

# Modulhandbuch

des Studiengangs

# Informatik - Smart Systems

Bachelor of Science (B.Sc.)

Fb2 Informatik und Ingenieurwissenschaften  
Computer Science and Engineering



## 1.) Empfohlener Studienverlaufsplan: Informatik – Smart Systems

### Für Studierende des Schwerpunktes „Mobile Anwendungen“:

\*=Schwerpunktmodul

							ECTS-Punkte (CP)
Semester 6	Praxisphase 18 CP			Bachelor-Arbeit mit Kolloquium 12 CP			30
Semester 5	Serviceorientierte Architekturen* 5 CP	Human Computer Interaction* 5 CP	Software-Projekt Mobile und verteilte Anwendungen* 5 CP	WP 1 5 CP	WP 2 5 CP	Interdisziplinäres Studium Generale 5 CP	30
Semester 4	Software Engineering – Design 5 CP	Mobile Devices* 5 CP	Mobile Application Exercises* 5 CP	Distributed Systems 5 CP	IT Security 5 CP	Practical Computer Networks and Applications 5 CP	30
Semester 3	Software Engineering – Analysis 5 CP	Databases 5 CP	Embedded Systems 5 CP	Operating Systems 5 CP	Object-oriented Programming in Java – Advanced Course 5 CP	Computer Networks 5 CP	30
Semester 2	Diskrete Mathematik 5 CP	Recht und Datenschutz* 5 CP	Mikro-computertechnik 5 CP	Einführung in die Programmierung mit C und Objekt-orientierte Programmierung Grundlagen 15 CP	Algorithmen und Datenstrukturen 5 CP	Englisch 5 CP	30
Semester 1	Algebra 5 CP	Analysis 5 CP	Digital-technik 5 CP		Einführung in die Informatik und Smart Systems 5 CP		30

### Für Studierende des Schwerpunktes „Technische Informatik“:

\*=Schwerpunktmodul

							ECTS-Punkte (CP)
Semester 6	Praxisphase 18 CP			Bachelor-Arbeit mit Kolloquium 12 CP			30
Semester 5	Digitale Signalver- arbeitung* 5 CP	FPGA Schaltungsentwurf* 5 CP	Databases 5 CP	WP 1 5 CP	WP 2 5 CP	Interdisziplinäres Stu- dium Generale 5 CP	30
Semester 4	Software Engineering – Design 5 CP	Computer Architecture* 5 CP	Embedded Systems Project* 5 CP	Distributed Systems 5 CP	IT Security 5 CP	Practical Computer Net- works and Applications 5 CP	30
Semester 3	Software Engineering – Analysis 5 CP	Elektronik* 5 CP	Embedded Systems 5 CP	Operating Systems 5 CP	Object-oriented Pro- gramming in Java – Ad- vanced Course 5 CP	Computer Networks 5 CP	30
Semester 2	Diskrete Mathematik 5 CP	Elektro- Technik* 5 CP	Mikro- computertechnik 5 CP	Einführung in die Pro- grammierung mit C und Objektorientierte Pro- grammierung Grundlagen 15 CP	Algorithmen und Daten- strukturen 5 CP	Englisch 5 CP	30
Semester 1	Algebra 5 CP	Analysis 5 CP	Digital- technik 5 CP		Einführung in die Infor- matik und Smart Sys- tems 5 CP		30

## 2.) Modul- und Prüfungsübersicht Informatik – Smart Systems

(Module – CP – Dauer – Prüfungsform – Sprache d. Moduls - Gewichtung)

Nr.	Modultitel	CP ECTS	Da uer [Se m.]	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 die Informatik und Smart Systems	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 %  VL: Übungen am Rechner (Gesamtaufwand 30 Stunden)	Deutsch	1/16
5	Digitaltechnik	5	1	Klausur (90 Minuten)  VL: Versuche im Labor in der Gruppe mit schriftlicher Ausarbeitung, Gesamtaufwand 15 Stunden	Deutsch	1/48
6	Englisch	5	1	A portfolio examination consisting of the following:  1. Written examination based on class language training content (90 minutes) weighting 70%  2. Presentation based on class language training content (at least 10 minutes, at most 15 minutes), weighting 30%	English	1/48
7	Diskrete Mathematik	5	1	Klausur (90 Minuten)	Deutsch	1/48
8	Mikrocomputer Technik	5	1	PC-Klausur (90 Minuten) VL: Laborversuche mit schriftlicher Ausarbeitung (80 h Gesamtaufwand)	Deutsch	1/48
9	Algorithmen und Datenstrukturen	5	1	Klausur (90 Minuten)	Deutsch	1/48
10a	Schwerpunkt Mobile Anwendungen: Recht und	5	1	Klausur (90 Minuten)	Deutsch	1/48

Nr.	Modultitel	CP ECTS	Da uer [Se m.]	Prüfungsform	Sprache	Gew.
	Datenschutz					
10b	Schwerpunkt Technische Informatik: Elektrotechnik	5	1	Klausur (90 Minuten)  VL*: Versuche im Labor mit schriftlicher Ausarbeitung zu jedem Versuch (Gesamtaufwand 9 Stunden)	Deutsch	1/48
11	Software Engineering - Analysis	5	1	Computer-based project with documentation (submission period 9 weeks, processing time 60 hours) Pass/fail	English	Keine
12	Embedded Systems	5	1	Written computer-based examination (90 minutes) VL: Written laboratory report (processing time 80 h)	English	1/36
13	Object-oriented Programming in Java – Advanced Course	5	1	Computer-based project with documentation (submission period 9 weeks) with presentation (at least 15, at most 30 minutes) VL: Laboratory exercises with documentation (processing time 80 hours)	English	1/36
14a	Databases(für Studierende im Schwerpunkt Mobile Anwendungen), entspricht Modul 25b	5	1	Written examination (120 minutes)	English	1/36
14b	Schwerpunkt Technische Informatik: Elektronik			Klausur (90 Minuten)  VL: Versuche im Labor mit schriftlicher Ausarbeitung zu jedem Versuch (Gesamtaufwand 15 Stunden)	Deutsch	1/36
15	Computer Networks	5	1	Written examination (90 minutes)	English	1/36
16	Operating Systems	5	1	Written examination (90 minutes)	English	1/36
17	Software Engineering - Design	5	1	Written examination (120 minutes)  VL: Computer-based exercises with written documentation (processing time 36 hours)	English	5/144
18a	Schwerpunkt Mobile Anwendungen: Mobile Devices	5	1	Project work (submission period 12 weeks) with presentation (at least 10	English	5/144

Nr.	Modultitel	CP ECTS	Da uer [Se m.]	Prüfungsform	Sprache	Gew.
				minutes, at most 20 minutes)		
18b	Schwerpunkt Technische Informatik: Computer Architecture	5	1	Written examination (90 minutes) VL: Laboratory Exercises with written documentation (processing time 60 hours)	English	5/144
19	IT-Security	5	1	Written examination (90 minutes)	English	5/144
20	Distributed Systems	5	1	Written examination (90 minutes) VL: Laboratory exercises (processing time 30 hours)	English	5/144
21	Practical Computer Networks and Applications	5	1	Computer-based project with documentation (submission period 9 weeks, processing time 60 hours)  Pass/fail	English	Keine
22a	Schwerpunkt Mobile Anwendungen: Mobile Application Exercises	5	1	Project work (submission period 12 weeks) with presentation in groups (at least 10, at most 20 minutes per person)	English	5/144
22b	Schwerpunkt Technische Informatik: Embedded Systems Project			Project work (submission period 15 weeks) with presentation (at least 10, at most 20 minutes)	English	1/24
23a	Schwerpunkt Mobile Anwendungen: Serviceorientierte Architekturen	5	1	Projektarbeit (Bearbeitungszeit 8 Wochen) mit Präsentation (mindestens 15, höchstens 20 Minuten)	Deutsch	5/144
23b	Schwerpunkt Technische Informatik: Digitale Signalverarbeitung	5	1	Klausur (90 Minuten)	Deutsch	5/144
24a	Schwerpunkt Mobile Anwendungen: Human Computer Interaction	5	1	Projektarbeit (Bearbeitungszeit 12 Wochen) mit Präsentation (mindestens 15, höchstens 20 Minuten)	Deutsch	5/144
24b	Schwerpunkt Technische Informatik: FGPA-Schal- tungsentwurf	5	1	Projektarbeit mit schriftlicher Ausarbeitung (Bearbeitungszeit 6 Wochen) und Präsentation (mindestens 10, höchstens 20 Minuten)	Deutsch	5/144
25a	Schwerpunkt Mobile Anwendungen: Software-Projekt Mobile und Verteilte Anwendungen	5	1	Projektarbeit (Bearbeitungszeit 15 Wochen) mit Präsentation (mindestens 10, höchstens 20	Deutsch	1/24

Nr.	Modultitel	CP ECTS	Da uer [Se m.]	Prüfungsform	Sprache	Gew.
				Minuten)		
25b	Databases (für Studierende im Schwerpunkt Technische Informatik), entspricht Modul 14a	5	1	Written examination (120 minutes)	English	5/144
26	Wahlpflichtmodul 1	5	1	abhängig vom gewählten Modul	Deutsch	5/144
27	Wahlpflichtmodul 2	5	1	abhängig vom gewählten Modul	Deutsch	5/144
28	Interdisziplinäres Studium Generale	5	1	Projektarbeit (Bearbeitungszeit: Variabel, je nach Modulexemplar) mit Präsentation (Variabel je nach Modulexemplar)	Deutsch	5/144
29	Praxisphase	18	1	Praxisbericht (Bearbeitungszeit 12 Wochen) mit Präsentation (mindestens 15, höchstens 20 Minuten)	Deutsch	10/144
30	Bachelor-Arbeit mit Kolloquium	12	1	Bachelor-Arbeit (Bearbeitungszeit 9 Wochen) mit Kolloquium (mindestens 30, höchstens 60 Minuten)	Deutsch	22/144

