C

Name der Unit	Übung Programmieren in C
Code	
Name des Moduls	Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen (4)
Inhalte der Unit	Übungen zur Einführung in die Programmierung, Umsetzung von Aufgaben mittels eines C-Programms, Feedback zum Lernprozess
Lehrformen	Übung
SWS der Unit	2 h
Workload (h)	125 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	95 h
Anteil Praxiszeit	50 h (einschl. Präsenzzeit)
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Baris Sertkaya, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Carsten Biemann
Basis – Literatur	Siehe Unit Vorlesung Einführung in die Programmierung mit C
Art und Form des Leistungsnachweises der Unit	Übungen am Rechner
Bewertung des Leistungsnachweises der Unit	bestanden / nicht bestanden
Hinweise	

Unitbeschreibung zum Modul 4: Vorlesung Objektorientierte Programmierung mit C und Objektorientierte Programmierung Grundlagen

Name der Unit	Vorlesung Objektorientierte Programmierung Grundlagen
Code	
Name des Moduls	Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen (4)
Inhalte der Unit	<p>Inhaltlichen Schwerpunkt bilden:</p> <ul style="list-style-type: none"> • Objektorientierte Ansatz – erstes Beispiel; evtl. Abgrenzung zur prozeduralen Programmierung • Klassenkonzept, UML-Darstellung • Klassen, Objekte • Konstruktor incl. Überladung, Destruktor • Kopieren und Zuweisen von Objekten • Operatoren, dynamische Speicherverwaltung • Referenzen • Setter/Getter Methoden • Statische Objektkomponenten • Vererbung, Zugriffsrechte • Polymorphie, spätes Binden <p>Weiterhin bieten sich ausgewählte Themen an, z.B.</p> <ul style="list-style-type: none"> • Mehrfachvererbung, virtuelle Vererbung • Klassen- und Funktions-Templates <p>Standard Libraries und Standard Template Libraries</p>
Lehrformen	Vorlesung
SWS der Unit	2 h
Workload (h)	100 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	60 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Baris Sertkaya, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Carsten Biemann
Basis – Literatur	<ul style="list-style-type: none"> • Stroustrup, Bjarne. Die C++ Programmiersprache, Addison Wesley • Breyman, Ulrich C++ Einführung und professionelle Programmierung, Carl Hanser Verlag

	<ul style="list-style-type: none">• RRZN- Handbuch. C++ für C Programmierer. 13. Auflage, RRZN Hannover• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.
Art und Form des Leistungsnachweises der Unit	TPL2: Eigenständige Objektorientierte Programmierung in Form einer Klausur am Rechner (120 Minuten), Gewichtung 50%
Bewertung des Leistungsnachweises der Unit	Differenziert
Hinweise	

Unitbeschreibung zum Modul 4: Übung Objektorientierte Programmierung Grundlagen

Name der Unit	Übung Objektorientierte Programmierung Grundlagen
Code	
Name des Moduls	Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen (4)
Inhalte der Unit	Beispiele und Programmieraufgaben mit Bezug zu den Vorlesungsinhalten, qualifiziertes Feedback durch die Lehrenden
Lehrformen	Übung
SWS der Unit	2 h
Workload (h)	125 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	95 h
Anteil Praxiszeit	50 h (einschl. Präsenzzeit)
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Baris Sertkaya, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Carsten Biemann
Basis – Literatur	Siehe Unit Vorlesung Objektorientierte Programmierung Grundlagen
Art und Form des Leistungsnachweises der Unit	Übungen am Rechner
Bewertung des Leistungsnachweises der Unit	bestanden / nicht bestanden
Hinweise	

Modulbeschreibung zum Modul 5: Betriebswirtschaftslehre

Modultitel	Betriebswirtschaftslehre
Modulnummer	M5
Modulcode	
Studiengang	Informatik – Mobile Anwendungen (B.Sc.)
Verwendbarkeit des Moduls	Informatik (B.Sc.)
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	1. oder 2. Semester, alternativ zu dem Modul 6: Englisch
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Keine
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur über 90 Minuten
Lernergebnisse und Kompetenzen	<p>Nach Abschluss des Moduls können die Studierenden:</p> <ul style="list-style-type: none"> • Grundbegriffe und Prinzipien aus den folgenden Bereichen benennen und erläutern: <ul style="list-style-type: none"> o wirtschaftliches Handeln o Aufbau- und Ablauforganisation o Produktion und Logistik o Finanzwesen und Controlling o Personalwesen, • ausgehend von betrieblichen Funktionsbereichen die Verbindung zur informationstechnologischen Unterstützung innerbetrieblich sowie zwischenbetrieblich herstellen und einordnen • wichtige Anwendungsfelder der Informatik in Beziehung setzen zu Unternehmensabläufen und Gesellschaft • Geschäftsprozesse skizzieren und mit geeigneten Verfahren analysieren
Inhalte des Moduls	Vorlesung Betriebswirtschaftslehre
Lehrformen des Moduls	Vorlesung
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Jedes Semester
Modulkoordination	Jürgen Jung

Unitbeschreibung zum Modul 5: Vorlesung Betriebswirtschaftslehre

Name der Unit	Vorlesung Betriebswirtschaftslehre
Code	
Name des Moduls	Betriebswirtschaftslehre (5)
Inhalte der Unit	<ul style="list-style-type: none"> • Grundlagen der Betriebswirtschaftslehre • Betriebliche Aufbauorganisation • Betriebliche Ablauforganisation • Ausgewählte betriebliche Geschäftsprozesse • Betriebliche Funktionsbereiche (Finanzwesen, Controlling, Marketing, Personalwesen, Produktion, Logistik) <p>Typische IT-Systeme</p>
Lehrformen	Vorlesung
SWS der Unit	4 h
Workload (h)	150 h
Anteil der Präsenzzeit	60 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	80 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	N.N.
Basis – Literatur	<ul style="list-style-type: none"> • Domschke, W. Et al.: „Grundlagen der Betriebswirtschaftslehre“, Springer Verlag, Berlin • Härder, J.: „Betriebswirtschaft für Ingenieure“, Fachbuchverlag, Leipzig • Laudon, K. C. et al.: „Wirtschaftsinformatik“, Pearson Studium, München • Ott: „Betriebswirtschaftslehre für Ingenieure und Informatiker“, München • Wöhe, G: „Einführung in die Allgemeine BWL“, Vahlen Verlag, München <p>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine
Hinweise	

Modulbeschreibung zum Modul 6: Englisch

Module title	Englisch
Module number	M6
Module code	
Study program	Informatik – Mobile Anwendungen
Module usability	Applicable to other study programmes. The module helps to pre-prepare students for the English-medium lectures and other classes of the 3rd and 4th semesters and promotes adequate participation in these.
Module duration	one semester
Recommended semester	1st or 2 nd semester: For reasons of capacity, the module may be scheduled either in the 1st or 2nd semester.
Type of module	compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	none
Module prerequisites	none
Module examination requirements	Presentation in English (15 minutes) on the basis of language practice sessions and group discussions in class; active participation in the presentations of the others. The presentation is graded.
Module examination	written examination (90 minutes)
Learning outcomes and skills	<p>Students can cope with the general requirements of professional communication in English; they can handle typical professional situations of international communication with both specialists of their own field and non-specialists; they can follow the English-medium lectures and other classes of the 3rd and 4th semesters adequately.</p> <p>By promoting the students' competence in English, this module also contributes to the development of non-subject-specific skills (key skills).</p> <p>Presentation skills, writing skills; verbal communication; subject-specific vocabulary and terminology.</p>
Module contents	English classes; supervised e-learning
Module teaching methods	English practice sessions including graded presentation in English; Supervised e-learning (1 SWS), mainly in the second half of the semester as additional preparation for the written exam
Module language	Englisch
Module availability	every Semester
Module coordination	Hartwell (Fachsprachenzentrum)
Comments	Der erfolgreiche Abschluss des Moduls „Englisch“ ist Voraussetzung für die Teilnahme an den englischsprachigen Veranstaltungen des 3. und 4. Semesters.

Unitbeschreibung zum Modul 6: Lecture Englisch

Unit title	English classes
Code	
Module title	English
Unit contents	<ul style="list-style-type: none"> – Professional communication in English, both with specialists of the same field and non-specialists; – Presentation skills; – Active listening skills and reading comprehension; – Subject-specific vocabulary and terminology; – Verbal communication and writing skills.
Teaching forms	English classes – practice sessions in listening, speaking, reading, writing, grammar; working alone, in pairs, small groups and in plenum, with communication of the results in class; supervised e-learning.
Semester periods (hours) per week	4
Workload (h)	150 h
Class hours	60 h + 15 h e-learning
Total time of examination incl. preparation (h)	
Total time of individual study (h)	75 h
Total time of practical training (h)	
Unit language	English
Lecturer	Permanent staff + freelance teachers of the University Language Center
Recommended reading	Up-to-date learning materials are provided in class; any coursebook used is indicated at the beginning of the semester.
Assessment type and form	Presentation in English (15 minutes) on the basis of language practice sessions and group discussions in class; active participation in the presentations of others.
Assessment grading	Graded
Comments	

Modulbeschreibung zum Modul 7: Diskrete Mathematik

Modultitel	Diskrete Mathematik
Modulnummer	M7
Modulcode	
Studiengang	Informatik – Mobile Anwendungen (B.Sc.)
Verwendbarkeit des Moduls	Informatik (B.Sc.)
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	2. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Die Inhalte des Moduls Algebra und Analysis des ersten Studienseesters werden vorausgesetzt.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Nach Abschluss des Moduls können die Studierenden:</p> <ul style="list-style-type: none"> - aufbauend auf den Modulen Algebra und Analysis auf vertiefter Ebene mit abstrakten Begriffen operieren - die wichtigsten mathematischen Techniken für Anwendungen in den Kerndisziplinen der Informatik (Theoretische Informatik und Rechnerarchitektur, Betriebssysteme, Rechnernetzen etc...) in Übungen anwenden - Begriffe und Verfahren der diskreten Mathematik anwenden - Lösungsverfahren der diskreten Mathematik in einfachen Anwendungsfällen selbständig einsetzen und ihre Ergebnisse bewerten - Bezüge der diskreten Mathematik zu Kerndisziplinen der Informatik herstellen und Verfahren der diskreten Mathematik in diesen Kontexten adäquat anwenden - auf vertieftem Niveau mit formalen Systemen und Modellen umgehen
Inhalte des Moduls	Vorlesung Diskrete Mathematik, Übung Diskrete Mathematik
Lehrformen des Moduls	Vorlesung und Übungen
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Sommersemester
Modulkoordination	Prof. Dr. Ruth Schorr

Unitbeschreibung zum Modul 7: Vorlesung Diskrete Mathematik

Name der Unit	Vorlesung Diskrete Mathematik
Code	
Name des Moduls	Diskrete Mathematik (7)
Inhalte der Unit	<p>Abzählungen</p> <ul style="list-style-type: none"> • Zählprinzipien • Zählkoeffizienten: Variationen, Kombinationen, Mengen- und Zahlpartitionen, Stirling Zahlen • Catalan-Zahlen • Lösung von linearen, homogenen und inhomogenen Rekursionsgleichungen <p>Einführung in die Codierungstheorie</p> <ul style="list-style-type: none"> • Grundbegriffe: Codierung, Decodierung, Fehlererkennung, Hamming und Singleton Schranke • Lineare Codes: Generator- und Kontrollmatrix, Hamming-Code, Reed-Solomon-Code <p>Graphentheorie</p> <ul style="list-style-type: none"> • Grundbegriffe • Euler'sche und Hamilton'sche Graphen, planare Graphen, Färbungen • Bäume inkl. Binäre Bäume • Netzwerkalgorithmen: Maximale Spannbäume, Kürzeste Wege, Maximale Flüsse <p>Matching in bipartiten Graphen</p>
Lehrformen	Vorlesung
SWS der Unit	4 h
Workload (h)	100 h
Anteil der Präsenzzeit	60 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	30 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr
Basis – Literatur	<ul style="list-style-type: none"> • Aigner, Martin: Diskrete Mathematik, Vieweg + Teubner, 2006 • Diestel, Reinhard: Graphentheorie, Springer, 2010