VL=Vorleistung

### 3.) Modulbeschreibungen

<b>Modulnummer: 1</b>		<b>Algebra</b>		
<b>Modultitel (englisch)</b>		Algebra		
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Keine		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"><li>• abstrakte mathematische Begriffe definieren</li><li>• strukturelle und logische Grundlagen elektronischer Informationsverarbeitung erläutern und in Beziehung zueinander setzen</li><li>• sich abstrakte Begriffe selbständig erarbeiten und Grundlegende Techniken oder Verfahren der Algebra aneignen.</li><li>• 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</li><li>• 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</li></ul>		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Algebra		Vorlesung		4 SWS
b) Übung Algebra		Übung		2 SWS
<b>Empfohlenes Fachsemester</b>		1. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 6 SWS / 90 Stunden		<b>Selbststudium (h)</b> 60	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) Grundlagen <ul style="list-style-type: none"><li>• Aussagen- und Prädikatenlogik, Beweismethoden</li><li>• Mengen, Relationen inkl. Funktionen</li></ul> Elementare Zahlentheorie <ul style="list-style-type: none"><li>• Primzahlen und Teiler</li><li>• Modulo Rechnung, Diophantische Gleichungen, Chinesischer Restsatz</li><li>• Anwendung Kryptographie</li></ul> Algebraische Strukturen <ul style="list-style-type: none"><li>• Halbgruppen, Gruppen, Permutationsgruppen, Untergruppen, Satz von Lagrange</li><li>• Ringe, Polynomringe, Nullteiler, Testklassenring</li><li>• Körper, Primzahlkörper</li></ul> Lineare Algebra <ul style="list-style-type: none"><li>• Vektorräume</li><li>• Lineare Gleichungssysteme</li><li>• Matrizen: Rang, Matrixprodukt, Inverse, Eigenwerte und charakteristisches Polynom</li><li>• Lineare Abbildungen: Kern, Basis, Basiswechsel</li></ul> Spezielle lineare Abbildungen: Spiegelungen, Projektionen, Drehungen sowie		



	Matrixdarstellung und Komposition von Abbildungen b) Aufgaben und Beispiele zu den Vorlesungsthemen.
<b>Literatur</b>	<ul style="list-style-type: none"> <li>• Brill, Manfred: Mathematik für Informatiker, Hanser, 2005</li> <li>• Denecke, Klaus: Algebra und Diskrete Mathematik für Informatiker, Vieweg + Teubner, 2003</li> <li>• Teschl, Gerold und Susanne: Mathematik für Informatiker, Band 1 Diskrete Mathematik und Lineare Algebra, Springer 2008</li> <li>• Witt, Kurt-Ulrich: Algebraische Grundlagen der Informatik, Vieweg, 2007</li> </ul> <p>- Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
<b>Modulkoordination</b>	Prof. Dr. Ruth Schorr
<b>Lehrende</b>	Prof. Dr. Egbert Falkenberg, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr, Thekla Unthan, Prof. Dr. Karsten Weronek
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	

<b>Modulnummer: 2</b>		<b>Analysis</b>	
<b>Modultitel (englisch)</b>			
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine	
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"><li>• die wichtigsten Begriffe, Verfahren und Techniken der Differential- und Integralrechnung einschließlich Potenzreihen sowie komplexe Zahlen benennen und in Berechnungen anwenden</li><li>• die typischen Methoden der Analysis unterscheiden</li><li>• einfache Anwendungsprobleme in mathematische Aufgabenstellungen umsetzen und diese lösen</li><li>• die Voraussetzungen und Grenzen der Methoden der Differenzial- und Integralrechnung erläutern</li><li>• die zentralen Aussagen zur Konvergenz von Folgen, Reihen und Funktionen erläutern und damit Beispiele behandeln</li><li>• die wichtigsten Eigenschaften der elementaren Funktionen wiedergeben</li><li>• das Konzept der Stetigkeit und Differenzierbarkeit beschreiben und daraus Eigenschaften von Funktionen herleiten</li><li>• eindimensionale (auch unentgeltliche) Integrale berechnen und die zugrunde liegende Theorie erläutern</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Analysis		Vorlesung	4 SWS
b) Übung Analysis		Übung	2 SWS
<b>Empfohlenes Fachsemester</b>		1. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 6 SWS / 90 Stunden		<b>Selbststudium (h)</b> 60	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>	a) • 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 b) Aufgaben und Beispiele zu den Vorlesungsthemen		
<b>Literatur</b>	• Mathematik für das Ingenieurstudium, Jürgen Koch, Martin Stämpfle, Hanser, 4. neu bearbeitete Auflage, 2018 - Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.		
<b>Modulkoordination</b>	Prof. Dr. Egbert Falkenberg		
<b>Lehrende</b>	Prof. Dr. Egbert Falkenberg, Prof. Dr. Doina Logofătu, Dr. Maike Kamlage, Prof. Dr. Ruth Schorr		
<b>Hinweise</b>			

(insbesondere empfohlene inhaltliche Vorkenntnisse)	
-----------------------------------------------------	--

<b>Modulnummer: 3</b>		<b>Einführung in die Informatik und Smart Systems</b>	
<b>Modultitel (englisch)</b>			
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>			
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Keine	
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls können die Studierenden: <ul style="list-style-type: none"><li>• die unterschiedlichen Bereiche der Informatik beschreiben</li><li>• die Verarbeitung und Berechnung von Zahlen auf der Hardware-Ebene durchführen und systematisch beschreiben</li><li>• den Aufbau und die Wirkungsweise eines Mikroprozessors darlegen und erläutern</li><li>• die Vorgänge im Rechner bei Programmerstellung und Programmabläufen darstellen und begründen</li><li>•die grundlegenden Klassen formaler Sprachen, Grammatiken und Automatenmodelle beschreiben und deren Bedeutung für andere Bereiche der Informatik, z.B. Compilerbau, beurteilen</li><li>• die Besonderheiten von mobilen Anwendungen nennen und erläutern</li><li>•smarte Systeme nennen und enthaltene Informatikanteile beschreiben</li><li>•die Fähigkeiten und Anwendungen von „Überall Rechner“ einordnen</li><li>• wissenschaftliche Definitionen und Begriffe erläutern und verwenden</li><li>•Prozesse und Technik unter Bedingungen der gesellschaftlichen Verantwortung erläutern</li><li>•die Vertiefungsrichtungen des Studiengangs Informatik - Smart Systems beschreiben</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Einführung in die Informatik und Smart Systems		Vorlesung	2 SWS
b) Übung Einführung in die Informatik und Smart Systems		Übung	2 SWS
c) Einführung in das wissenschaftliche Arbeiten		Übung	1 SWS
<b>Empfohlenes Fachsemester</b>		1. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>		a) Das Modul liefert als Einführung in die Informatik und Smart Systems allgemeine und grundlegende Kenntnisse und Fähigkeiten für das gesamte Studium. <ul style="list-style-type: none"><li>• Informationsdarstellung im Rechner (Speicherformate, Logische Größen, Zeichen, Zahlen, Zahlensysteme, Ganze Zahlen, Gleitkommazahlen)</li><li>• Einführung in Rechnerarchitekturen</li><li>• Verarbeitung von Zahlen</li><li>• Massenspeicher</li><li>• Funktionsweise eines Mikroprozessors</li></ul> •Formale Sprachen und zugehörige Automatenmodelle <ul style="list-style-type: none"><li>• Herausforderungen für mobile Anwendungen</li><li>• Ressourcennutzung (Speicher, Rechenzeit, Stromverbrauch)</li></ul> •Smarte Systeme <ul style="list-style-type: none"><li>• „Überall-“ Rechnersysteme</li></ul> •Vertiefungsrichtungen des Studiengangs Informatik - Smart Systems b) Aufgaben und Beispiele zu den Vorlesungsthemen. Um dem besonderen Fokus des Studiengangs Informatik - Smart Systems Rechnung zu tragen, werden Beispiele auch aus dem Bereich Mobile Endgeräte und Embedded Systems verwendet. c) Wissenschaftliches Arbeiten ist die Basis für Hausarbeiten, schriftliche Ausarbeitungen und Präsentationen. Dieses erste Modul konzentriert sich auf die Herangehensweise und die schriftliche Ausarbeitung.	

	<ul style="list-style-type: none"> <li>• Grundlagen wissenschaftlicher Herangehensweisen</li> <li>• Fachliteratur recherchieren, bearbeiten, analysieren und darstellen</li> <li>• Daten erheben, aufbereiten, analysieren und präsentieren</li> </ul>
<b>Literatur</b>	<ul style="list-style-type: none"> <li>• H. Ernst, J. Schmidt, G. Beneken: Grundkurs Informatik, Springer Vieweg</li> <li>• H. Herold, B. Lurz, J. Wohlrab, M. Hopf: Grundlagen der Informatik, Pearson Studium</li> <li>• J. E. Hopcroft, R. Motwani, J. D. Ullman - Einführung in Automatentheorie, Formale Sprachen und Berechenbarkeit, Pearson Studium</li> <li>• HD. W. Hoffmann: Theoretische Informatik, Hanser</li> <li>• M. Weiser (1991) The Computer for the 21st Century. Scientific American 265(3): 94–104</li> </ul> <p>- Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
<b>Modulkoordination</b>	Prof. Dr. Jens Liebehenschel
<b>Lehrende</b>	Prof. Dr. Jens Liebehenschel, Prof. Dr. Matthias Deegener
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Vorausgesetzt werden die zum Erwerb der Fachhochschulreife erwarteten Kenntnisse der Mathematik

<b>Modulnummer: 4</b>		<b>Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen</b>	
<b>Modultitel (englisch)</b>			
<b>Dauer des Moduls</b> Zwei Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 15 CP / 450 Stunden	
<b>Verwendbarkeit des Moduls</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Voraussetzung für die Teilprüfungsleistung 2: bestandene Teilprüfungsleistung 1	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Übungen am Rechner (Gesamtaufwand 30 Stunden)  bestanden/nicht bestanden	
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Zwei Teilprüfungsleistungen: 1. Eigenständige Programmierung C in Form einer Klausur am Rechner (120 Minuten), Gewichtung: 50%  2.Eigenständige Objektorientierte Programmierung in Form einer Klausur am Rechner (120 Minuten), Gewichtung: 50% <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"><li>• die wichtigsten Sprachelemente und Bibliotheksfunktionen benennen und unterscheiden</li><li>• Lösungen für einfache Aufgabenstellungen als strukturierten Entwurf formulieren und in C umsetzen</li><li>• Methoden zur Fehlererkennung und Fehlerbeseitigung fallbezogen einsetzen</li><li>• Denk- und Herangehensweisen der objektorientierten Programmierung skizzieren und kritisch vergleichen</li><li>• Begriffe wie Datenkapselung, Wiederverwendung von Code, Klassen, Vererbung und Polymorphie erläutern und einordnen.</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Einführung in die Programmierung mit C		Vorlesung	2 SWS
b) Übung Einführung in die Programmierung mit C		Übung	2 SWS
c) Vorlesung Objektorientierte Programmierung Grundlagen		Vorlesung	2 SWS
d) Übung Objektorientierte Programmierung Grundlagen		Übung	2 SWS
<b>Empfohlenes Fachsemester</b>		1. und 2. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 8 SWS / 120 Stunden		<b>Selbststudium (h)</b> 330	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>		a) • Einführendes Beispiel mit grundlegenden Sprachelementen <ul style="list-style-type: none"><li>• Analyse einfacher Aufgabenstellungen und Erstellen eines strukturierten Lösungsvorschlags</li><li>• Editieren, Übersetzen, Ausführen von Programmen</li><li>• Elementare Datentypen, Variablen und Arithmetik</li><li>• Ein-/Ausgabe Verzweigung und Schleifen</li><li>• Felder, Zeichenkette</li><li>• Fehlersuche und Fehlerbeseitigung</li><li>• Zeiger, dynamische Speicherverwaltung</li><li>• Unterprogramme (Funktionen) und Parameter, modularer Programmaufbau,</li><li>• Bibliotheksfunktionen Dateien Strukturierte Datentypen</li></ul> b) Übungen zur Einführung in die Programmierung, Umsetzung von Aufgaben mittels eines C-Programms, Feedback zum Lernprozess c) Inhaltlichen Schwerpunkt bilden: <ul style="list-style-type: none"><li>• Objektorientierte Ansatz – erstes Beispiel; evtl. Abgrenzung zur prozeduralen Programmierung</li><li>• Klassenkonzept, UML-Darstellung</li><li>• Klassen, Objekte</li><li>• Konstruktor incl. Überladung, Destruktor</li><li>• Kopieren und Zuweisen von Objekten</li><li>• Operatoren, dynamische Speicherverwaltung</li><li>• Referenzen</li></ul>	

	<ul style="list-style-type: none"> <li>• Setter/Getter Methoden</li> <li>• Statische Objektkomponenten</li> <li>• Vererbung, Zugriffsrechte</li> <li>• Polymorphie, spätes Binden</li> </ul> <p>Weiterhin bieten sich ausgewählte Themen an, z.B.</p> <ul style="list-style-type: none"> <li>• Mehrfachvererbung, virtuelle Vererbung</li> <li>• Klassen- und Funktions-Templates</li> </ul> <p>Standard Libraries und Standard Template Libraries</p> <p>d) Beispiele und Programmieraufgaben mit Bezug zu den Vorlesungsinhalten, qualifiziertes Feedback durch die Lehrenden</p>
<b>Literatur</b>	<ul style="list-style-type: none"> <li>• Brian W. Kernighan, Dennis Ritchie, The C Programming Language, Prentice Hall Software, 2000</li> <li>• Erlenkötter, H., C Programmieren von Anfang an, Rowohlt, 2008</li> <li>• Mittelbach, H., Einführung in C, Fachbuchverlag Leipzig, 2002</li> <li>• Die Programmiersprache C. Ein Nachschlagewerk, Regionales Rechenzentrum für Niedersachsen/Universität Hannover, 1RRZN.</li> <li>• Das Handbuch zu den zum GNU-C-Compiler gehörenden Standardfunktionen (GNU C Library) kann man einsehen unter <a href="http://www.gnu.org/software/libc/manual/">http://www.gnu.org/software/libc/manual/</a></li> <li>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</li> </ul>
<b>Modulkoordination</b>	Prof. Dr. Barış Sertkaya
<b>Lehrende</b>	Prof. Dr. Barış Sertkaya, Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Carsten Biemann
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	<p>Es werden keine Programmierkenntnisse vorausgesetzt.</p> <p>Die Modulteilprüfung C erfolgt im 1. Semester des/der Studierenden. Wiederholungsprüfungen sind gegebenenfalls in dem auf das Nichtbestehen folgenden Semester abzulegen.</p>

<b>Modulnummer: 5</b>		<b>Digitaltechnik</b>		
<b>Modultitel (englisch)</b>		Logic Design		
<b>Dauer des Moduls</b> Ein Semester		<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul		<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>		Elektro- und Informationstechnik (B.Eng.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Versuche im Labor in der Gruppe mit schriftlicher Ausarbeitung (Gesamtaufwand 15 Stunden)  bestanden/nicht bestanden		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls sind die Studierenden in der Lage <ul style="list-style-type: none"><li>• die Vorteile des dualen Zahlensystems zu erläutern, die Vorteile der Darstellung negativer Zahlen im Zweierkomplement darzulegen und Rechenoperationen in diesem Zahlensystem durchzuführen;</li><li>• die Eigenschaften unterschiedlicher Codierungen zu benennen und für eine bestimmte Aufgabe einen geeigneten Code auszuwählen, insbesondere zur Fehlererkennung und Fehlerkorrektur;</li><li>• boolesche Gleichungen aufzustellen und umzuformen sowie diese in digitale Schaltungen umzusetzen;</li><li>• die Unterschiede von Schaltnetzen und Schaltwerken und die Funktionsweise und den Aufbau von Flip-Flops zu erläutern;</li><li>• die Vorteile und Nachteile asynchroner und synchroner Schaltungen zu erläutern;</li><li>• einfache Schaltnetze und Schaltwerke zu analysieren sowie mit unterschiedlichen Entwurfsverfahren zu entwerfen und zu optimieren;</li><li>• einfache Zustandsautomaten zu entwerfen und als Schaltung zu realisieren;</li><li>• die verschiedenen Möglichkeiten zur technischen Realisierung digitaler Schaltungen darzulegen;</li><li>• in Teams Laborversuche zu bearbeiten, Erfahrungen auszutauschen und Ergebnisse in schriftlichen Berichten zu präsentieren;</li></ul>		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Digitaltechnik Vorlesung		Seminaristischer Unterricht		3 SWS
b) Digitaltechnik Übung		Übung		1 SWS
c) Digitaltechnik Labor		Laborversuche in Kleingruppen		1 SWS
<b>Empfohlenes Fachsemester</b>		1. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) •Zahlendarstellung, Codes •Logische Verknüpfungen •Schaltalgebra (Boolesche Algebra) •Karnaugh-Veitch-Diagramme •Analyse und Synthese von Schaltnetzen •praktische Realisierung digitaler Schaltungen •Schaltnetze und Schaltwerke •Flip-Flops •Zähler •Addierer •Multiplexer •Zustandsautomaten •Entwurfsverfahren digitaler Schaltungen b) an die Inhalte der Vorlesung Digitaltechnik angepasste Übungsaufgaben c) An die Inhalte der Vorlesung Digitaltechnik angepasste Laborversuche. Die Studierenden entwickeln im Rahmen kleiner Gruppen funktionsfähige Hardware-Lösungen für die jeweils gestellten Aufgaben. Während der Präsenzphase werden die selbständig ermittelten Symbolschaltungen in funktionsfähige Hardware umgesetzt, die sich vorführen und mit anderen		



	Lösungen vergleichen lässt, sodass es Betreuenden möglich ist, die erworbenen Kenntnisse und die erzielten Fortschritte bei der Schaltungsentwicklung begleitend zu verfolgen.
<b>Literatur</b>	<ul style="list-style-type: none"> <li>• Borgmeyer: Grundlagen der Digitaltechnik, Hanser-Verlag</li> <li>• Pernards: Digitaltechnik, Hüthig-Verlag</li> <li>• Fricke: Digitaltechnik, Vieweg-Verlag</li> <li>• Jorke: Rechnergestützter Entwurf digitaler Schaltungen, Fachbuchverlag Leipzig</li> <li>• Jeweils in der aktuellsten Auflage. Weiterführende Literatur wird in der jeweiligen Lehrveranstaltung bekanntgegeben</li> </ul>
<b>Modulkoordination</b>	Prof. Dr.-Ing. Heiko Hinkelmann
<b>Lehrende</b>	Prof. Dr.-Ing. Heiko Hinkelmann, Prof. Dr. Thomas Hollstein
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	c) Zur Teilnahme am Labor ist die rechtzeitige Anmeldung erforderlich.

<b>Module number: 6</b>	<b>English</b>		
<b>Module title (English)</b>	English		
<b>Module duration</b> One semester	<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English	
<b>Module type</b> Compulsory module	<b>Availability</b> Every semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h	
<b>Module relevance</b>	Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)		
<b>Admission requirement(s) for the module and module examination</b>	None		
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments for admission to the module examination (type, workload, duration and grading scheme)</b>	None. Active participation in language practice sessions related to aural skills, reading, writing and oral communication in a variety of forms (with 75% certified participation) is essential in order to successfully complete the portfolio examination. Students must complete the additional e-learning module independently in order to participate effectively in the classroom <u>Workload:</u> pass/fail		
<b>B) Final module assessment (type, workload, duration and grading scheme)</b>	A portfolio examination consisting of: 1. Written examination based on class language training content (90 minutes) weighting 70 % 2. Presentation based on class language training content (at least 10 minutes, at most 15 minutes), weighting 30 % The examination is considered passed if a student has gained at least 50 % of total attainable points <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS		
<b>Learning outcomes / skills</b>	<p>Application, Use and Production of Knowledge:</p> <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 in 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. Communication and Collaboration Skills:</p> <p>Students can handle typical professional situations of international communication in English with both specialists in their own field and non-specialists. They can follow English-medium lectures and other language-based activities when studying abroad and also have the necessary skills for doing an internship in English. Students can cope with the general requirements of communicating in English in their professional field as well as in the academic environment. Students can understand the main ideas of complex texts, including technical discussions in their field of specialisation.</p> <p>The students can use language structures and vocabulary - with special consideration of subject-relevant lexical fields. They can understand a certain range of different types of text appropriate to their language level and related to diverse application areas related to their area of study, e.g. technical documents, reports, academic paper, and simple specialist articles.</p>		
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>	<b>Learning and teaching formats</b>		<b>Course SCH</b>
a) Practice session seminar	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.		4 SWS
<b>Recommended semester of study</b>	Semester 1 and 2		
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h	<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0	
<b>Module content</b>	<ul style="list-style-type: none"><li>• Professional communication in English, both with specialists of the same field and non-specialists;</li><li>• Presentation skills;</li><li>• Active listening skills and reading comprehension;</li><li>• Subject-specific vocabulary and terminology;</li><li>• Verbal communication and writing skills..</li></ul>		
<b>Literature</b>	Authentic sources such as recent research published in scientific and technical journal, blogposts, vlogs as well as business and governmental newsletters and reports		
<b>Module coordination</b>	L. Banerji/ S. Schmidt		