	<ul style="list-style-type: none">• Grimaldi, Ralph P.: Discrete and Combinatorial Mathematics, Addison Wesley, 2003• Steger, Angelika: Diskrete Strukturen 1, Springer, 2007• Teschl, Gerold und Susanne: Mathematik für Informatiker, Band 1 Diskrete Mathematik und Lineare Algebra, Springer, 2008 <p>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
Art und Form des Leistungsnachweises der Unit	n/a
Bewertung des Leistungsnachweises der Unit	n/a
Hinweise	

Unitbeschreibung zum Modul 7: Übung Diskrete Mathematik

Name der Unit	Übung Diskrete Mathematik
Code	
Name des Moduls	Diskrete Mathematik (7)
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen.
Lehrformen	Übung
SWS der Unit	1 h
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr
Basis – Literatur	Siehe Vorlesung Unit Vorlesung Diskrete Mathematik
Art und Form des Leistungsnachweises der Unit	n/a
Bewertung des Leistungsnachweises der Unit	n/a
Hinweise	

Modulbeschreibung zum Modul 8: Mikrocomputer Technik

Modultitel	Mikrocomputer Technik
Modulnummer	M8
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	2. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Keine
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Labortestat: schriftlicher Bericht (80h inkl. Durchführung Labor)
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Die Studierenden erwerben theoretische und praktische Kompetenzen in den folgenden Feldern:</p> <p>Digitaltechnik :</p> <ul style="list-style-type: none"> • Sie ordnen elektrotechnische Randbedingungen beim Betrieb digitaler Schaltungen ein. • Sie können den systematischen Entwurf digitaler Schaltungen erläutern. <p>Rechnerarchitekturen:</p> <ul style="list-style-type: none"> • Sie können die wichtigsten architektonischen Prinzipien für den Entwurf von Rechenanlagen beschreiben. • Sie unterscheiden die verschiedenen Ebenen der Befehlsverarbeitung in Assemblersprachen. • Sie wenden die wichtigsten Prinzipien von Assemblersprachen an. <p>Microcomputertechnik:</p> <ul style="list-style-type: none"> • Sie können die Funktionsweise und den Aufbau von Mikrocomputern beschreiben. • Sie realisieren hardwarenahe Programme in Assembler und einer Hochsprache. • Sie zählen typische Anwendungsgebiete auf und entwickeln mikroprozessorgesteuerte Systeme.

	Außerdem werden die folgenden außerfachlichen Kompetenzen erworben: Strukturierte Problemlösung, kreative Problemlösung, Teamarbeit (bei Laborversuchen)
Inhalte des Moduls	Vorlesung Mikrocomputertechnik Labor Mikrocomputertechnik
Lehrformen des Moduls	Seminaristischer Unterricht, Labor
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Jährlich
Modulkoordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul Modul 8:Vorlesung Mikrocomputertechnik

Name der Unit	Vorlesung Mikrocomputertechnik
Code	
Name des Moduls	Mikrocomputertechnik
Inhalte der Unit	<p>Digitaltechnik: digitallogische Bausteine und Schaltungen, sequentielle Schaltungen.</p> <p>Aufbau und Design von Mikrocomputern. Funktionsweise eines Mikrocontrollers.</p> <p>Steuerwerk, Rechenwerk, Register, Ports, Programm- und Datenspeicher,</p> <p>Stackspeicher, Befehlsabarbeitung, Timing-Diagramm.</p> <p>Programmierung in Assembler: Befehlsarten, Special Function Register,</p> <p>Adressierungsarten, Ein-/Ausgabe von Daten mittels Ports, Unterprogramme,</p> <p>Tabellenbearbeitung, Echtzeitprogramme, Interrupts.</p> <p>Grundlagen der hardwarenahen Programmierung in einer Hochsprache.</p>
Lehrformen	Seminaristischer Unterricht
SWS der Unit	2
Workload (h)	70 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	30 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literatur	<p>Nauth: Embedded Intelligent Systems, Oldenbourg Verlag, 2005</p> <p>Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 8: Labor Mikrocomputertechnik

Name der Unit	Labor Mikrocomputertechnik
Code	
Name des Moduls	Mikrocomputertechnik
Inhalte der Unit	Durchführung von Laborversuchen in Assembler und in einer Hochsprache, die auf einem Mikrocontrollerboard ausgeführt werden.
Lehrformen	Laborübungen
SWS der Unit	2
Workload (h)	80 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	
Anteil Selbststudium	50 h
Anteil Praxiszeit	In Präsenzzeit enthalten
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literatur	Laboranleitung, siehe Unit Vorlesung
Art und Form des Leistungsnachweises der Unit	Labortestat: schriftlicher Bericht (80h inkl. Durchführung Labor)
Bewertung des Leistungsnachweises der Unit	Undifferenziert (bestanden/nicht bestanden)
Hinweise	Zur Teilnahme am Labor ist die rechtzeitige Anmeldung erforderlich. Näheres wird durch Aushang geregelt.

Modulbeschreibung zum Modul 9: Algorithmen und Datenstrukturen

Modultitel	Algorithmen und Datenstrukturen
Modulnummer	M9
Modulcode	
Studiengang	Informatik – Mobile Anwendungen (B.Sc.)
Verwendbarkeit des Moduls	Informatik – Mobile Anwendungen (B.Sc.)
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	2. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Module Einführung in die Programmierung mit C, Praxisorientierte Einführung in die Informatik, Mathematische Grundlagen – Algebra
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Die mit den Begriffen Algorithmus und abstrakte Datenstruktur verbundenen Kenntnisse sind zentral für die gesamte Informatik. Das Modul vermittelt somit die notwendigen Schlüsselqualifikationen für die berufliche Tätigkeit als Informatiker, zudem liefert es die Voraussetzungen zum Verständnis nahezu aller Folgekurse im Verlauf des Studiums. In den Folgemodulen Informatik sollen die hier vermittelten Begriffe und Techniken selbstverständlich und souverän eingesetzt werden können.</p> <p>Nach Abschluss des Moduls können die Studierenden</p> <ul style="list-style-type: none"> - Algorithmen und Datenstrukturen analysieren und dokumentieren - geeignete neue Datenstrukturen (aufbauend auf den in dem Kurs behandelten Standardstrukturen) gestalten - Algorithmen zur Bearbeitung entwickeln und nach den gelernten Methoden darstellen - Lösungsmöglichkeiten hinsichtlich Korrektheit, Komplexität und Eleganz beurteilen. - Durch die Analyse und Ausarbeitung von abstrakten Algorithmen werden wissenschaftliches und exaktes Arbeiten, sowie strukturierte und kreative Problemlösung eingeübt.
Inhalte des Moduls	Vorlesung Algorithmen und Datenstrukturen Übung Algorithmen und Datenstrukturen
Lehrformen des Moduls	Vorlesung und Übungen
Sprache	Deutsch

Häufigkeit des Angebots von Modulen	Sommersemester
Modulkoordination	Prof. Dr. Jörg Schäfer

Unitbeschreibung zum Modul 9: Vorlesung Algorithmen und Datenstrukturen

Name der Unit	Vorlesung Algorithmen und Datenstrukturen
Code	
Name des Moduls	Algorithmen und Datenstrukturen (9)
Inhalte der Unit	<p>Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden.</p> <p>Algorithmen:</p> <ul style="list-style-type: none"> • Definition eines Algorithmus und Anforderungen an ihn, semantische Korrektheit • Beschreibungsformen von Algorithmen (natürliche Sprache, Pseudocode, Struktogramm, Datenflußdiagramm, Programmablaufplan) • Kapazitätsbetrachtungen: Platz- und Rechenzeit, asymptotische Notationen, Kapazitätsmaße (worst case, average case), P-NP- Problematik • Typen algorithmischer Vorgehensweisen (Rekursion, Greedy, Divide and Conquer, Backtracking ...) • Grundlegende Begriffe der Parallelen Algorithmen: Work-Law, Span-Law, Speed-Up, Parallelism, Notwendigkeit für Synchronisierungsverfahren <p>Datenstrukturen:</p> <ul style="list-style-type: none"> • elementare Datenstrukturen • lineare Standardstrukturen (Felder, Listen, Stapel, Warteschlangen) • Bäume • Mengen • Graphen • Algorithmen zu den Grundproblemen der Informatik • Sortieren • Suchen <ul style="list-style-type: none"> • Hashing
Lehrformen	Vorlesung
SWS der Unit	4 h
Workload (h)	100 h
Anteil der Präsenzzeit	60 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	30 h

Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Justus Klingemann, Prof. Dr. Jens Liebehenschel, Prof. Dr. Jörg Schäfer, Prof. Dr. Barış Sertkaya
Basis – Literatur	<ol style="list-style-type: none"> 1. Aho, A. V., Hopcroft, J. E., Ullman, J. D.: The Design and Analysis of Computer Algorithms, Addison-Wesley 1974 2. Brunskill, D., Turner, J.: Understanding Algorithms and Data Structures, McGraw-Hill 1996 3. Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. Algorithmen - Eine Einführung. Oldenbourg Wissenschaftsverlag, 2. Auflage, 2007 4. Güting, R. H.: Datenstrukturen und Algorithmen, Teubner 1992 5. Lewis, T. G., Smith, M. Z.: Datenstrukturen und ihre Anwendung, Oldenbourg 1978 6. Mehlhorn, K.: Datenstrukturen und effiziente Algorithmen, Bd 1, 2, 3, Teubner 1986 7. Preparata, F. P., Shamos, M. I.: Computational Geometry, Springer 1985 8. T. Ottmann, P. Widmayer: Algorithmen und Datenstrukturen. BI Wissenschaftsverlag, Mannheim 9. H. Reß, G. Vorbeck: Datenstrukturen und Algorithmen. Hanser, München 10. Sedgewick, R.: Algorithms, Addison-Wesley 1984 <ul style="list-style-type: none"> • Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.
Art und Form des Leistungsnachweises der Unit	n/a
Bewertung des Leistungsnachweises der Unit	n/a
Hinweise	

Unitbeschreibung zum Modul 9: Übung Algorithmen und Datenstrukturen

Name der Unit	Übung Algorithmen und Datenstrukturen
Code	
Name des Moduls	Algorithmen und Datenstrukturen (9)
Inhalte der Unit	<ul style="list-style-type: none"> Aufgaben und Beispiele zu den Themen des Moduls.
Lehrformen	Übung
SWS der Unit	2 h
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Justus Klingemann, Prof. Dr. Jens Liebehenschel, Prof. Dr. Jörg Schäfer, Prof. Dr. Barış Sertkaya
Basis – Literatur	Siehe Unit Vorlesung Algorithmen und Datenstrukturen
Art und Form des Leistungsnachweises der Unit	n/a
Bewertung des Leistungsnachweises der Unit	n/a
Hinweise	

Modulbeschreibung zum Modul 10: Recht und Datenschutz

Modultitel	Recht und Datenschutz
Modulnummer	M10
Modulcode	
Studiengang	Informatik – Mobile Anwendungen (B.Sc.)
Verwendbarkeit des Moduls	Informatik (B.Sc.)
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	2. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Vorausgesetzt werden die zum Erwerb der Fachhochschulreife erwarteten Kenntnisse im Bereich Recht, Politik oder Gesellschaftslehre.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur (90 Minuten)
Lernergebnisse und Kompetenzen	<p>Nach Abschluss des Moduls können die Studierenden</p> <ul style="list-style-type: none"> - die grundlegenden Rechtsbegriffe des Zivilrechts (Vertragsabschluss, AGB, Urheberrecht) benennen, erläutern und gegenüberstellen - Datenschutzrecht auf vertiefter Ebene skizzieren, analysieren und deuten - Juristische Fallgestaltungen strukturiert lösen
Inhalte des Moduls	<p>Vorlesung Recht und Datenschutz</p> <p>Übung Recht und Datenschutz</p>
Lehrformen des Moduls	Vorlesung und Übungen
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	Sommersemester
Modulkoordination	Prof. Dr. Ingo Ritter

Unitbeschreibung zum Modul 10: Vorlesung Recht und Datenschutz

Name der Unit	Vorlesung Recht und Datenschutz
Code	
Name des Moduls	Recht und Datenschutz
Inhalte der Unit	<p>Das Modul hat die folgenden beiden Zielrichtungen:</p> <ul style="list-style-type: none"> • Zum einen werden rechtliche Grundlagen vermittelt, die für die praktischen Abläufe bei der Gestaltung und Durchführung von Verträgen in der Informationsverarbeitung wesentlich sind. Hierbei sind neben Fragen des Vertragsabschlusses, der Leistungserbringung und der Gewährleistungs-/ Haftungsansprüche ebenso Problemstellungen der Allgemeinen Geschäftsbedingungen sowie Querverbindungen zum Urheberrecht relevant. • Zum anderen stehen Grundbegriffe des Datenschutzrechts im Fokus, da bei der Verarbeitung personenbezogener Daten gleichermaßen die Rechte der Betroffenen gewahrt sein müssen. Lernziele sind der Erwerb von Grundkenntnissen des Datenschutzrechts unter Betrachtung der europäischen und nationalen Gesetzgebung. Die erworbenen Kenntnisse befähigen die Studierenden Datenschutzrecht in der Informationsverarbeitung unter Einbeziehung der Schnittstellen zur IT-Sicherheit zu berücksichtigen. <p>Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden.</p> <ul style="list-style-type: none"> • Grundlagen Recht <ol style="list-style-type: none"> 1. Vertragsgestaltung 2. Allgemeine Geschäftsbedingungen 3. Gewährleistungs- und Haftungsansprüche 4. Schnittstellen zum Urheberrecht • Grundlagen Datenschutz <ol style="list-style-type: none"> 1. Begriffe und Grundsätze des Datenschutzes 2. Rechte der Betroffenen 3. Pflichten der Verantwortlichen 4. Datenschutz im internationalen Bereich <p>Schnittstelle IT-Sicherheit</p> <ol style="list-style-type: none"> 1. Technische und organisatorische Maßnahmen 2. Risikobasierter Ansatz
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	70
Anteil der Präsenzzeit	30

Anteil Prüfungszeit inkl. Vorbereitung	10
Anteil Selbststudium	30
Anteil Praxiszeit	0
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Ingo Ritter
Basis – Literatur	<ul style="list-style-type: none"> • Albrecht/Jotzo: Das neue Datenschutzrecht der EU, Nomos, 2017 • Brox/Walker: Allgemeiner Teil des BGB, 42. Auflage, Verlag Franz Vahlen, 2018 • Wedde: EU-Datenschutz-Grundverordnung, Bund-Verlag, Frankfurt, 2016 <p>Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine
Hinweise	

Unitbeschreibung zum Modul 10: Übung Recht und Datenschutz

Name der Unit	Übung Recht und Datenschutz
Code	
Name des Moduls	Recht und Datenschutz
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen. Die Studierenden lernen die juristischen Fragestellungen zu verstehen und selbstständig zu lösen. Die Studierenden bekommen durch die Übungen kontinuierlich ein qualifiziertes Feedback, das ihren Lernprozess gezielt unterstützt.
Lehrformen	Übung
SWS der Unit	2
Workload (h)	80
Anteil der Präsenzzeit	30
Anteil Prüfungszeit inkl. Vorbereitung	0
Anteil Selbststudium	50
Anteil Praxiszeit	0
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Ingo Ritter
Basis – Literatur	Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben
Art und Form des Leistungsnachweises der Unit	
Bewertung des Leistungsnachweises der Unit	
Hinweise	

Modulbeschreibung zum Modul 11: Software Engineering - Analysis

Module title	Software Engineering - Analysis
Module number	M11
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Informatik (B.Sc.)
Module duration	1 semester
Recommended semester	3. semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Einführung in die Programmierung mit C; Englisch; Objektorientierte Programmierung Grundlagen und Konzepte -- C++
Module prerequisites	none
Module examination requirements	None
Module examination	Computer-based project with documentation (submission period 9 weeks, processing time 60 hours) Rating: pass/fail
Learning outcomes and skills	Upon completion of the module the student is able to <ul style="list-style-type: none"> - identify and explain different models of the software process and to analyse software requirements - outline and apply agile project management methods - assess the applicability of software engineering methods in an application development context - explain the roles of software developers and project managers - demonstrate basic proficiency in the software engineering of large software systems and object-oriented software analysis - cooperate and communicate in project-based teamwork
Module contents	Lectures Software Engineering – Analysis Exercises Software Engineering - Analysis
Module teaching methods	Lecture and exercises
Module language	English
Module availability	winter semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	N/A

Unitbeschreibung zum Modul 11: Lecture Software Engineering – Analysis

Unit title	Lecture Software Engineering – Analysis
Code	
Module title	Software Engineering – Analysis (11)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • the software product • different software process models • object-oriented software analysis concepts • different UML diagrams • requirements engineering <p>planning and scheduling, and metrics</p>
Teaching methods	Interactive Lecture
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 practical training (h)	30 h
Unit language	English
Lecturer	Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Recommended reading	<ul style="list-style-type: none"> • Software Engineering: A Practitioner's Approach; Roger S. Pressman and Bruce Maxim, 8th Edition, 2014 • Mastering the Requirements Process: Getting Requirements Right; Suzanne Robertson and James Robertson, 3rd Edition, 2012 <p>Further recommended reading will be announced in the first lectures</p>
Assessment type and form	None
Assessment grading	None
Comments	

Unitbeschreibung zum Modul 11: Exercise Software Engineering – Analysis

Unit title	Exercise Software Engineering – Analysis
Code	
Module title	Software Engineering – Analysis (11)
Unit contents	Exercises relating to topics of lectures (see unit lectures)
Teaching methods	Working in small groups
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 practical training (h)	50 h
Unit language	English
Lecturer	Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Recommended reading	see Lecture Software Engineering – Analysis
Assessment type and form	None
Assessment grading	None
Comments	

Modulbeschreibung zum Modul 12: Embedded Systems

Module title	Embedded Systems
Module number	M12
Module code	
Study program	Informatik – Mobile Anwendungen
Applicability of the module	
Module duration	1 semester
Recommended semester in program	3. semester
Type of module	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	none
Requirements for participation in the module	module 8: Microcomputertechnik
Requirements for participation in the module examination	written laboratory report for every laboratory task (80h)
Module examination	Written examination 90 minutes
Learning outcomes and skills	On successful completion of the module the student understands the architecture of Embedded Systems. They know how to design hardware with micro-controllers and how to program in a problem-oriented language by using hardware functions such as: acquisition and processing of analog and digital data, reaction on interrupt- and timer-generated events, communication via serial interfaces. Additionally the students understand Real time Scheduling. They have the ability to model and implement embedded systems and communication with external devices. Extracurricular skills: project-work, structured problem solving, English language
Module contents	Lecture Embedded Systems Laboratory Embedded Systems
Forms of teaching	Lecture, laboratory
Language	English
Availability of module	annually
Module coordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul 12: Lecture Embedded Systems

Name of unit	Lecture Embedded Systems
Code	
Name of module	Embedded Systems
Unit contents	Architecture of Embedded Systems; assessment of microcontrolles and peripheral components (e.g. key boards, LC Displays, data I/O) with respect to the hardware design of Embedded Intelligent Systems; Special Function Register for programming analog-/ digital converters, timer, serial interfaces and pulsewidth modulation; programming of embedded intelligent systems in a problemoriented language; intelligent algorithms such as pattern recognition and their application for Embedded Intelligent Systems such as intelligent sensors and robots.
Teaching forms	lecture
Semester periods (hours) per week	2
Workload (h)	70 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	10 h
Proportion of private study	30 h
Proportion of practice	
Unit language	English
Tutor/s	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literature	Nauth: Embedded Intelligent Systems, Oldenbourg Verlag, 2005 Schiffmann/Schmitz: Technische Informatik Bd. 1 und 2, Berlin, Springer 2001 Berns/Schürmann/Trapp: Eingebettete Systeme, Vieweg+Teubner, 2010 Current literature will be announced at the beginning of the semester
Type and form of the unit's proof of achievement	None
Assessment of unit achievement	None

Unitbeschreibung zum Modul 12: Labor Embedded Systems

Name of unit	Labor Embedded Systems
Code	
Name of module	Embedded Systems
Unit contents	Projects regarding acquisition, processing and display of analog signals with embedded systems, Programming of timers and actor control.
Teaching forms	Several projects
Semester periods (hours) per week	2
Workload (h)	70 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	0
Proportion of private study	40 h
Proportion of practice	
Unit language	English
Tutor/s	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel
Basis – Literature	See Unit Lecture Embedded Systems
Type and form of the unit's proof of achievement	written laboratory report for every laboratory task (80h)
Assessment of unit achievement	undifferentiated

Modulbeschreibung zum Modul 13: Object-oriented Programming in Java – Advanced Course

Module title	Object-Oriented Programming in Java - Advanced Course
Module number	M13
Module code	
Study program	Informatik (B.Sc.)
Applicability of the module	Applicable to other Computer Science Bachelor programmes
Module duration	One semester
Recommended semester in program	3. Semester
Type of module	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Successful participation in module M4 "Einführung in die Programmierung mit C" and "Objektorientierte Programmierung Grundlagen"
Requirements for participation in the module	Partial examination of "Einführung in die Programmierung mit C" of module 4: Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen
Requirements for participation in the module examination	Partial examination of "Einführung in die Programmierung mit C" of module 4: Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen Laboratory Exercises with documentation Total workload time 80 hours
Module examination	Computer-based project with documentation (submission period 9 weeks) with presentation (min. 15 and max. 30 minutes)
Learning outcomes and skills	Upon completion of the module the student is able to <ul style="list-style-type: none"> - demonstrate sound knowledge of advanced concepts in object-oriented programming - apply advanced techniques of state-of-the-art object-oriented programming paradigms in order to produce an application program of moderate complexity - structure and generate technical texts in English - organise himself/herself as member of a team in a project context - plan and realise an application within a given timeframe
Module contents	Lecture Object-Oriented Programming in Java Exercise Object-Oriented Programming in Java,
Forms of teaching	Lectures Exercises

Language	English
Availability of module	Winter term
Module coordination	Prof. Dr. Ute Bauer-Wersing