<b>Teachers</b>	Members of the staff of the University Language Center (fully-employed lecturers, or freelancers supervised by them)
<b>Notes</b> (In particular, recommended prior knowledge)	

<b>Modulnummer: 7</b>	<b>Diskrete Mathematik</b>	
<b>Modultitel (englisch)</b>	Discrete Mathematics	
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Sommersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden
<b>Verwendbarkeit des Moduls</b>	Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>	Keine	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>		
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )	Keine	
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )	Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>	Nach Abschluss des Moduls können die Studierenden: <ul style="list-style-type: none"><li>• aufbauend auf den Modulen Algebra und Analysis auf vertiefter Ebene mit abstrakten Begriffen operieren</li><li>• die wichtigsten mathematischen Techniken für Anwendungen in den Kerndisziplinen der Informatik (Theoretische Informatik und Rechnerarchitektur, Betriebssysteme, Rechnernetzen etc...) in Übungen anwenden</li><li>• Begriffe und Verfahren der diskreten Mathematik anwenden</li><li>• Lösungsverfahren der diskreten Mathematik in einfachen Anwendungsfällen selbständig einsetzen und ihre Ergebnisse bewerten</li><li>• Bezüge der diskreten Mathematik zu Kerndisziplinen der Informatik herstellen und Verfahren der diskreten Mathematik in diesen Kontexten adäquat anwenden</li><li>• auf vertieftem Niveau mit formalen Systemen und Modellen umgehen</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>		
<b>Titel der Lehrveranstaltung</b>	<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Diskrete Mathematik	Vorlesung	4 SWS
b) Übung Diskrete Mathematik	Übung	2 SWS
<b>Empfohlenes Fachsemester</b>	2. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 6 SWS / 90 Stunden	<b>Selbststudium (h)</b> 60	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>	a) Abzählungen <ul style="list-style-type: none"><li>• Zählprinzipien</li><li>• Zählkoeffizienten: Variationen, Kombinationen, Mengen- und Zahlpartitionen, Stirling Zahlen</li><li>• Catalan-Zahlen</li><li>• Lösung von linearen, homogenen und inhomogenen Rekursionsgleichungen</li></ul> Einführung in die Codierungstheorie <ul style="list-style-type: none"><li>• Grundbegriffe: Codierung, Decodierung, Fehlererkennung, Hamming und Singleton Schranke</li><li>• Lineare Codes: Generator- und Kontrollmatrix, Hamming-Code, Reed-Solomon-Code</li></ul> Graphentheorie <ul style="list-style-type: none"><li>• Grundbegriffe</li><li>• Euler'sche und Hamilton'sche Graphen, planare Graphen, Färbungen</li><li>• Bäume inkl. Binäre Bäume</li><li>• Netzwerkalgorithmen: Maximale Spannbäume, Kürzeste Wege, Maximale Flüsse</li></ul> Matching in bipartiten Graphen b) Aufgaben und Beispiele zu den Vorlesungsthemen.	
<b>Literatur</b>	<ul style="list-style-type: none"><li>• Aigner, Martin: Diskrete Mathematik, Vieweg + Teubner, 2006</li><li>• Diestel, Reinhard: Graphentheorie, Springer, 2010</li><li>• Grimaldi, Ralph P.: Discrete and Combinatorial Mathematics, Addison Wesley, 2003</li><li>• Steger, Angelika: Diskrete Strukturen 1, Springer, 2007</li><li>• Teschl, Gerold und Susanne: Mathematik für Informatiker, Band 1 Diskrete Mathematik und Lineare Algebra, Springer, 2008</li><li>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</li></ul>	
<b>Modulkoordination</b>	Prof. Dr. Ruth Schorr	
<b>Lehrende</b>	Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr	

<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Die Inhalte des Moduls Algebra und Analysis des ersten Studienseesters werden vorausgesetzt.
------------------------------------------------------------------------	----------------------------------------------------------------------------------------------

<b>Modulnummer: 8</b>		<b>Mikrocomputer Technik</b>		
<b>Modultitel (englisch)</b>		Microcomputing techniques		
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Sommersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Informatik- Mobile Anwendungen (B.Sc.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Laborversuche mit schriftlicher Ausarbeitung (80h Gesamtaufwand)  bestanden/nicht bestanden		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		PC-Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls verfügen die Studierenden über die folgenden Kompetenzen <ul style="list-style-type: none"><li>• Sie ordnen elektrotechnische Randbedingungen beim Betrieb digitaler Schaltungen ein.</li><li>• Sie können den systematischen Entwurf digitaler Schaltungen erläutern.</li><li>• Sie können die wichtigsten architektonischen Prinzipien für den Entwurf von Rechenanlagen beschreiben.</li><li>• Sie unterscheiden die verschiedenen Ebenen der Befehlsverarbeitung in Assemblersprachen.</li><li>• Sie wenden die wichtigsten Prinzipien von Assemblersprachen an.</li><li>• Sie können die Funktionsweise und den Aufbau von Mikrocomputern beschreiben.</li><li>• Sie realisieren hardwarenahe Programme in Assembler und einer Hochsprache.</li><li>• Sie zählen typische Anwendungsgebiete auf und entwickeln mikroprozessorgesteuerte Systeme.</li><li>• Sie können strukturiert und kreativ Probleme lösen</li><li>• Sie arbeiten im Rahmen von Laborversuchen mit anderen im Team zusammen.</li><li>• Sie können die Ressourcennutzung (Speicher, Laufzeit, Stromverbrauch) von Programmen für Mikrocomputern beurteilen.</li></ul>		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Mikrocomputertechnik		Vorlesung, Übung, Labor		2 SWS
b) Labor Mikrocomputertechnik		Laborübungen		2 SWS
<b>Empfohlenes Fachsemester</b>		2. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 4 SWS / 60 Stunden		<b>Selbststudium (h)</b> 90	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) • Digitaltechnik: digitallogische Bausteine und Schaltungen, sequentielle Schaltungen. <ul style="list-style-type: none"><li>• Aufbau und Design von Mikrocomputern</li><li>• Funktionsweise eines Mikrocontrollers,</li><li>• Steuerwerk, Rechenwerk, Register, Ports, Programm- und Datenspeicher, Stackspeicher, Befehlsabarbeitung, Timing-Diagramm.</li><li>• Programmierung in Assembler: Befehlsarten, Special Function Register, Adressierungsarten, Ein-/Ausgabe von Daten mittels Ports,</li><li>• Unterprogramme, Tabellenbearbeitung, Echtzeitprogramme, Interrupts.</li><li>• Grundlagen der hardwarenahen Programmierung in einer Hochsprache.</li></ul> b) Durchführung von Laborversuchen in Assembler und in einer Hochsprache, die auf einem Mikrocontrollerboard ausgeführt werden.		
<b>Literatur</b>		<ul style="list-style-type: none"><li>• Nauth: Embedded Intelligent Systems, Oldenbourg Verlag, 2005</li><li>• Schiffmann, Schmitz: Technische Informatik 1: Grundlagen der digitalen Elektronik, Springer-Verlag, 2013</li><li>• Fricke: Digitaltechnik: Lehr- und Übungsbuch für Elektrotechniker und Informatiker. Springer Fachmedien Wiesbaden, 2021.</li><li>• Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</li></ul>		
<b>Modulkoordination</b>		Prof. Dr. Matthias Deegener		

<b>Lehrende</b>	Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel, Prof. Dr. Peter Tawdross
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	b) Zur Teilnahme am Labor ist die rechtzeitige Anmeldung erforderlich. Näheres wird durch Aushang geregelt.

<b>Modulnummer: 9</b>		<b>Algorithmen und Datenstrukturen</b>	
<b>Modultitel (englisch)</b>		Algorithms and data structures	
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Sommersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine	
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		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. Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"><li>• Algorithmen und Datenstrukturen analysieren und dokumentieren</li><li>• geeignete neue Datenstrukturen (aufbauend auf den in dem Kurs behandelten Standardstrukturen) gestalten</li><li>• Algorithmen zur Bearbeitung entwickeln und nach den gelernten Methoden darstellen</li><li>• Lösungsmöglichkeiten hinsichtlich Korrektheit, Komplexität und Eleganz beurteilen.</li><li>• Durch die Analyse und Ausarbeitung von abstrakten Algorithmen werden wissenschaftliches und exaktes Arbeiten, sowie strukturierte und kreative Problemlösung eingeübt.</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Algorithmen und Datenstrukturen		Vorlesung	4 SWS
b) Übung Algorithmen und Datenstrukturen		Übung	2 SWS
<b>Empfohlenes Fachsemester</b>		2. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 6 SWS / 90 Stunden		<b>Selbststudium (h)</b> 60	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>		a) Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden. Algorithmen: <ul style="list-style-type: none"><li>• Definition eines Algorithmus und Anforderungen an ihn, semantische Korrektheit</li><li>• Beschreibungsformen von Algorithmen (natürliche Sprache, Pseudocode, Struktogramm, Datenflußdiagramm, Programmablaufplan)</li><li>• Kapazitätsbetrachtungen: Platz- und Rechenzeit, asymptotische Notationen, Kapazitätsmaße (worst case, average case), P-NP- Problematik</li><li>• Typen algorithmischer Vorgehensweisen (Rekursion, Greedy, Divide and Conquer, Backtracking ...)</li><li>• Grundlegende Begriffe der Parallelen Algorithmen: Work- Law, Span-Law, Speed-Up, Parallelism, Notwendigkeit für Synchronisierungsverfahren</li></ul> Datenstrukturen: <ul style="list-style-type: none"><li>• elementare Datenstrukturen</li><li>• lineare Standardstrukturen (Felder, Listen, Stapel, Warteschlangen)</li><li>• Bäume</li><li>• Mengen</li><li>• Graphen</li><li>• Algorithmen zu den Grundproblemen der Informatik</li><li>• Sortieren</li><li>• Suchen</li><li>• Hashing</li></ul> b) Aufgaben und Beispiele zu den Themen des Moduls.	