Unitbeschreibung zum Modul 13: Lecture OOP with Java

Name of unit	Lecture Object-Oriented Programming in Java
Code	
Name of module	Object-Oriented Programming in Java - Advanced Course (13)
Unit contents	<ul style="list-style-type: none"> • Basic and advanced concepts of object-oriented programming based on the Java programming language • Introduction to the Java Development Kit and Java Application Programming Interface • Platform-independent specification <p>Design and implementation of a Java application which comprises a graphical user interface and is based on API packages</p>
Teaching forms	Lecture
Semester periods (hours) per week	2 h
Workload (h)	70 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	10 h
Proportion of private study	30 h
Proportion of practice	0 h
Unit language	English
Tutor/s	Prof. Dr. Ute Bauer-Wersing , Dr. Maike Kamlage, Prof. Dr. Doina Logofatu
Basis – Literature	<ul style="list-style-type: none"> • Gosling, James et al. The Java Language Specification. (see https://docs.oracle.com/javase/specs/jls/se10/jls10.pdf) • C. Horstmann. Core Java Volume I---Fundamentals. <i>Prentice Hall, Boston, 10 edition, (2016)</i> • Oracle Corp.(Hrsg.). JDK 10 Documentation. https://docs.oracle.com/javase/10/ <p>Additional literature may be announced at the beginning of the course</p>
Type and form of the unit's proof of achievement	none
Assessment of unit achievement	none
Comments	

Unitbeschreibung zum Modul 13: Exercise OOP with Java

Name of unit	Exercise Object-Oriented Programming in Java
Code	
Name of module	Object-Oriented Programming in Java - Advanced Course (13)
Unit contents	Practice and programming examples covering all relevant lecture topics. Exercises support the students in gaining in depth knowledge and hands-on expertise on how to solve programming tasks using appropriate object-oriented techniques. Students will receive continuous feedback to expand their learning achievements.
Teaching forms	Exercise
Semester periods (hours) per week	2 h
Workload (h)	80 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	0 h
Proportion of private study	50 h
Proportion of practice	0 h
Unit language	English
Tutor/s	Prof. Dr. Ute Bauer-Wersing , Dr. Maike Kamlage, Prof. Dr. Doina Logofatu
Basis – Literature	cf. Unit Lecture Object-oriented Programming in Java
Type and form of the unit's proof of achievement	Laboratory Exercises with documentation (total workload 80 hours)
Assessment of unit achievement	
Comments	Self-study time is equivalent to exam preparation time since the examination is via project work

Modulbeschreibung zum Modul 14: Databases

Module title	Databases
Module number	M14
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable as well to other Computer Science bachelor programmes
Module duration	1 semester
Recommended semester	3rd semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	The module builds on the content of the mathematical foundation modules, the programming modules as well as on the module software engineering analysis, which is taught in parallel.
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination (120 minutes)
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - Outline the relational data model and apply it practically - Master the standard database language SQL by using a specific database management system - The students solve a given problem in a structured manner and have to develop their creative skills.
Module contents	Lecture Databases Exercises Databases
Module teaching methods	Lecture Databases, Exercises Databases
Module language	English
Module availability	Winter term
Module coordination	Prof. Dr. Christian Rich
Comments	N/A

Unitbeschreibung zum Modul 14: Lecture Databases

Unit title	Databases Lecture
Code	
Module title	Databases (14)
Unit contents	<ul style="list-style-type: none"> • Introduction: database system concepts, database architecture, data models • The relational data model, relational database constraints, relational algebra • Data modeling and relational database schema: Entity-Relationship model, ER-to-relational mapping, normalization • The relational data definition and manipulation language SQL (restricting and sorting data, single-row and aggregate functions, different join types, subqueries, set operations, working with null values, manipulating data, using DDL statements to create and manage tables). • Introduction to SQL Programming techniques: database functions, procedures, trigger, interfaces <p>Transaction processing concepts, user administration, data dictionary views</p>
Teaching methods	Lecture
Semester periods (hours) per week	4 h
Workload (h)	100 h
Class hours	60 h
Total time of examination incl. preparation	10 h
Time of individual study (h)	30 h
Time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	<ul style="list-style-type: none"> • Elmasri, R. and S. Navathe. Fundamentals of Database Systems, Addison Wesley Publishing Company • Date, C.J., An Introduction to Database Systems. AddisonWesley • Garcia-Molina, H., J. D. Ullman and J. D. Widom, Database Systems: The Complete Book, Prentice Hall. • Härder, Theo; Rahm, Erhard: Datenbanksysteme - Konzepte und Techniken der Implementierung, Springer • Kemper, A. und A. Eickler. Datenbanksysteme, Oldenbourg. • Kifer, M., A. Bernstein and P.M. Lewis. Database Systems: An Application-Oriented Approach. Pearson International

	<ul style="list-style-type: none">• Ramakrishnan, R. and J. Gehrke, Database Management Systems, McGraw-Hill Silberschatz, A., H.F. Korth, S. Sundershan, Database System Concepts, McGraw Hill.
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Unitbeschreibung zum Modul 14: Database Exercise

Unit title	Database Exercise
Code	
Module title	Databases (14)
Unit contents	The focus is on relational databases. Students will apply the knowledge gained in the lecture and practice data modelling and relational database design. They will implement relational databases using SQL DDL statements and practice data retrieval and manipulation using SQL in an interactive manner using a professional database management system. By doing the exercises, students will receive continuously qualified feedback, which will support the learning process.
Teaching methods	Exercises
Semester periods (hours) per week	2 h
Workload (h)	50 h
Class hours	30 h
Total time of examination incl. preparation (h)	0 h
Total time of individual study (h)	20 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	see unit databases lecture
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Modulbeschreibung zum Modul 15: Computer Networks

Module title	Computer Networks
Module number	M15
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable to other Computer Science Bachelor programmes
Module duration	1 semester
Recommended semester	3. Semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	10 cp / 300 h
Recommended previous knowledge	Successful completion of modules of semesters 1-2 particularly Theoretische Informatik, Algorithmen und Datenstrukturen, Rechnerarchitektur, Einführung in die Programmierung mit C.
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination 90 minutes
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - Describe and explain fundamental concepts of computer systems and their interconnection via computer networks - Outline basic concepts of communication protocols and their use in computer networks - produce technical texts in English - The students solve a given problem in a structured manner and have to develop their creative skills.
Module contents	Lecture Computer Networks Exercises Computer Networks
Module teaching methods	Lecture, Exercise
Module language	English
Module availability	Winter term
Module coordination	Prof. Dr. Christian Baun
Comments	The students also learn project- and teamwork as they work in teams

Unitbeschreibung zum Modul 15: Lecture Computer Networks

Unit title	Computer Networks Lecture
Code	
Module title	Computer Networks (15)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Introduction to Computer Networks • Data transmission • OSI - Reference Model • Local Networks • LAN - Extensions • Internet • Network management • Routing, Bridging, Switching • Protocols at all layers of the reference model • IEEE 802 family protocols <p>ARP, IPv4, IPv6, TCP, HTTP, DNS, etc.</p>
Teaching methods	Lecture
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	<ul style="list-style-type: none"> • Christian Baun, Computernetze kompakt, Springer Vieweg, 2018. • Andrew S. Tanenbaum and David J. Wetherall, Computer Networks, Pearson Education 2013. • James F. Kurose and Keith D. Ross, Computer Networking: A Top-Down Approach, Pearson Education, 2016. <p>Further recommended reading will be announced in the first lectures</p>
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Unitbeschreibung zum Modul 15: Exercise Computer Networks

Unit title	Computer Networks Exercise
Code	
Module title	Computer Networks (15)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Introduction to Computer Networks • Data transmission • OSI - Reference Model • Local Networks • LAN - Extensions • Internet • Network management • Routing, Bridging, Switching • Protocols at all layers of the reference model • IEEE 802 family protocols <p>ARP, IPv4, IPv6, TCP, HTTP, DNS, etc.</p>
Teaching methods	Exercise
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	<ul style="list-style-type: none"> • Christian Baun, Computernetze kompakt, Springer Vieweg, 2018. • Andrew S. Tanenbaum and David J. Wetherall, Computer Networks, Pearson Education 2013. • James F. Kurose and Keith D. Ross, Computer Networking: A Top-Down Approach, Pearson Education, 2016. <p>Further recommended reading will be announced in the first lectures.</p>
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Modulbeschreibung zum Modul 16: Operating Systems

Module title	Operating Systems
Module number	M16
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable to other Computer Science Bachelor programmes
Module duration	1 semester
Recommended semester	4th semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Successful completion of modules of semesters 1-2, particularly Theoretische Informatik, Algorithmen und Datenstrukturen, Rechnerarchitektur, Einführung in die Programmierung mit C
Module prerequisites	Passed examination module 9: Algorithmen und Datenstrukturen
Module examination requirements	Passed examination module 9: Algorithmen und Datenstrukturen
Module examination	Written Examination (90 Minutes)
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - Name and describe the fundamental concepts of computer systems, especially the task of operating systems - Outline basic concepts and methods for implementation of operating systems
Module contents	Lecture Operating Systems Exercises Operating Systems
Module teaching methods	Lecture, Exercise
Module language	English
Module availability	winter term
Module coordination	Prof. Dr. Christian Baun
Comments	The students also learn project- and teamwork as they work in teams in the lab. The students solve a given problem in a structured manner and have to develop their creative skills.

Unitbeschreibung zum Modul 16: Lecture Operating Systems

Unit title	Operating Systems Lecture
Code	
Module title	Operating Systems (16)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Processes and process management • Memory management • File systems • Input/Output devices • Distributed operating systems • Windows and Unix based operating systems <p>System management and administration</p>
Teaching methods	Lecture
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	<ul style="list-style-type: none"> • Christian Baun, Betriebssysteme kompakt, Springer Vieweg, 2017. • William Stallings, Operating Systems: Internals and Design Principles, Pearson, 2017. • Andrew Tanenbaum, Modern Operating Systems, Pearson, 2014. <p>Further recommended reading will be announced in the first lectures.</p>
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Unitbeschreibung zum Modul 16: Exercise Operating Systems

Unit title	Operating Systems Exercise
Code	
Module title	Operating Systems (16)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Processes and process management • Memory management • File systems • Input/Output devices • Distributed operating systems • Windows and Unix based operating systems <p>System management and administration</p>
Teaching methods	Exercise
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Sergej Alekseev, Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	<ul style="list-style-type: none"> • Christian Baun, Betriebssysteme kompakt, Springer Vieweg, 2017. • William Stallings, Operating Systems: Internals and Design Principles, Pearson, 2017. • Andrew Tanenbaum, Modern Operating Systems, Pearson, 2014. <p>Further recommended reading will be announced in the first lectures.</p>
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Modulbeschreibung zum Modul 17: Software Engineering - Design

Module title	Software Engineering - Design
Module number	M17
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Informatik (B.Sc.)
Module duration	1 semester
Recommended semester	4. semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	None
Module prerequisites	None
Module examination requirements	Computer-based exercises with written documentation, (processing time 36 hours)
Module examination	Written examination (120 minutes)
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - outline and reproduce the basic principles and concepts of software design and implementation - critically assess and estimate the usage of the various methods of software design in the application development context - classify and illustrate the roles of software developers and project managers - demonstrate enhanced proficiency in the software engineering of large software systems - employ methods of project management - use IDE and CASE tools
Module contents	Lecture Software Engineering – Design Exercise Software Engineering – Design
Module teaching methods	Lectures, Exercises
Module language	Englisch
Module availability	Summer term
Module coordination	Prof. Dr. Eicke Godehardt
Comments	

Unitbeschreibung zum Modul 17: Lecture Software Engineering – Design

Unit title	Lecture Software Engineering – Design
Code	
Module title	Software Engineering – Design (17)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • the software design process • object-oriented design principles and concepts • software architecture • software testing • design patterns <p>different UML diagrams</p>
Teaching methods	Interactive Lecture
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. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Recommended reading	<ul style="list-style-type: none"> • Software Engineering: A Practitioner's Approach; Roger S. Pressman and Bruce Maxim, 8th Edition, 2014 • Mastering the Requirements Process: Getting Requirements Right; Suzanne Robertson and James Robertson, 3rd Edition, 2012 <p>Further recommended reading will be announced in the first lectures</p>
Assessment type and form	n/a
Assessment grading	n/a
Comments	

Unitbeschreibung zum Modul 17: Exercise Software Engineering – Design

Unit title	Exercise Software Engineering – Design
Code	
Module title	Software Engineering – Design (17)
Unit contents	Students gain experience and deeper understand of lecture's topics, and working in teams to solve specific exercises.
Teaching methods	Working in small groups
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)	0 h
Unit language	English
Lecturer	Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Recommended reading	see Lecture Software Engineering – Design
Assessment type and form	Computer-based exercises with written documentation, (processing time 36 hours)
Assessment grading	passed/failed
Comments	

Modulbeschreibung zum Modul 18: Mobile Devices

Module title	Mobile Devices
Module number	M18
Module code	
Study program	Informatik – Mobile Anwendungen
Applicability of the module	Applicable to other Computer Science Bachelor programmes
Module duration	1 semester
Recommended semester in program	4. semester
Type of module	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	None
Requirements for participation in the module	None
Requirements for participation in the module examination	None
Module examination	Written project report (8 weeks) and oral presentation (min. 15min, max. 20min)
Learning outcomes and skills	<p>The students are able to generate programs for mobile computer systems (e.g. smartphones), which require the involvement of additional requirements.</p> <p>They are able to judge time variant behaviour of mobile systems and, therefore design real-time scheduling and integrate sensors and actors. On successful completion of the module the students can design mobile systems and implement them as executable programs. This includes the utilization of cross-building toolchains and development kits e.g. Android Development Kit.</p> <p>The students acquire skills in team work, negotiation, presentation, assertiveness and scientific work.</p>
Module contents	Lecture Mobile Devices Laboratory Mobile Devices
Forms of teaching	Lecture and Laboratory
Language	English
Availability of module	annually
Module coordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul 18: Lecture Mobile Devices

Name of unit	Lecture Mobile Devices
Code	
Name of module	Mobile Devices
Unit contents	<p>Introduction into the programming of mobile computer systems (e.g. smartphones) exploiting the involvement of additional requirements.</p> <ul style="list-style-type: none"> • time variant behaviour • realtime scheduling • integration of sensors and actors • modelling of mobile systems • implementation of mobile applications as executable programs. • utilization of cross-building toolchains and development kits e.g. Android Development Kit. <p>The students acquire skills in team work, negotiation, presentation, assertiveness and scientific work.</p>
Teaching forms	Lecture
Semester periods (hours) per week	2
Workload (h)	50 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	
Proportion of private study	20 h
Proportion of practice	
Unit language	English
Tutor/s	Prof. Dr. Matthias Deegener
Basis – Literature	<ul style="list-style-type: none"> • Stuart Bennett: Real-Time Computer Control, Prentice Hall, 1994 • Liu, Jane W. S.: Real-time systems. Prentice Hall, 2000 • Schiffmann/Schmitz: Technische Informatik Bd. 1 und 2, Berlin, Springer 2001 • Berns/Schürmann/Trapp: Eingebettete Systeme, Vieweg+Teubner, 2010 • Steele/To: The Android Developer's Cookbook: Building Applications with the Android SDK <p>Current reference will be published at the beginning of the lectures</p>
Type and form of the unit's proof of achievement	None
Assessment of unit achievement	None

Unitbeschreibung zum Modul 18: Laboratory Mobile Devices

Name of unit	Laboratory Mobile Devices
Code	
Name of module	Mobile Devices
Unit contents	<ul style="list-style-type: none"> • Programming of mobile devices • Debugging of microcontrollers • implementation of mobile applications as executable programs. • utilization of cross-building toolchains and development kits e.g. Android Development Kit.
Teaching forms	Laboratory
Semester periods (hours) per week	2
Workload (h)	100 h
Proportion of attendance time	30 h
Proportion of examination time including preparation	50 h
Proportion of private study	20 h
Proportion of practice	
Unit language	English
Tutor/s	Prof. Dr. Jens Liebehenschel, Prof. Dr. Thomas Gabel, Prof. Dr. Matthias Deegener
Basis – Literature	See unit lecture
Type and form of the unit's proof of achievement	None
Assessment of unit achievement	None

Modulbeschreibung zum Modul 19: IT-Security

Module title	IT-Security
Module number	M19
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable to other Computer Science Bachelor programmes
Module duration	1 semester
Recommended semester	4th semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Successful completion of all modules of semesters 1- 3
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination (90 minutes)
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - Name and describe fundamental concepts of IT Security - detect and interpret IT Security aims and risks and analyze security mechanisms and their applicability with respect to exemplary scenarios - develop basic solutions, concepts and methods to implement IT Security and assess security risks in simple scenarios.
Module contents	IT Security Lecture, IT Security Exercise
Module teaching methods	Lecture, Exercise
Module language	Englisch
Module availability	Summer term
Module coordination	Prof. Dr. Martin Kappes
Comments	

Unitbeschreibung zum Modul 19: Lecture IT Security

Unit title	IT Security Lecture
Code	
Module title	IT Security (19)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Cryptographical principles and methods • Authentication • Operating system security • Application security • Malware • Network security • Firewalls • Virtual Private Networks • Network surveillance • Availability • Network applications • Security of realtime communications • Local network security • Standards • Practical implications
Teaching methods	Lecture
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. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	<ul style="list-style-type: none"> • Martin Kappes, Netzwerk- und Datensicherheit, Springer, Wiesbaden, 2. Auflage, 2013. • Claudia Eckert, IT-Sicherheit: Konzepte, Verfahren, Protokolle, Oldenbourg-Verlag, München, 10. Auflage, 2018. • Original NIST, IETF, IEEE and ITU Standards. • Recommended reading will be announced in the first lectures.
Assessment type and form of	none
Assessment grading	none

Comments	
----------	--

Unitbeschreibung zum Modul 19: Exercise IT Security

Unit title	IT Security Exercise
Code	
Module title	IT Security (19)
Unit contents	<p>Selection from areas such as, but not limited to:</p> <ul style="list-style-type: none"> • Cryptographical principles and methods • Authentication • Operating system security • Application security • Malware • Network security • Firewalls • Virtual Private Networks • Network surveillance • Availability • Network applications • Security of real-time communications • Local Network security • Standards <p>Practical implications</p>
Teaching methods	Exercise
Semester periods (hours) per week	2 h
Workload (h)	80 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	50 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	see Unit "IT Security Lecture"
Assessment type and form of	none
Assessment grading	none
Comments	

Modulbeschreibung zum Modul 20: Distributed Systems

Module title	Distributed Systems
Module number	M20
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable to other Computer Science Bachelor programmes
Module duration	ein Semester
Recommended semester	4. Semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Besides the module "Object-Oriented Programming in Java - Advanced Course" knowledge from the modules "Databases" and "Software Engineering – Analysis" is advisable.
Module prerequisites	Module 13 "Object-oriented Programming in Java – Advanced Course"
Module examination requirements	Laboratory Exercises (total time 30 hours)
Module examination	Written examination (90 minutes)
Learning outcomes and skills	<p>Upon completion of the module the student is able to</p> <ul style="list-style-type: none"> - Realize distributed applications by using different technologies common in the industry - Implement practical examples in programming - Install and configurate software components - Use frameworks and middleware tools in own programs - assess different technologies and decide upon their bene-fits in concrete application contexts in order to being able to design suitable applications themselves - solve problems by developing distributed applications on the basis of a sound theoretical foundation.
Module contents	Distributed Systems Exercise Distributed Systems Lecture
Module teaching methods	Vorlesung und Übungen
Module language	English
Module availability	Summer term
Module coordination	Prof. Dr. Justus Klingemann
Comments	

Unitbeschreibung zum Modul 20: Lecture Distributed Systems

Unit title	Distributed Systems Lecture
Code	
Module title	Distributed Systems (20)
Unit contents	<p>Besides a discussion of properties of and challenges for distributed systems different implementation technologies for the development of modern applications are introduced. In the following, different possible topics for the unit content are listed. The different subjects can be covered with variable degree of depth depending on the emphasis of the lecturer:</p> <ul style="list-style-type: none"> • Sockets as a base technology for distributed applications • Object-oriented Middleware-Technologies • Message Queues • Web-based distributed systems • Web Services • REST • Techniques for integrating databases <p>If suitable, additional relevant topics shall be covered.</p>
Teaching methods	Lecture
Semester periods (hours) per week	4 h
Workload (h)	100 h
Class hours	60 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. Christian Baun, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Karsten Weronek
Recommended reading	<ol style="list-style-type: none"> 1. Günther Bengel: Grundkurs Verteilte Systeme, Springer Vieweg 2. Dietmar Abts: Masterkurs Client/Server-Programmierung mit Java, Springer Vieweg 3. Rainer Oechsle: Parallele und verteilte Anwendungen in Java, Hanser 4. Peter Mandl: Masterkurs Verteilte betriebliche Informationssysteme, Vieweg + Teubner 5. Frank Müller-Hofmann, Martin Hiller, Gerhard Wanner: Programmierung von verteilten Systemen und Webanwendungen mit Java EE, Springer Vieweg 6. Steffen Heinzl, Markus Mathes: Middleware in Java, Vieweg 7. G. Coulouris, J. Dollimore, T. Kindberg: Distributed Systems: Concepts and Design, Addison-Wesley

	8. A. Tanenbaum, M. van Steen: Distributed Systems: Principles and Paradigms, Prentice-Hall / Pearson Additional literature may be announced at the beginning of the course.
Assessment type and form of	none
Assessment grading	none
Comments	

Unitbeschreibung zum Modul 20: Exercise Distributed Systems

Unit title	Distributed Systems Exercise
Code	
Module title	Distributed Systems (20)
Unit contents	<ul style="list-style-type: none"> • Programming exercises based on technologies taught in lecture • Installation and configuration of software components • Creation of distributed applications using technologies taught in lecture • Individual feedback to students
Teaching methods	Exercises
Semester periods (hours) per week	2 h
Workload (h)	50 h
Class hours	30 h
Total time of examination incl. preparation (h)	0 h
Total time of individual study (h)	20 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Karsten Weronek
Recommended reading	see unit Distributed Systems Lecture
Assessment type and form of	Laboratory Exercises (total time 30 hours)
Assessment grading	passed/failed
Comments	

Modulbeschreibung zum Modul 21: Practical Computer Networks and Applications

Module title	Practical Computer Networks and Applications
Module number	21
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Informatik (B.Sc.)
Module duration	1 semester
Recommended semester	4. semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Successful completion of modules of semesters 1-3, particularly Computer Networks
Module prerequisites	None
Module examination requirements	None
Module examination	Computer-based project with documentation (submission period 9 weeks, processing time 60 hours) Rating: Pass/fail
Learning outcomes and skills	Upon completion of the module the student is able to <ul style="list-style-type: none"> - set up computer networks and computer network applications - apply network surveillance technologies for debugging, performance analysis and problem mitigation - distinguish and outline recent communication paradigms, such as, but not limited to, , e.g. Peer-to-Peer, Cloud Computing, Edge Computing, Fog Computing.
Module contents	Practical Computer Networks and Applications Lab Practical Computer Networks and Applications Lecture
Module teaching methods	Lecture, Laboratory Exercise
Module language	English
Module availability	Summer term
Module coordination	Prof. Dr. Christian Baun
Comments	

Unitbeschreibung zum Modul 21: Practical Computer Networks and Applications Lecture