<b>Literatur</b>	<ul style="list-style-type: none"> <li>• Aho, A. V., Hopcroft, J. E., Ullman, J. D.: The Design and Analysis of Computer Algorithms, Addison-Wesley 1974</li> <li>• Brunskill, D., Turner, J.: Understanding Algorithms and Data Structures, McGraw-Hill 1996</li> <li>• Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. Algorithmen - Eine Einführung. Oldenbourg Wissenschaftsverlag, 2. Auflage, 2007</li> <li>• Güting, R. H.: Datenstrukturen und Algorithmen, Teubner 1992</li> <li>• Lewis, T. G., Smith, M. Z.: Datenstrukturen und ihre Anwendung, Oldenbourg 1978</li> <li>• Mehlhorn, K.: Datenstrukturen und effiziente Algorithmen, Bd 1, 2, 3, Teubner 1986</li> <li>• Preparata, F. P., Shamos, M. I.: Computational Geometry, Springer 1985</li> <li>• T. Ottmann, P. Widmayer: Algorithmen und Datenstrukturen. BI Wissenschaftsverlag, Mannheim</li> <li>• H. Reß, G. Vorbeck: Datenstrukturen und Algorithmen. Hanser, München</li> <li>• Sedgewick, R.: Algorithms, Addison-Wesley 1984</li> </ul> <p>- Weitere Literaturhinweise können zu Beginn des Semesters bekanntgegeben werden.</p>
<b>Modulkoordination</b>	Prof. Dr. Jörg Schäfer
<b>Lehrende</b>	Prof. Dr. Justus Klingemann, Prof. Dr. Jens Liebehenschel, Prof. Dr. Jörg Schäfer, Prof. Dr. Barış Sertkaya
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Module Einführung in die Programmierung mit C, Praxisorientierte Einführung in die Informatik, Mathematische Grundlagen – Algebra

<b>Modulnummer: 10a</b>		<b>Recht und Datenschutz</b>	
<b>Modultitel (englisch)</b>		Legal aspects and data protection	
<b>Dauer des Moduls</b> Ein Semester		<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch
<b>Art des Moduls</b> Wahlpflichtmodul		<b>Häufigkeit des Angebots</b> Jedes Sommersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden
<b>Verwendbarkeit des Moduls</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine	
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls können die Studierenden <ul style="list-style-type: none"><li>die grundlegenden Rechtsbegriffe des Zivilrechts (Vertragsabschluss, AGB, Urheberrecht) benennen, erläutern und gegenüberstellen</li><li>Datenschutzrecht auf vertiefter Ebene skizzieren, analysieren und deuten</li><li>Juristische Fallgestaltungen strukturiert lösen</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Vorlesung Recht und Datenschutz		Vorlesung	2 SWS
b) Übung Recht und Datenschutz		Übung	2 SWS
<b>Empfohlenes Fachsemester</b>		2. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 4 SWS / 60 Stunden		<b>Selbststudium (h)</b> 90	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>		a) Das Modul hat die folgenden beiden Zielrichtungen: <ul style="list-style-type: none"><li>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.</li><li>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.</li></ul> Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden. <ul style="list-style-type: none"><li>Grundlagen Recht<ol style="list-style-type: none"><li>Vertragsgestaltung</li><li>Allgemeine Geschäftsbedingungen</li><li>Gewährleistungs- und Haftungsansprüche</li><li>Schnittstellen zum Urheberrecht</li></ol></li><li>Grundlagen Datenschutz<ol style="list-style-type: none"><li>Begriffe und Grundsätze des Datenschutzes</li><li>Rechte der Betroffenen</li><li>Pflichten der Verantwortlichen</li><li>Datenschutz im internationalen Bereich</li></ol></li><li>Schnittstelle IT-Sicherheit<ol style="list-style-type: none"><li>Technische und organisatorische Maßnahmen</li><li>Risikobasierter Ansatz</li></ol></li></ul> b) 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.	
<b>Literatur</b>		<ul style="list-style-type: none"><li>Albrecht/Jotzo: Das neue Datenschutzrecht der EU, Nomos, 2017</li><li>Brox/Walker: Allgemeiner Teil des BGB, 42. Auflage, Verlag Franz Vahlen, 2018</li></ul>	

	<ul style="list-style-type: none"> <li>• Wedde: EU-Datenschutz-Grundverordnung, Bund-Verlag, Frankfurt, 2016</li> </ul> Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben
<b>Modulkoordination</b>	Prof. Dr. Ingo Ritter
<b>Lehrende</b>	Prof. Dr. Ingo Ritter
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Vorausgesetzt werden die zum Erwerb der Fachhochschulreife erwarteten Kenntnisse im Bereich Recht, Politik oder Gesellschaftslehre.

<b>Modulnummer: 10b</b>		<b>Elektrotechnik</b>		
<b>Modultitel (englisch)</b>		Electrical engineering		
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Wahlpflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Sommersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Mechatronik (B.Eng.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Versuche im Labor mit schriftlicher Ausarbeitung zu jedem Versuch, (Gesamtaufwand 9 Stunden)  bestanden/nicht bestanden		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Die Studierenden können die inhaltlichen Grundlagen der Gleich- und Wechselstromtechnik wiedergeben und erläutern. Sie sind in der Lage elektrische Schaltungen zu entwerfen und zu dimensionieren.		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Elektrotechnik		Seminaristischer Unterricht mit integrierter Übung		4 SWS
b) Labor Elektrotechnik		Labor		1 SWS
<b>Empfohlenes Fachsemester</b>		2. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) Gleichstromkreis, elektrisches Feld, elektrisches Strömungsfeld, elektrostatisches Feld, magnetisches Feld, Wechselstromtechnik b) Laborversuche zu den Inhalten der Unit 1		
<b>Literatur</b>		•Frohne H et al: Moeller Grundlagen der ET •Teubner Verlag Stuttgart 2002 Lunze K: •Einführung in die ET •Hüthig Verlag Heidelberg 1968 Grafe H, Loose J •Kühn H: Grundlagen der ET Band 1 – •Gleichspannungstechnik und Band 2 – •Wechselspannungstechnik •Hüthig Verlag Heidelberg 1972 und 1973 Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben		
<b>Modulkoordination</b>		N.N		
<b>Lehrende</b>		n.n		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)		a) Keine b) Zur Teilnahme am Labor ist die rechtzeitige Anmeldung erforderlich.		

<b>Module number: 11</b>		<b>Software Engineering – Analysis</b>	
<b>Module title (English)</b>		Software Engineering – Analysis	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every winter semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Computer-based project with documentation (submission period 9 weeks) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• identify and explain different models of the software process and to analyze software requirements</li><li>• outline and apply agile project management methods</li><li>• assess the applicability of software engineering methods in an application development context</li><li>• explain the roles of software developers and project managers</li><li>• demonstrate basic proficiency in the software engineering of large software systems and object-oriented software analysis</li><li>• cooperate and communicate in project-based teamwork</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Software Engineering – Analysis		Interactive lecture	2 SWS
b) Exercise Software Engineering – Analysis		Working in small groups	2 SWS
c) Introduction to scientific work		Übung	1 SWS
<b>Recommended semester of study</b> Semester 3			
<b>SCH / contact hours of the module</b> 5 SCH/ 75 h		<b>Independent study (h)</b> 75	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• the software product</li><li>• different software process models</li><li>• object-oriented software analysis concepts</li><li>• different UML diagrams</li><li>• requirements engineering</li></ul> planning and scheduling, and metrics b) Exercises relating to topics of lectures (see unit lectures) c) Scientific work is the basis for written elaborations and presentations. This second module focuses on the application for written elaborations and presentations of project results. <ul style="list-style-type: none"><li>• Researching, editing, analyzing, and presenting technical literature.</li><li>• Finding an appropriate research question</li><li>• Documenting work steps</li><li>• Collecting, processing, analyzing and presenting data.</li></ul>	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Software Engineering: A Practitioner's Approach; Roger S. Pressman and Bruce Maxim, 8th Edition, 2014</li><li>• Mastering the Requirements Process: Getting Requirements Right; Suzanne Robertson and James Robertson, 3rd Edition, 2012</li></ul> - Further recommended reading will be announced in the first lectures	
<b>Module coordination</b>		Prof. Dr. Matthias Wagner	
<b>Teachers</b>		Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner	
<b>Notes</b> (In particular, recommended prior knowledge)		Einführung in die Programmierung mit C; Englisch; Objektorientierte Programmierung Grundlagen und Konzepte -- C++	

Module number: 12		Embedded Systems	
Module title (English)		Embedded Systems	
Module duration One semester		Language(s) of instruction for this module English	Language(s) of examination for this module English
Module type Compulsory module		Availability Every winter semester	ECTS points (CP) / workload (h) 5 CP / 150 h
Module relevance		Informatik- Mobile Anwendungen (B.Sc.)	
Admission requirement(s) for the module and module examination		Module 8: Microcomputer Technik	
Requirements for awarding ECTS points:			
A) Prerequisite assessments for admission to the module examination (type, workload, duration and grading scheme)		Written laboratory report for every laboratory task (processing time 80 hours)  pass/fail	
B) Final module assessment (type, workload, duration and grading scheme)		Written computer-based examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
Learning outcomes / skills		Upon successful completion of the module the students <ul style="list-style-type: none"><li>• understand the architecture of Embedded Systems,</li><li>• 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,</li><li>• understand real time scheduling.</li><li>• know how to optimize embedded programs in terms of size and power consumption.</li><li>• model and implement embedded systems and communication with external devices.</li><li>• organize themselves in terms of content and time within a project context,</li><li>• solve problems in a structured mannner,</li><li>• apply technical English to a project work context</li></ul>	
Courses, learning and teaching formats, and SCH of the course:			
Course title		Learning and teaching formats	Course SCH
a) Lecture Embedded Systems		Lecture	2 SWS
b) Laboratory Embedded Systems		Several projects	2 SWS
Recommended semester of study Semester 3			
SCH / contact hours of the module 4 SCH/ 60 h		Independent study (h) 90	Practice hours (h) 0
Module content		a) <ul style="list-style-type: none"><li>• Architecture of Embedded Systems;</li><li>• 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;</li><li>• Special Function Register for programming analog-/ digital converters, timer, serial interfaces and pulsewidth modulation;</li><li>• programming of embedded intelligent systems in a problemoriented language;</li><li>• intelligent algorithms such as pattern recognition and their application for Embedded Intelligent Systems such as intelligent sensors and robots.</li></ul> b) Projects regarding acquisition, processing and display of analog signals with embedded systems, Programming of timers and actor control..	
Literature		<ul style="list-style-type: none"><li>• Nauth: Embedded Intelligent Systems, Oldenbourg Verlag, 2005</li><li>• Schiffmann/Schmitz: Technische Informatik Bd. 1 und 2, Berlin, Springer 2001</li><li>• Berns/Schürmann/Trapp: Eingebettete Systeme, Vieweg+Teubner, 2010</li><li>- Current literature will be announced at the beginning of the semester</li></ul>	
Module coordination		Prof. Dr. Matthias Deegener	
Teachers		Prof. Dr. Matthias Deegener, Prof. Dr. Jens Liebehenschel, Prof. Dr. Peter Tawdross	
Notes (In particular, recommended prior knowledge)			