Unit title	Practical Computer Networks and Applications Lecture
Code	
Module title	Practical Computer Networks and Applications (21)
Unit contents	<ul style="list-style-type: none"> • recent communication paradigms, such as, but not limited to, e.g. Peer-to-Peer, Cloud Computing, Edge Computing, Fog Computing. • current network applications and their foundations, such as, but not limited to, e.g. realtime communication, file sharing, content distribution networks
Teaching methods	Lecture
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	
Assessment type and form of	none
Assessment grading	none
Comments	

Unitbeschreibung zum Modul 21: Practical Computer Networks and Applications Lab

Unit title	Practical Computer Networks and Applications Lab
Code	
Module title	Practical Computer Networks and Applications (21)
Unit contents	<ul style="list-style-type: none"> • setting up of computer networks and computer network applications. • application of network surveillance technologies for debugging, performance analysis and problem mitigation.
Teaching methods	Lab
Semester periods (hours) per week	2 h
Workload (h)	75 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	35 h
Total time of practical training (h)	0 h
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
Recommended reading	
Assessment type and form of	None
Assessment grading	None
Comments	

Modulbeschreibung zum Modul 22: Mobile Application Exercises

Module title	Mobile Application Exercises
Module number	22
Module code	
Study program	Informatik – Mobile Anwendungen (B.Sc.)
Module usability	Applicable in other computer science bachelor courses
Module duration	1 semester
Recommended semester	4 th semester
Module type	Compulsory module
ECTS points (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	none
Module prerequisites	Passed (partial) examination „Einführung in die Programmierung mit C“ (M4)
Module examination requirements	None
Module examination	Written project report (8 weeks) and oral presentation (min. 15min, max. 20min)
Learning outcomes and skills	<p>On successful completion of the module the student understands the architecture of Embedded Systems. They know how to design hardware with micro-controllers and how to program in a problem-oriented language by using hardware functions such as: acquisition and processing of analog and digital data, reaction on interrupt- and timer-generated events, communication via serial interfaces. Additionally the students understand Real time Scheduling. They have the ability to model and implement embedded systems and communication with external devices.</p> <p>Extracurricular skills: project-work, structured problem solving, English language</p>
Module contents	Mobile Application Exercises
Module teaching methods	Project
Language	English
Module availability	Every semester
Module coordination	N/A

Unitbeschreibung zum Modul 22: Mobile Application Exercises

Name of unit	Mobile Application Exercises
Code	
Name of module	Mobile Application Exercises
Unit contents	The knowledge acquired so far in the areas of programming, software engineering, mobile devices and distributed systems are utilized within a realistic programming environment. This includes, in particular, the complete development and implementation of a mobile application (design, implementation, test, presentation).
Teaching forms	Projekt
Semester periods (hours) per week	4
Workload (h)	150 h
Proportion of attendance time	60 h
Proportion of examination time including preparation	0 h
Proportion of private study	90 h
Proportion of practice	
Unit language	Englisch
Tutor/s	Prof. Dr. Jens Liebehenschel, Prof. Dr. Thomas Gabel, Prof. Dr. Matthias Deegener
Basis – Literature	Readings depend on the actual topic and will be announced in the first lectures.
Type and form of the unit's proof of achievement	None
Assessment of unit achievement	None

Modulbeschreibung zum Modul 23: Serviceorientierte Architekturen

Modultitel	Serviceorientierte Architekturen
Modulnummer	M23
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Inhaltlich werden die Module der objektorientierten Programmierung und des Software Engineerings vorausgesetzt.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Klausur 90 Minuten
Lernergebnisse und Kompetenzen	<p>Die Studierenden verstehen serviceorientierte Architekturen (SOA) als a) Managementkonzept, das eine schnelle Reaktion auf veränderte Anforderungen im Geschäftsumfeld erlaubt, und als b) unternehmensweites IT-Architekturkonzept, das fachliche Dienste und Funktionalitäten in Form von Services modelliert und in eine Ablaufumgebung integriert.</p> <p>Die Studierenden können Services in einfachen Geschäftsprozessen identifizieren, den Informationsfluss zwischen Services in einer Choreographie modellieren, die Services orchestrieren und auf einer geeigneten Infrastruktur (z.B. einem Hub- & Spoke-System) implementieren. Bezüglich der dabei erstellten verteilten Anwendungen verfügen die Studierenden über einen grundlegenden Qualitätsbegriff.</p> <p>Die Studierenden sind damit in der Lage, einfache Geschäftsprozesse von ihrem betriebswirtschaftlichen Rationale bis zu deren Umsetzung ganzheitlich zu begleiten.</p>
Inhalte des Moduls	Vorlesung Übung
Lehrformen des Moduls	Vorlesung und Übung
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	jährlich
Modulkoordination	Prof. Dr. Josef Fink

Unitbeschreibung zum Modul 23: Vorlesung Serviceorientierte Architekturen

Name der Unit	Vorlesung Serviceorientierte Architekturen
Code	
Name des Moduls	Serviceorientierte Architekturen
Inhalte der Unit	<p>Die Studierenden werden zunächst aus einer betrieblich-organisatorischen Perspektive an die Identifikation und Modellierung von Services, die Integration betrieblicher Anwendungssysteme (Legacy Systeme) und die Komposition von Services im Sinne einer Orchestrierung bzw. Choreographie herangeführt.</p> <p>Aus einer technischen Perspektive werden anschließend Komponententechnologien (insbesondere Web Services und diesbez. Grundlagen wie SOAP, WSDL, UDDI) eingeführt und Ausprägungen serviceorientierter Architekturen (im Sinne einer Schnittstellen-, Nachrichten- bzw. Ressourcen-Orientierung) anhand konkreter Softwareprodukte (z.B. MS Biztalk) vorgestellt. Betrachtungen zur Granularität von Services, zu Sicherheit, Qualität und Qualitätssicherung, sowie zu aktuellen Trends runden diesen Teil der Veranstaltung ab.</p> <p>In den Übungen werden die in der Vorlesung vermittelten Inhalte anhand ausgewählter Beispiele vertieft.</p>
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	75 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	15 h
Anteil Selbststudium	30 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Weronek
Basis – Literatur	<p>Bussler, C. B2B Integration. Concepts and Architecture., Springer, Berlin, 2003</p> <p>Fowler, M., Rice, D., und Foemmel, M. Patterns of Enterprise Application Architecture, Addison-Wesley, München, 2002</p> <p>Krafzig, D., Banke, K., und Slama, D. "Enterprise SOA. Service Oriented Architecture Best Practices," Prentice Hall, New Jersey, 2004</p> <p>Starke, G., u. Tilkov, S. SOA-Expertenwissen, dpunkt, Heidelberg, 2007</p>
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 23: Übung Serviceorientierte Architekturen

Name der Unit	Übung Serviceorientierte Architekturen
Code	
Name des Moduls	Serviceorientierte Architekturen
Inhalte der Unit	Ziel der Übungen ist es, die Studierenden anhand eines selbst aufzu-bauenden, kleinen Szenarios für die behandelten Themenkreise zu sensibilisieren.
Lehrformen	Übung
SWS der Unit	1
Workload (h)	75
Anteil der Präsenzzeit	15
Anteil Prüfungszeit inkl. Vorbereitung	0
Anteil Selbststudium	60
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	In Präsenzzeit enthalten.
Basis – Literatur	siehe Unit Vorlesung Serviceorientierte Architekturen
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	keine

Modulbeschreibung zum Modul 24 Human Machine Interfaces

Modultitel	Human Machine Interfaces
Modulnummer	M24
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Inhaltlich werden die Module der objektorientierten Programmierung, des Software Engineerings und Echtzeit-Systeme vorausgesetzt.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Projektarbeit (Bearbeitungszeit: 8 Wochen) mit Präsentation (mindestens 15 Minuten und höchstens 20 Minuten)
Lernergebnisse und Kompetenzen	<p>Die Studierenden können Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion ausarbeiten. Schwerpunkt sind die Benutzerschnittstellen von mobilen Geräten, die Sie präzisieren können.</p> <p>Sie entwickeln eine softwaretechnologische Problemlösungskompetenz für die Spezifikation und Implementierung von Benutzerschnittstellen auf Basis theoretischer Grundlagen.</p>
Inhalte des Moduls	<p>Vorlesung Human Machine Interfaces</p> <p>Übung Human Machine Interfaces</p>
Lehrformen des Moduls	Vorlesung und Übung
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	jährlich
Modulkoordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul 24: Vorlesung Human Machine Interfaces

Name der Unit	Vorlesung Human Machine Interfaces
Code	
Name des Moduls	Distributed Systems
Inhalte der Unit	<p>Die Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen ist besonders für mobile Geräte von entscheidender Bedeutung. Hierfür werden Konzepte, Modelle, Methoden und Techniken zum Entwurf und zur Implementierung vorgestellt.</p> <p>Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden.:</p> <ul style="list-style-type: none"> • Einführung in die Grundlagen der Mensch-Computer-Schnittstellen • Entwurfsprinzipien und Modelle für interaktive Systeme • Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen • Methoden zur Realisierung und Implementierung von Benutzerschnittstellen • Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	100 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	20 h
Anteil Selbststudium	50 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jörg Schäfer
Basis – Literatur	<ul style="list-style-type: none"> • Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin; 2. Auflage. 2010 • Andreas Heinecke, Mensch-Computer-Interaktion, Fachbuchverlag Leipzig; 2004 • Dix,, Finlay, Abowd, Beale, Human-Computer Interaction, Pearson, 2004 • David Benyon, Designing Interactive Systems, Pearson 2010 <p>Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben</p>
Art und Form des Leistungsnachweises der Unit	Keine

Bewertung des Leistungsnachweises der Unit	Kein
--	------

Unitbeschreibung zum Modul 24: Übung Human Machine Interfaces

Name der Unit	Übung Human Machine Interfaces
Code	
Name des Moduls	Human Machine Interfaces
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen
Lehrformen	Übung
SWS der Unit	2
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener, Prof. Dr. Jörg Schäfer
Basis – Literatur	Siehe Unit Vorlesung Human Machine Interfaces (HMI)
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Modulbeschreibung zum Modul 25 : Software-Projekt Mobile und Verteilte Anwendungen

Modultitel	Software-Projekt Mobile und Verteilte Anwendungen
Modulnummer	M25
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	10 cp / 300 h
Empfohlene inhaltliche Vorkenntnisse	Module der objektorientierten Programmierung, des Software Engineerings und Echtzeit-Systeme
Voraussetzungen für die Teilnahme am Modul	<ol style="list-style-type: none"> 1. Beständenes Modul M4 "Einführung in die Programmierung" 2. Beständenes Modul M11 "Software Engineering – Analysis“ oder am Modul M17 „Software Engineering – Design“ 3. Mindestens 80 CP aus den ersten 4 Semestern
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	<p>Projektarbeit (15 Wochen), dies beinhaltet entweder niedergelegten Software-Code oder niedergelegte Software-Dokumentation oder niedergelegte Dokumentation des Projekt-Managements/Fortschritts oder aufgeschriebene Recherche-Ergebnisse, die zum Fortschritt des Projektes beitragen oder weitere schriftliche Dokumente, die zum Fortschritt des Projektes Relevanz aufweisen (z. B. Qualitätssicherungsdokumente) sowie eine Präsentation eigener Ergebnisse (min. 10, max. 20 Minuten) auf mindestens einer der Projektsitzungen und regelmäßige (wöchentliche) Berichterstattung des eigenen Fortschritts (zugewiesene Arbeitspakete) in den Projektbesprechungen mit Diskussionsbeiträgen und Arbeitspaketzuweisung.</p>
Lernergebnisse und Kompetenzen	<p>Durch das Bearbeiten eines komplexen Software-Projekts aus dem Bereich mobile Systeme oder verteilte Anwendungen optimieren die Studierenden ihre der technischen Fähigkeiten in Programmierung, Dokumentation, SW-Engineering, Präsentation und Kommunikation. Sie lösen Aufgaben aus einem oder mehreren Gebieten des Curriculums (z.B. Verteilte Anwendung, Mobile Devices etc.). Außerdem werden außerfachlichen Kompetenzen erworben: Projekterfahrung aufbauen, durch das Erreichen eines Ziels in der vorgegebenen Zeit. Weiterhin sammeln Sie Erfahrung im Team sowie die eigene Zeitschiene zu organisieren.</p> <p>Auf hohem technischen Niveau kommunizieren Sie mit anderen und</p>

	können unerwartete Schwierigkeiten überwinden (sowohl technischer Art als auch sozialer Art). Sie lernen Toleranz gegenüber den Projektpartnern und Verantwortung zu übernehmen.
Inhalte des Moduls	Software-Projekt Mobile und Verteilte Anwendungen
Lehrformen des Moduls	Projektarbeit
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	In jedem Semester
Modulkoordination	Prof. Dr. Andreas Orth

Unitbeschreibung zum Modul 25: Software-Projekt Mobile und Verteilte Anwendungen

Name der Unit	Software-Projekt Mobile und Verteilte Anwendungen
Code	
Name des Moduls	Software-Projekt Mobile und Verteilte Anwendungen
Inhalte der Unit	Ein Thema aus dem Bereich mobile Systeme oder verteilte Anwendungen.
Lehrformen	Projektarbeit
SWS der Unit	8
Workload (h)	300 h
Anteil der Präsenzzeit	120 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	180 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	alle Professoren des Studiengangs Informatik – Mobile Anwendungen
Basis – Literatur	
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Differenziert

Modulbeschreibung zum Modul 26.1 Datenbankadministration

Modultitel	Datenbankadministration
Modulnummer	M26.1
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Wahlpflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Das Modul baut auf den Inhalten des Moduls Datenbanken auf.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	<p>2 Teilprüfungsleistungen (je 50 %):</p> <ul style="list-style-type: none"> • Vortrag (min. 15, max. 30 Minuten) und Durchführung einer praktischen Übung (80 Minuten). • Klausur (90 Minuten)
Lernergebnisse und Kompetenzen	Die Studierenden kennen die Architektur eines DBMS und die vielfältigen Aufgaben der Datenbankadministration. Als konkretes Referenzsystem kennen sie das Oracle DBMS und können für dieses DBMS wichtige Aufgaben der Datenbankadministration übernehmen
Inhalte des Moduls	<p>Vorlesung Datenbankadministration</p> <p>Übung Datenbankadministration</p>
Lehrformen des Moduls	Seminaristischer Unterricht mit praktischen Übungen / Workshops
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	jährlich
Modulkoordination	Prof. Dr. Christian Rich

Unitbeschreibung zum Modul 26.1 Vorlesung Datenbankadministration

Name der Unit	Vorlesung Datenbankadministration
Code	
Name des Moduls	Datenbankadministration
Inhalte der Unit	<p>Aufgaben des Datenbankadministrators</p> <p>Datenbankarchitektur</p> <p>Installation der DBMS Software</p> <p>Erstellen und Konfigurieren von Datenbankinstanzen</p> <p>Speicherstrukturen</p> <p>Security</p> <p>Auditing</p> <p>Netzwerkdienste</p> <p>Undo Data</p> <p>Database Maintenance</p> <p>Performance Management</p> <p>Backup & Recovery</p>
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	50 h
Anteil der Präsenzzeit	35 h
Anteil Prüfungszeit inkl. Vorbereitung	5 h
Anteil Selbststudium	10 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Christian Rich
Basis – Literatur	<ul style="list-style-type: none"> • Study Guide “Oracle Database 11g: Administration Workshop I”, Oracle • OCA/OCP Oracle Database 11g All-in-One Exam Guide, John Watson, Roopesh Ramklass und Bob Bryla von Mcgraw-Hill Professional • Oracle Database Administrator’s Guide 11g, Oracle
Art und Form des Leistungsnachweises der Unit	TPL: Klausur (90 Minuten), Gewichtung 50%
Bewertung des Leistungsnachweises der Unit	differenziert

Unitbeschreibung zum Modul 26.1 Übung Datenbankadministration

Name der Unit	Übung Datenbankadministration
Code	
Name des Moduls	Datenbankadministration
Inhalte der Unit	Aufgaben und Beispiele zu den Vorlesungsthemen.
Lehrformen	Übung / Workshop
SWS der Unit	2
Workload (h)	100 h
Anteil der Präsenzzeit	35 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	50 h
Anteil Praxiszeit	5 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Christian Rich
Basis – Literatur	
Art und Form des Leistungsnachweises der Unit	TPL: Vortrag (min. 15, max. 30 Minuten) und Durchführung einer praktischen Übung (80 Minuten)
Bewertung des Leistungsnachweises der Unit	Differenziert Noten 1 bis 4, 5 = nicht ausreichend

Modulbeschreibung zum Modul 26.2 Ausgewählte Probleme aus dem ACM Programming Contest

Modultitel	Ausgewählte Probleme aus dem ACM Programming Contest
Modulnummer	M26.2
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Wahlpflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Einführung in die Programmierung Objektorientierte Programmierung Vertiefung – Java.
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	2 Teilprüfungsleistungen: <ul style="list-style-type: none"> • Hausarbeit (12 Wochen) (40%) und • mündliche Prüfung (mindestens 15 Minuten höchstens 30 Minuten) (60%)
Lernergebnisse und Kompetenzen	Die Studierenden sind in der Lage Algorithmen in Java/C/ C++ zu implementieren. Sie haben praktische Erfahrungen mit der Implementierung von Programmen gemacht. Weiterhin haben sie gelernt, praktische algorithmischen/mathematischen Methoden, die ein Problem von der Analyse bis zum Programm komplett behandeln, anzuwenden. Sie können die eigene Teamarbeit und Präsentationstechniken, sowie die der anderen Studierenden evaluieren. Die Studierenden können diese Kompetenzen bei der Teilnahme an Programmier Wettbewerben anwenden.
Inhalte des Moduls	Vorlesung Ausgewählte Probleme aus dem ACM Programming Contest Labor Ausgewählte Probleme aus dem ACM Programming Contest
Lehrformen des Moduls	Seminaristischer Unterricht mit Labor
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	In jedem Semester
Modulkoordination	Prof. Dr. Doina Logofătu

Unitbeschreibung zum Modul 26.2 Vorlesung Ausgewählte Probleme aus dem ACM

Programming Contest

Name der Unit	Vorlesung Ausgewählte Probleme aus dem ACM Programming Contest
Code	
Name des Moduls	Ausgewählte Probleme aus dem ACM Programming Contest
Inhalte der Unit	<p>Es werden in der Woche 2 Stunden Referate, Implementierungen und Diskussionen stattfinden.</p> <p>In Referaten stellen die Teilnehmer Problemstellungen aus dem seit 1970 alljährlich stattfindenden ACM Programming Contest vor. Es geht überwiegend um:</p> <ul style="list-style-type: none"> • Mengen, Relationen, Algebra • Algorithmische Zahlentheorie, Codierungstheorie • Algorithmische Geometrie • Kombinatorik • Greedy • Backtracking • Dynamische Programmierung • Graphentheorie • Rekursion, Teile und Herrsche • Zeichenketten-Such-Algorithmen (String-Matching-Algorithms) • Komplexitätstheorie, NP-vollständige Probleme • Datenstrukturen <p>Im praktischen Teil entwerfen und implementieren die Teilnehmer Programme für ausgewählte Probleme. Die Auswahl der Referatsthemen erfolgt in einer Besprechung spätestens zwei Wochen vor dem Vortrag. Die Länge eines Vortrags ist auf maximal 30 Minuten begrenzt. Der Inhalt soll ein Problem klar und mit Hilfe einiger Beispiele erläutern und eventuell verwandte Probleme präsentieren. Kurz soll auch die Realisierung der Lösung/en in einer Programmiersprache gezeigt werden, sowie der Kern der Methode im Pseudocode/Diagramm. Anschließend besprechen wir jeden Vortrag, wobei wir auch gerne Implementierungsdetails analysieren können.</p> <p>Als Beispiele sehen Sie sich [2] Grundlegende Algorithmen mit Java an: Verschachtelte Schachteln S. 25, Die Zahl 4 S. 108, Die Torte S. 115, Collatz-Funktion S. 125, Die Türme von Hanoi S. 146, Orangensport S. 196, Alle Wege des Springers S. 188, Sudoku S. 214, Das Haus des Nikolaus S. 221, Verteilung der Geschenke S. 258, Schotten auf dem Oktoberfest S. 266 oder Arbitrage S. 305.</p>
Lehrformen	Vorlesung
SWS der Unit	2

Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	5 h
Anteil Selbststudium	15 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Christian Rich
Basis – Literatur	<ol style="list-style-type: none"> 1. Doina Logofătu, Algorithmen und Problemlösungen mit C++, Vieweg Verlag, ISBN 978-3-8348-0763-2, 2010. 2. Doina Logofătu, Grundlegende Algorithmen mit Java, Vieweg Verlag, ISBN 978-3-8348-0369-6, 2008. 3. Thomas H. Cormen, Charles Leiserson, Ronald L. Rivest, Clifford Stein: Algorithmen - Eine Einführung, 2. Auflage, Oldenbourg Wissenschaftsverlag, München 2007, ISBN 978-3-486-58262-8. 4. - Thomas H. Cormen, Charles Leiserson, Ronald L. Rivest, Clifford Stein: Introduction to Algorithms. MIT Press, Boston 2001, 2002, 2003, ISBN 0-262-53196-8. (engl. Orig.-Fass.)
Art und Form des Leistungsnachweises der Unit	TPL: mündliche Prüfung (mindestens 15 Minuten höchstens 30 Minuten), Gewichtung 60%
Bewertung des Leistungsnachweises der Unit	Differenziert Noten 1 bis 4, 5 = nicht ausreichend

Unitbeschreibung zum Modul 26.2 Labor Ausgewählte Probleme aus dem ACM

Programming Contest

Name der Unit	Labor Ausgewählte Probleme aus dem ACM Programming Contest
Code	
Name des Moduls	Ausgewählte Probleme aus dem ACM Programming Contest
Inhalte der Unit	Programmierprojekte entsprechend der Vorlesung
Lehrformen	Labor
SWS der Unit	2
Workload (h)	100 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	60 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Doina Logofătu
Basis – Literatur	
Art und Form des Leistungsnachweises der Unit	Hausarbeit (Bearbeitungsdauer 12 Wochen), Gewichtung 40%
Bewertung des Leistungsnachweises der Unit	Differenziert Noten 1 bis 4, 5 = nicht ausreichend

Modulbeschreibung zum Modul 26.3: Fahrzeug zu Fahrzeug Kommunikation

Modultitel	Fahrzeug zu Fahrzeug Kommunikation
Modulnummer	M26.3
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Wahlpflichtmodul
ECTS-Punkte (cp) / Workload (h)	5 cp / 150 h
Empfohlene inhaltliche Vorkenntnisse	Grundlagen der Programmierung
Voraussetzungen für die Teilnahme am Modul	Keine
Voraussetzungen für die Teilnahme an der Modulprüfung	Keine
Modulprüfung	Projektarbeit (Bearbeitungsdauer: 8 Wochen) mit Präsentation (mindestens 15 Minuten und höchstens 20 Minuten)
Lernergebnisse und Kompetenzen	<p>Die Studierenden sind in der Lage Techniken zum Datenaustausch zwischen Fahrzeugen (C2C) und anderen Partnern (C2X) zu verstehen. Sie können Sicherheit und Privatheit in Kommunikationsnetzen analysieren.</p> <p>Die Studierenden wenden ihr Wissen durch die Programmierung von:</p> <ul style="list-style-type: none"> • Algorithmen zur Erhöhung der Verkehrssicherheit oder zur Verbesserung des Verkehrsflusses, • Simulationen von Fahrzeug-Netzwerken oder • Kommunikationsstack, Treiber, Sensoren, etc. an.
Inhalte des Moduls	Vorlesung Fahrzeug zu Fahrzeug Kommunikation Labor Fahrzeug zu Fahrzeugkommunikation
Lehrformen des Moduls	Vorlesung und Labor
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	jährlich
Modulkoordination	Prof. Dr. Matthias Deegener