<b>Module number: 13</b>		<b>Object-oriented Programming in Java – Advanced Course</b>	
<b>Module title (English)</b>		Object-oriented Programming in Java – Advanced Course	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every winter semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		Partial examination of “Einführung in die Programmierung mit C” of module 4: Einführung in die Programmierung mit C und Ob-jektorientierte Programmierung Grundlagen	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		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 (processing time 80 hours)  pass/fail	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Computer-based project with documentation (submission period 9 weeks) with presentation (at least 15, at most 30 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• demonstrate sound knowledge of advanced concepts in object-oriented programming</li><li>• apply advanced techniques of state-of-the-art object-oriented programming paradigms in order to produce an application program of moderate complexity</li><li>• structure and generate technical texts in English</li><li>• organize himself/herself as member of a team in a project context</li><li>• plan and realize an application within a given timeframe</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Object-oriented Programming in Java		Lecture	2 SWS
b) Exercise Object-oriented Programming in Java		Exercise	3 SWS
<b>Recommended semester of study</b>		Semester 3	
<b>SCH / contact hours of the module</b> 5 SCH/ 75 h		<b>Independent study (h)</b> 75	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) • Basic and advanced concepts of object-oriented programming based on the Java programming language <ul style="list-style-type: none"><li>• Introduction to the Java Development Kit and Java Application Programming Interface</li><li>• Platform-independent specification</li></ul> Design and implementation of a Java application which comprises a graphical user interface and is based on API packages b) 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..	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Gosling, James et al. The Java Language Specification. (see <a href="https://docs.oracle.com/javase/specs/jls/se10/jls10.pdf">https://docs.oracle.com/javase/specs/jls/se10/jls10.pdf</a>)</li><li>• C. Horstmann. Core Java Volume I---Fundamentals. Prentice Hall, Boston, 10 edition, (2016)</li><li>• Oracle Corp.(Hrsg.). JDK 10 Documentation. <a href="https://docs.oracle.com/javase/10/">https://docs.oracle.com/javase/10/</a></li></ul> Additional literature may be announced at the beginning of the course	
<b>Module coordination</b>		Prof. Dr. Ute Bauer-Wersing	
<b>Teachers</b>		Prof. Dr. Ute Bauer-Wersing , Dr. Maike Kamlage, Prof. Dr. Doina Logofătu	
<b>Notes</b> (In particular, recommended prior knowledge)		Successful participation in module M4 "Einführung in die Programmierung mit C" and "Objektorientierte Programmierung Grundlagen" b) Self-study time is equivalent to exam preparation time since the examination is via project work	

<b>Module number: 14a</b>		<b>Databases</b>	
<b>Module title (English)</b>		Databases	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every winter semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (120 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• outline the relational data model and apply it practically</li><li>• master the standard database language SQL by using a specific database management system</li><li>• solve a given problem in a structured manner and develop their creative skills.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Databases		Lecture	4 SWS
b) Exercise Databases		Exercise	2 SWS
<b>Recommended semester of study</b>	Semester 3 (für Studierende der Vertiefungsrichtung Mobile Anwendungen)		
<b>SCH / contact hours of the module</b> 6 SCH/ 90 h		<b>Independent study (h)</b> 60	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) <ul style="list-style-type: none"><li>• Introduction: database system concepts, database architecture, data models</li><li>• The relational data model, relational database constraints, relational algebra</li><li>• Data modeling and relational database schema: Entity-Relationship model, ER-to-relational mapping, normalization</li><li>• 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).</li><li>• Introduction to SQL Programming techniques: database functions, procedures, trigger, interfaces</li></ul> Transaction processing concepts, user administration, data dictionary views b) 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.	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Elmasri, R. and S. Navathe. Fundamentals of Database Systems, Addison Wesley Publishing Company</li><li>• Date, C.J., An Introduction to Database Systems. AddisonWesley</li><li>• Garcia-Molina, H., J. D. Ullman and J. D. Widom, Database Systems: The Complete Book, Prentice Hall.</li><li>• Härder, Theo; Rahm, Erhard: Datenbanksysteme - Konzepte und Techniken der Implementierung, Springer</li><li>• Kemper, A. und A. Eickler. Datenbanksysteme, Oldenbourg.</li><li>• Kifer, M., A. Bernstein and P.M. Lewis. Database Systems: An Application-Oriented Approach. Pearson International</li><li>• Ramakrishnan, R. and J. Gehrke, Database Management Systems, McGraw-Hill</li><li>• Silberschatz, A., H.F. Korth, S. Sundershan, Database System Concepts, McGraw Hill.</li></ul>	
<b>Module coordination</b>		Prof. Dr. Christian Rich	
<b>Teachers</b>		Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich	



<b>Notes</b> (In particular, recommended prior 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.
--------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Modulnummer: 14b</b>		<b>Elektronik</b>		
<b>Modultitel (englisch)</b>		Electronics		
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Wahlpflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Mechatronik (B.Eng.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Versuche im Labor mit schriftlicher Ausarbeitung zu jedem Versuch, (Gesamtaufwand 15 Stunden)  bestanden/nicht bestanden		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Die Studierenden können die Funktionsweise elektronischer Bauelemente und ihre Beschreibung in Simulationsprogrammen (SPICE) darstellen und die Wirkungsweise von analogen und digitalen Schaltkreisen auf vertiefter Ebene skizzieren Die Studierenden sind in der Lage elektronischer Schaltungen zu entwerfen und zu dimensionieren. Sie können Schnittstellenprobleme zwischen digitalen und analogen Schaltkreisen erkennen und analysieren. Simulationswerkzeugen können sie einsetzen. Die Studierenden können Techniken wissenschaftlichen Arbeitens anwenden.		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Elektronik		Seminaristischer Unterricht mit integrierter Übung		4 SWS
b) Labor Elektronik		Labor		1 SWS
<b>Empfohlenes Fachsemester</b>		3. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75		<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>		a) Aufbau und Wirkungsweise von Halbleiterdioden, Bipolar- und Feldeffekttransistoren. Grundschaltungen von Kleinsignalverstärkern. Differenzverstärker und Funktionsprinzipien der integrierten Schaltungstechnik. Operationsverstärker und seine Grundschaltungen. Grundschaltungen der digitalen Schaltungstechnik. Digital/Analog und Analog/Digital-Umsetzer. Simulation von digitalen / analogen Schaltkreisen b) Laborversuche zu den Inhalten der Unit 1		
<b>Literatur</b>		Tietze; Schenk: Halbleiterschaltungstechnik, Springer-Verlag Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben		
<b>Modulkoordination</b>		Prof. Dr. Hollstein		
<b>Lehrende</b>		Prof. Dr. Hollstein		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)		a) Erfolgreich abgeschlossenes Modul 10b Elektrotechnik b) Zur Teilnahme am Labor ist die rechtzeitige Anmeldung erforderlich. Näheres wird durch Aushang geregelt.		

<b>Module number: 15</b>		<b>Computer Networks</b>	
<b>Module title (English)</b>		Computer Networks	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every winter semester	<b>ECTS points (CP) / workload (h)</b> 10 CP / 300 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• describe and explain fundamental concepts of computer systems and their interconnection via computer networks</li><li>• outline basic concepts of communication protocols and their use in computer networks</li><li>• produce technical texts in English</li><li>• solve a given problem in a structured manner and develop their creative skills.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Computer Networks		Lecture	2 SWS
b) Exercise Computer Networks		Exercise	2 SWS
<b>Recommended semester of study</b> Semester 3			
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Introduction to Computer Networks</li><li>• Data transmission</li><li>• OSI - Reference Model</li><li>• Local Networks</li><li>• LAN - Extensions</li><li>• Internet</li><li>• Network management</li><li>• Routing, Bridging, Switching</li><li>• Protocols at all layers of the reference model</li><li>• IEEE 802 family protocols</li></ul> ARP, IPv4, IPv6, TCP, HTTP, DNS, etc. b) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Introduction to Computer Networks</li><li>• Data transmission</li><li>• OSI - Reference Model</li><li>• Local Networks</li><li>• LAN - Extensions</li><li>• Internet</li><li>• Network management</li><li>• Routing, Bridging, Switching</li><li>• Protocols at all layers of the reference model</li><li>• IEEE 802 family protocols</li></ul> ARP, IPv4, IPv6, TCP, HTTP, DNS, etc..	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Christian Baun, Computernetze kompakt, Springer Vieweg. 2018.</li><li>• Andrew S. Tanenbaum and David J. Wetherall, Computer Networks, Pearson Education 2013.</li><li>• James F. Kurose and Keith D. Ross, Computer Networking: A Top-Down Approach, Pearson Education, 2016.</li></ul> Further recommended reading will be announced in the first lectures	
<b>Module coordination</b>		Prof. Dr. Christian Baun	
<b>Teachers</b>		Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes	