Unitbeschreibung zum Modul 26.3: Vorlesung Fahrzeug zu Fahrzeug Kommunikation

Name der Unit	Vorlesung Fahrzeug zu Fahrzeug Kommunikation
Code	
Name des Moduls	Fahrzeug zu Fahrzeug Kommunikation
Inhalte der Unit	Car2Car Communication Consortium Netzwerkgrundlagen Adhoc Netzwerke C2C Security C2C Projekte C2C Applikationen C2C Simulation
Lehrformen	Vorlesung
SWS der Unit	2
Workload (h)	50 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	20 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Christian Rich
Basis – Literatur	Car 2 Car Communication Consortium Manifesto: http://car-2-car.org/fileadmin/downloads/C2C-CC_manifesto_v1.1.pdf Rech: Wireless LANs ISBN: 978-3-936931-51-8 Perkins: AD HOC Networking ISBN 0-201-309768 Nett, Mock, Gergeleit: Das drahtlose Ethernet ISBN 3-8273-1741-X
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 26.3: Labor Fahrzeug zu Fahrzeug Kommunikation

Name der Unit	Labor Fahrzeug zu Fahrzeug Kommunikation
Code	
Name des Moduls	Fahrzeug zu Fahrzeug Kommunikation
Inhalte der Unit	Praktische Umsetzung spezieller Themen aus der Vorlesung
Lehrformen	Labor
SWS der Unit	2
Workload (h)	100 h
Anteil der Präsenzzeit	30 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	60 h
Anteil Praxiszeit	
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Matthias Deegener
Basis – Literatur	
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Modulbeschreibung zum Modul 27 Interdisziplinäre Studium Generale

Modultitel	Interdisziplinäre Studium Generale
Modulnummer	M27
	Es gilt die Allgemeine Modulbeschreibung Interdisziplinäres Studium Generale gemäß Anlage 1 zu § 7 Abs. 12 Satz 1 der Allgemeinen Bestimmungen für Prüfungsordnungen mit den Abschlüssen Bachelor und Master an der Frankfurt University of Applied Sciences (AB Bachelor/Master) vom 10. November 2004 (Staatsanzeiger für das Land Hessen 2005 S. 519) in der Fassung der Änderung vom 20. Februar 2019 (veröffentlicht am 13. März 2019 auf der Internetseite in den Amtlichen Mitteilungen der Frankfurt University of Applied Sciences).

Modulbeschreibung zum Modul 28: Praxisphase

Modultitel	Praxisphase
Modulnummer	M28
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	Verwendbar in anderen Informatik Bachelor-Studiengängen
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	6. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	18 cp / 540 h
Empfohlene inhaltliche Vorkenntnisse	Programmierung, Software Engineering und Kenntnisse aus den vertiefenden Vorlesungen der vorangehenden Semester.
Voraussetzungen für die Teilnahme am Modul	120 Creditpoints aus Vorlesungen der ersten 5 Semester.
Voraussetzungen für die Teilnahme an der Modulprüfung	Nachweis über die Durchführung des berufspraktischen Zeitraumes durch die Praxisstelle
Modulprüfung	Praxisbericht (Arbeitsaufwand: 24 Stunden) mit mündlichem Vortrag (20 Minuten) sowie Teilnahme an 80% aller Seminartermine.
Lernergebnisse und Kompetenzen	<p>In der Praxisphase haben sich die Studierenden im angestrebten Berufsfeld orientiert und die Aufnahme einer späteren internationalen Berufstätigkeit vorbereitet. Im begleitenden Seminar haben die Studierenden ihre Erfahrungen vertieft, reflektiert und mit anderen Teilnehmern ausgetauscht.</p> <p>In der Arbeit an den berufspraktischen Projekten haben sie Erfahrungen mit dem Theorie-Praxis-Transfer gesammelt. Sie sind in der Lage, ihre Fähigkeiten realistisch einzuschätzen und ihre Fortschritte zu analysieren. Außerdem haben sie in dieser Phase Anregungen für die Bachelor-Arbeit gewonnen.</p> <p>Neben der fachlichen Projektarbeit sind sich die Studierenden mit betrieblichen Abläufen und Organisationsformen vertraut geworden. Sie sind in der Lage, selbstständig und verantwortungsbewusst im Kontext des Unternehmens zu arbeiten. Mit der eigenständigen Orientierung im angestrebten Berufsfeld und in der Kooperation beziehungsweise Teamarbeit mit anderen Fachkräften intensivieren sie ihre überfachlichen Kompetenzen; sie kommunizieren mit Kollegen/-innen, Vorgesetzten und Kunden/-innen. Dadurch können sie ihre Rolle in diesen Beziehungen verantwortlich ausfüllen.</p> <p>Sie haben Einblick in wichtige Anwendungsfelder der Informatik und Verständnis gewonnen, besitzen die Fähigkeit zur Beurteilung von</p>

	fremden Software-Systemen und verstehen die Bedeutung der IT für Unternehmen und Gesellschaft. Sie verfügen über die Fähigkeit, einen Vortrag zur beruflichen Tätigkeit selbstständig zu erarbeiten und diesen Vortrag unter Nutzung moderner Präsentationstechniken in einem vorgegebenen Zeitrahmen zu halten.
Inhalte des Moduls	Seminar zur Praxisphase Betreutes Praxisprojekt
Lehrformen des Moduls	Seminar und betreutes Projekt
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	In jedem Semester
Modulkoordination	Prof. Dr. Jens Liebehenschel
Hinweise	Zur Vorbereitung der Praxisphase erfolgt in vorangehenden Semestern eine einführende Veranstaltung, deren Termin jeweils per Aushang mitgeteilt wird.

Unitbeschreibung zum Modul 28: Seminar zur Praxisphase

Name der Unit	Seminar zur Praxisphase
Code	
Name des Moduls	Praxisphase
Inhalte der Unit	<p>Zeitgemäße Präsentation der Praxisprojekte in einem 20-minütigen Vortrag mit</p> <ul style="list-style-type: none"> • Diskussion zum eigenen Vortrag und Beteiligung an der Diskussion zu anderen Vorträgen. • Layout von Präsentationsseiten (Folien/Laptop/Beamer) • Erstellung eines 15-30-seitigen Berichtes zum Vortrag in optisch ansprechender Form mit korrekter Rechtschreibung. Der Bericht muss bis zum Tag des Vortrages vom Praxis-Betrieb durch Stempel und Unterschrift freigegeben worden sein und dem Seminarleiter vor Beginn des Vortrages vorgelegt worden sein.
Lehrformen	Seminar
SWS der Unit	2
Workload (h)	40 h
Anteil der Präsenzzeit	20 h
Anteil Prüfungszeit inkl. Vorbereitung	10 h
Anteil Selbststudium	10 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Jens Liebehenschel
Basis – Literatur	
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine

Unitbeschreibung zum Modul 28: Betreutes Praxisprojekt

Name der Unit	Betreutes Praxisprojekt
Code	
Name des Moduls	Praxisphase
Inhalte der Unit	<p>Qualifizierte Mitarbeit an einem oder an mehreren kleinen Projekten aus den Gebieten</p> <ul style="list-style-type: none"> • Systemanalyse • Projektierung • Anwendungsprogrammierung • Systemprogrammierung • Fachgebiete, die sich aus Modulen des Studiengangs ableiten lassen
Lehrformen	Projekt in einem Betrieb
SWS der Unit	
Workload (h)	500 h
Anteil der Präsenzzeit	500 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	0 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	Prof. Dr. Jens Liebehenschel
Basis – Literatur	Die Literatur wird vom Betrieb gestellt, der das betreute Praxisprojekt durchführt. Ergänzende Literaturquellen sind eigenständig zu beschaffen, gegebenenfalls nach Beratung durch die Dozenten.
Art und Form des Leistungsnachweises der Unit	Nachweis über die Durchführung des berufspraktischen Zeitraumes durch die Praxisstelle
Bewertung des Leistungsnachweises der Unit	Undifferenziert
Hinweise	Das Praxisprojekt umfasst 14 Wochen a 5 Tage. Die Teilnahme an dem Seminar zur Praxisphase muss durch die Firma ermöglicht werden.

Modulbeschreibung zum Modul 29: Bachelor-Arbeit mit Kolloquium

Modultitel	Bachelor-Arbeit mit Kolloquium
Modulnummer	M29
Modulcode	
Studiengang	Informatik – Mobile Anwendungen
Verwendbarkeit des Moduls	
Dauer des Moduls	ein Semester
Empfohlenes Semester im Studienverlauf	6. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (cp) / Workload (h)	12 cp / 360 h
Empfohlene inhaltliche Vorkenntnisse	Alle Module der ersten 5 Studiensemester
Voraussetzungen für die Teilnahme am Modul	Nachweis des Beginns des Moduls M27 Praxisphase in Form eines unterzeichneten Ausbildungsvertrages sowie erfolgreicher Abschluss aller Module M1 bis M26 der ersten 5 Studiensemester
Voraussetzungen für die Teilnahme an der Modulprüfung	keine
Modulprüfung	Bachelor-Arbeit (Gewichtung 80%) mit Kolloquium (Dauer: mindestens 30 Minuten und höchstens 60 Minuten, Gewichtung 20%)
Lernergebnisse und Kompetenzen	<p>Die Studierenden verfügen über die fachlichen und interdisziplinären Fähigkeiten um als InformatikerIn arbeiten zu können.</p> <p>Die Studierenden beherrschen die Kompetenzen in den Bereichen Techniken des wissenschaftlichen Arbeitens, Gesprächsführung, Durchsetzungsfähigkeit, Präsentationstechniken, Projektmanagement, Konfliktmanagement, Planen neuer Systeme, vernetztes Denken, Kreativität und Transferfähigkeit.</p>
Inhalte des Moduls	Bachelor-Arbeit mit Kolloquium
Lehrformen des Moduls	Selbständiges Arbeiten
Sprache	Deutsch
Häufigkeit des Angebots von Modulen	In jedem Semester
Modulkoordination	Prof. Dr. Jörg Schäfer
Hinweise	Das Kolloquium findet am Ende der Arbeit mit den Referenten statt. Im Kolloquium zur Bachelorarbeit soll die oder der Studierende ihre oder seine Bachelorarbeit gegenüber kritischen Fragen verteidigen.

Unitbeschreibung zum Modul 29: Bachelor-Arbeit mit Kolloquium

Name der Unit	Bachelor-Arbeit mit Kolloquium
Code	
Name des Moduls	Bachelor-Arbeit mit Kolloquium
Inhalte der Unit	Abhängig vom individuellen Thema der Bachelor-Arbeit
Lehrformen	Selbstständiges Arbeiten
SWS der Unit	1
Workload (h)	360 h
Anteil der Präsenzzeit	15 h
Anteil Prüfungszeit inkl. Vorbereitung	0 h
Anteil Selbststudium	345 h
Anteil Praxiszeit	0 h
Sprache der Unit	Deutsch
Lehrende/-r	
Basis – Literatur	Keine
Art und Form des Leistungsnachweises der Unit	Keine
Bewertung des Leistungsnachweises der Unit	Keine