<b>Notes</b> (In particular, recommended prior knowledge)	Successful completion of modules of semesters 1-2 particularly Theoretische Informatik, Algorithmen und Datenstrukturen, Rechnerarchitektur, Einführung in die Programmierung mit C. The students also learn project- and teamwork as they work in teams.
--------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Module number: 16</b>		<b>Operating Systems</b>	
<b>Module title (English)</b>		Operating Systems	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		Passed examination module 9: Algorithmen und Datenstrukturen	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• name and describe the fundamental concepts of computer systems, especially the task of operating systems</li><li>• outline basic concepts and methods for implementation of operating systems</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Operating Systems		Lecture	2 SWS
b) Exercise Operating Systems		Exercise	2 SWS
<b>Recommended semester of study</b>		Semester 3	
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Processes and process management</li><li>• Memory management</li><li>• File systems</li><li>• Input/Output devices</li><li>• Distributed operating systems</li><li>• Windows and Unix based operating systems</li></ul> System management and administration b) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Processes and process management</li><li>• Memory management</li><li>• File systems</li><li>• Input/Output devices</li><li>• Distributed operating systems</li><li>• Windows and Unix based operating systems</li></ul> System management and administration.	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Christian Baun, Betriebssysteme kompakt, Springer Vieweg. 2017.</li><li>• William Stallings, Operating Systems: Internals and Design Principles, Pearson, 2017.</li><li>• Andrew Tanenbaum, Modern Operating Systems, Pearson, 2014.</li></ul> Further recommended reading will be announced in the first lectures.	
<b>Module coordination</b>		Prof. Dr. Christian Baun	
<b>Teachers</b>		Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes	
<b>Notes</b> (In particular, recommended prior knowledge)		Successful completion of modules of semesters 1-2, particularly Theoretische Informatik, Algorithmen und Datenstrukturen, Rechnerarchitektur, Einführung in die Programmierung mit C  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.	

<b>Module number: 17</b>		<b>Software Engineering – Design</b>	
<b>Module title (English)</b>		Software Engineering – Design	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		Computer-based exercises with written documentation (processing time 36 hours)  pass/fail	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (120 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• outline and reproduce the basic principles and concepts of software design and implementation</li><li>• critically assess and estimate the usage of the various methods of software design in the application development context</li><li>• classify and illustrate the roles of software developers and project managers</li><li>• demonstrate enhanced proficiency in the software engineering of large software systems</li><li>• employ methods of project management</li><li>• use IDE and CASE tools</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Software Engineering – Design		Interactive lecture	2 SWS
b) Exercise Software Engineering – Design		Working in small groups	2 SWS
<b>Recommended semester of study</b>		Semester 4	
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• the software design process</li><li>• object-oriented design principles and concepts</li><li>• software architecture</li><li>• software testing</li><li>• design patterns</li></ul> different UML diagrams b) Students gain experience and deeper understanding of lecture's topics, and working in teams to solve specific exercises.	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Software Engineering: A Practitioner's Approach; Roger S. Pressman and Bruce Maxim, 8th Edition, 2014</li><li>• Mastering the Requirements Process: Getting Requirements Right; Suzanne Robertson and James Robertson, 3rd Edition, 2012</li></ul> Further recommended reading will be announced in the first lectures	
<b>Module coordination</b>		Prof. Dr. Eicke Godehardt	
<b>Teachers</b>		Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner	
<b>Notes</b> (In particular, recommended prior knowledge)			

<b>Module number: 18a</b>		<b>Mobile Devices</b>	
<b>Module title (English)</b>		Mobile Devices	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Elective module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Project work software development (submission period 12 weeks) with presentation (at least 10, at most 20 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module students are able to <ul style="list-style-type: none"><li>• generate programs for mobile computer systems (e.g. smartphones), which require the involvement of additional requirements,</li><li>• judge the time-variant behaviour of mobile systems and, therefore cope with real-time scheduling and the integration of sensors and actors,</li><li>• design mobile systems and implement them as executable programs which includes the utilization of cross-building toolchains and development kits, e.g. the Android Software Development Kit,</li><li>• optimize mobile applications in terms of power consumption and memory usage.</li><li>• read, understand, and present scientific work.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Mobile Devices		Lecture	2 SWS
b) Laboratory Mobile Devices		Laboratory	2 SWS
<b>Recommended semester of study</b> Semester 4			
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) Introduction into the programming of mobile computer systems (e.g. smartphones) exploiting the involvement of additional requirements such as <ul style="list-style-type: none"><li>• modelling of mobile systems,</li><li>• integration of sensors and actors,</li><li>• design of user interfaces for mobile devices,</li><li>• data exchange and data storage on mobile devices,</li><li>• time variant behaviour, real-time scheduling and concurrency in mobile systems,</li><li>• implementation and deployment of mobile applications as executable programs,</li><li>• utilization of cross-building toolchains and development kits like the Android Software Development Kit.</li></ul> b) <ul style="list-style-type: none"><li>• Programming of mobile devices</li><li>• Debugging of microcontrollers</li><li>• implementation of mobile applications as executable programs.</li><li>• utilization of cross-building toolchains and development kits e.g. Android Development Kit.</li></ul>	
<b>Literature</b>		Specific references, also covering current trends and developments, will be published at the beginning of the lectures. Contemporary references include: <ul style="list-style-type: none"><li>• Mawlood-Yunis: Android for Java Programmers, Springer, 2022</li><li>• Kopetz, Steiner: Real-Time Systems - Design Principles for Distributed Embedded Applications. Springer, 2022</li><li>• Künne: Android 11: Das Praxisbuch für App-Entwickler, RheinWerk Computing, 2020</li><li>• various online resources, such as documentation to established development kits and frameworks</li></ul>	
<b>Module coordination</b>		Prof. Dr. Thomas Gabel	
<b>Teachers</b>		Prof. Dr. Thomas Gabel, Prof. Dr. Jens Liebehenschel	
<b>Notes</b> (In particular, recommended prior knowledge)			

<b>Module number: 18b</b>		<b>Computer Architecture</b>	
<b>Module title (English)</b>		Computer Architecture	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Elective module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>			
<b>Admission requirement(s) for the module and module examination</b>		Module 5 Digitaltechnik, Module 4 Einführung in die Programmierung (Introduction to Programming) und Objektorientierte Programmierung (Object-oriented Programming).	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		Laboratory exercises with written report (processing time 60 hours)  pass/fail	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module students are able to : <ul style="list-style-type: none"><li>• design digital circuits using programmable logic devices and Field Programmable Gate-Ar-rays (FPGAs);</li><li>• design sequential/clocked digital circuits on Register Transfer Level (RTL);</li><li>• name and explain the major relevant components of microprocessors and arithmetic-logic units, adder and multiplier architectures, registers, controllers</li><li>• explain modern processor architectures (Single Cycle, Multi-Cycle, Pipelined)</li><li>• evaluate the performance of different processor architectures</li><li>• use IDE tools for digital circuit design</li><li>• solve a given circuit design problem in a structured manner</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Computer Architecture: Lecture with integrated Exercises		Lecture: Seminar Style with integrated Exercises	3 SWS
b) Computer Architecture: Lab		CAD Design Lab	1 SWS
<b>Recommended semester of study</b>		Semester 4	
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) • Digital Circuit Design based on Programmable Logic und Field-Programmable Gate-Ar-rays (FPGA) <ul style="list-style-type: none"><li>• Sequential (e.g. clocked) Digital Circuits</li><li>• Microprozessor Components: Arithmetic-Logic Units, Architekturen für Arithmetic Components, Registers, Controllers</li><li>• Prozessor Architectures: Single Cycle, Multi-Cycle, Pipelined</li><li>• Memory Components</li></ul> b) • Digital Circuit Design based on programmable Logic and Field-Programmable Gate-Ar-rays (FPGA) <ul style="list-style-type: none"><li>• - Circuit Modeling based on Hardware Description Languages (HDL) (e.g. SystemVerilog, VHDL, SystemC).</li></ul>	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Harris, Sarah L. und Harris, David: Digital Design and Computer Architecture, ARM Edition, Morgan Kaufmann;</li><li>• Hennessy, John L. und Patterson, David: Computer Architecture: A Quantitative Approach, Morgan Kaufmann;</li></ul>	
<b>Module coordination</b>		Prof. Dr.-Ing. Thomas Hollstein	
<b>Teachers</b>		Prof. Dr.-Ing. Thomas Hollstein	
<b>Notes</b> (In particular, recommended prior knowledge)		Modules Programming, “Digitaltechnik (Logic Design)” or equivalent knowledge in the fields of Logic Design and Boolean Algebra, Electronics a) Use of English language literature b) The laboratory exercises consist of 5 partial lab attestations, which have to be passed completely: Lab designs have to be presented and discussed with the lab supervisor . Use of English language literature and Databooks and CAD tools	

<b>Module number: 19</b>	<b>IT-Security</b>
--------------------------	--------------------



Module title (English)		IT-Security	
Module duration One semester		Language(s) of instruction for this module English	Language(s) of examination for this module English
Module type Compulsory module		Availability Every summer semester	ECTS points (CP) / workload (h) 5 CP / 150 h
Module relevance		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
Admission requirement(s) for the module and module examination		None	
Requirements for awarding ECTS points:			
A) Prerequisite assessments for admission to the module examination (type, workload, duration and grading scheme)		None	
B) Final module assessment (type, workload, duration and grading scheme)		Written examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
Learning outcomes / skills		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• name and describe fundamental concepts of IT Security</li><li>• detect and interpret IT Security aims and risks and analyze security mechanisms and their applicability with respect to exemplary scenarios</li><li>• develop basic solutions, concepts and methods to implement IT Security and assess security risks in simple scenarios.</li></ul>	
Courses, learning and teaching formats, and SCH of the course:			
Course title		Learning and teaching formats	Course SCH
a) Lecture IT-Security		Lecture	2 SWS
b) Exercise IT-Security		Exercise	2 SWS
Recommended semester of study Semester 4			
SCH / contact hours of the module 4 SCH/ 60 h		Independent study (h) 90	Practice hours (h) 0
Module content		a) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Cryptographical principles and methods</li><li>• Authentication</li><li>• Operating system security</li><li>• Application security</li><li>• Malware</li><li>• Network security</li><li>• Firewalls</li><li>• Virtual Private Networks</li><li>• Network surveillance</li><li>• Availability</li><li>• Network applications</li><li>• Security of realtime communications</li><li>• Local network security</li><li>• Standards</li><li>• Practical implications</li></ul> b) Selection from areas such as, but not limited to: <ul style="list-style-type: none"><li>• Cryptographical principles and methods</li><li>• Authentication</li><li>• Operating system security</li><li>• Application security</li><li>• Malware</li><li>• Network security</li><li>• Firewalls</li><li>• Virtual Private Networks</li><li>• Network surveillance</li><li>• Availability</li><li>• Network applications</li><li>• Security of realtime communications</li><li>• Local network security</li><li>• Standards</li><li>• Practical implications.</li></ul>	
Literature		<ul style="list-style-type: none"><li>• Martin Kappes, Netzwerk- und Datensicherheit, Springer, Wiesbaden, 2. Auflage, 2013.</li><li>• Claudia Eckert, IT-Sicherheit: Konzepte, Verfahren, Protokolle, Oldenbourg-Verlag,</li></ul>	

	München, 10. Auflage, 2018. <ul style="list-style-type: none"> <li>• Original NIST, IETF, IEEE and ITU Standards.</li> <li>• Recommended reading will be announced in the first lectures.</li> </ul>
<b>Module coordination</b>	Prof. Dr. Martin Kappes
<b>Teachers</b>	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes
<b>Notes</b> (In particular, recommended prior knowledge)	Successful completion of all modules of semesters 1- 3

<b>Module number: 20</b>		<b>Distributed Systems</b>	
<b>Module title (English)</b>		Distributed Systems	
<b>Module duration</b> One semester	<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English	
<b>Module type</b> Compulsory module	<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h	
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		Module 13 “Object-oriented Programming in Java – Advanced Course”	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		Laboratory Exercises (processing time 30 hours)  pass/fail	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (90 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• Realize distributed applications by using different technologies common in the industry</li><li>• Implement practical examples in programming</li><li>• Install and configurate software components</li><li>• Use frameworks and middleware tools in own programs</li><li>• assess different technologies and decide upon their bene-fits in concrete application contexts in order to being able to design suitable applications themselves</li><li>• solve problems by developing distributed applications on the basis of a sound theoretical foundation.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Distributed Systems		Lecture	4 SWS
b) Exercise Distributed Systems		Exercise	2 SWS
<b>Recommended semester of study</b> Semester 4			
<b>SCH / contact hours of the module</b> 6 SCH/ 90 h		<b>Independent study (h)</b> 60	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) 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: <ul style="list-style-type: none"><li>• Sockets as a base technology for distributed applications</li><li>• Object-oriented Middleware-Technologies</li><li>• Message Queues</li><li>• Web-based distributed systems</li><li>• Web Services</li><li>• REST</li><li>• Techniques for integrating databases</li></ul> If suitable, additional relevant topics shall be covered. b) <ul style="list-style-type: none"><li>• Programming exercises based on technologies taught in lecture</li><li>• Installation and configuration of software components</li><li>• Creation of distributed applications using technologies taught in lecture</li><li>• Individual feedback to students.</li></ul>	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Günther Bengel: Grundkurs Verteilte Systeme, Springer Vieweg</li><li>• Dietmar Abts: Masterkurs Client/Server-Programmierung mit Java, Springer Vieweg</li><li>• Rainer Oechsle: Parallele und verteilte Anwendungen in Java, Hanser</li><li>• Peter Mandl: Masterkurs Verteilte betriebliche Informationssysteme, Vieweg + Teubner</li><li>• Frank Müller-Hofmann, Martin Hiller, Gerhard Wanner: Programmierung von verteilten Systemen und Webanwendungen mit Java EE, Springer Vieweg</li><li>• Steffen Heinzl, Markus Mathes: Middleware in Java, Vieweg</li><li>• G. Coulouris, J. Dollimore, T. Kindberg: Distributed Systems: Concepts and Design, Addison-Wesley</li><li>• A. Tanenbaum, M. van Steen: Distributed Systems: Principles and Paradigms, Prentice-Hall / Pearson</li><li>- Additional literature may be announced at the beginning of the course.</li></ul>	

<b>Module coordination</b>	Prof. Dr. Justus Klingemann
<b>Teachers</b>	Prof. Dr. Christian Baun, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Karsten Weronek
<b>Notes</b> (In particular, recommended prior knowledge)	Besides the module "Object-Oriented Programming in Java - Advanced Course" knowledge from the modules "Databases" and "Software Engineering – Analysis" is advisable.

<b>Module number: 21</b>		<b>Practical Computer Networks and Applications</b>	
<b>Module title (English)</b>		Practical Computer Networks and Applications	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Compulsory module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Computer-based project with documentation (submission period 9 weeks, processing time 60 hours) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• set up computer networks and computer network applications</li><li>• apply network surveillance technologies for debugging, per-formance analysis and problem mitigation</li><li>• distinguish and outline recent communication paradigms, such as, but not limited to, , e.g. Peer-to-Peer, Cloud Computing, Edge Computing, Fog Computing.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Practical Computer Networks and Applications		Lecture	2 SWS
b) Laboratory Practical Computer Networks and Applications		Laboratory	2 SWS
<b>Recommended semester of study</b>		Semester 4	
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) • 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 b) • setting up of computer networks and computer network applications. • application of network surveillance technologies for debugging, performance analysis and problem mitigation.	
<b>Literature</b>		flexible	
<b>Module coordination</b>		Prof. Dr. Christian Baun	
<b>Teachers</b>		Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes	
<b>Notes</b> (In particular, recommended prior knowledge)		Successful completion of modules of semesters 1-3, particularly Computer Networks	

<b>Module number: 22a</b>		<b>Mobile Application Exercises</b>	
<b>Module title (English)</b>		Mobile Application Exercises	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Elective module		<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		Partial examination of “Einführung in die Programmierung mit C” of module 4: Einführung in die Programmierung mit C und Objektorientierte Programmierung Grundlagen	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Project work software development (submission period 12 weeks) with presentation (at least 10, at most 20 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module students are able to <ul style="list-style-type: none"><li>• design, implement, and test software and/or hardware for Mobile and/or Embedded Systems based on given requirements</li><li>• program in a problem-oriented language in a modern programming environment</li><li>• outline the architecture of Mobile and/or Embedded Systems</li><li>• schedule Mobile and/or Embedded Systems</li><li>• communicate with internal and/or external units like sensors and/or data bases</li><li>• Cooperate and communicate in a project context</li><li>• solve problems in a structured manner</li><li>• present work results</li><li>• apply advanced technical English skills</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Mobile Application Exercises		Project	4 SWS
<b>Recommended semester of study</b> Semester 4			
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h		<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>		The knowledge acquired in the areas of programming, software engineering, mobile devices and distributed systems is utilized within a realistic programming environment. This includes, in particular, the complete development and implementation of a mobile application (design, implementation, test, presentation)..	
<b>Literature</b>		Readings depend on the actual topic and will be announced in the first lectures.	
<b>Module coordination</b>		Prof. Dr. Jens Liebehenschel	
<b>Teachers</b>		Prof. Dr. Jens Liebehenschel, Prof. Dr. Thomas Gabel, Prof. Dr. Matthias Deegener	
<b>Notes</b> (In particular, recommended prior knowledge)			

<b>Module number: 22b</b>	<b>Embedded Systems Project</b>	
<b>Module title (English)</b>	Embedded Systems Project	
<b>Module duration</b> One semester	<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Elective module	<b>Availability</b> Every summer semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		
<b>Admission requirement(s) for the module and module examination</b>	Module 8 “Mikrocomputer Technik”	
<b>Requirements for awarding ECTS points:</b>		
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )	None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )	Computer-based project with documentation (submission period 12 weeks) with presentation (at least 15, at most 30 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>	Upon completion of the module students will be able to <ul style="list-style-type: none"><li>• optimize their technical skills in programming,, documentation,, software engineering, presentation and communication by working on a complex software project in the field of embedded systems,</li><li>• solve tasks from one or more areas of embedded systems</li><li>• achieve a project goal within a given timeframe</li><li>• organize as a team and as a member of a team,</li><li>• communicate with others at a high technical level and overcome unexpected difficulties (both technical and social),</li><li>• practicing tolerance towards project partners,</li><li>• take responsibility in a project context.</li><li>• contribute social responsibility to the implementation of project goals.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>		
<b>Course title</b>	<b>Learning and teaching formats</b>	<b>Course SCH</b>
Embedded Systems Project	Project	4 SWS
<b>Recommended semester of study</b>	Semester 4	
<b>SCH / contact hours of the module</b> 4 SCH/ 60 h	<b>Independent study (h)</b> 90	<b>Practice hours (h)</b> 0
<b>Module content</b>	A topic from the field of embedded systems..	
<b>Literature</b>	Literature depends on the current project, will be announced at the beginning of the course.	
<b>Module coordination</b>	Prof. Dr. Matthias Deegener	
<b>Teachers</b>	All professors of the Computer Science Smart Systems degree program	
<b>Notes</b> (In particular, recommended prior knowledge)	Passed examination module 12 Embedded Systems and 14b: Elektronik	

<b>Modulnummer: 23a</b>		<b>Serviceorientierte Architekturen</b>		
<b>Modultitel (englisch)</b>				
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Wahlpflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Informatik- Mobile Anwendungen (B.Sc.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Modul 20: Distributed Systems		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine		
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Projektarbeit (Bearbeitungszeit 8 Wochen) mit Präsentation (mindestens 15, höchstens 20 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		<p>Das Modul erweitert die „Verteilten Anwendungen“ um die Möglichkeiten der „Service-Orientierung (SOA)“ und behandelt die wichtigsten Konzepte und Bausteine.</p> <p>Intendierte Lernergebnisse:</p> <p>Die Studierenden verstehen SOA als Management-Konzept im Geschäftsumfeld für Aufbau und Ablauforganisation, gleichfalls als IT-Architekturkonzept, sowohl unternehmensweit als auch Anwendungs-spezifisch und somit als inner- und interbetriebliches Integrationskonzept, sowie als Mittel zur Integration heterogener Technologien, als Mittel zur Erhöhung der Flexibilität von Anwendungen hinsichtlich Funktionalität, Skalierbarkeit, sowie Releasezyklen (Deployment).</p> <p>Sie sind in der Lage, Services in Geschäftsprozessen (BPMN, eEPK) zu identifizieren (Domain-Driven Design, Business Capabilities), diese zu modellieren und im Rahmen der Servicekomposition durch Orchestrierung bzw. Choreographierung, oder als SAGAs zu implementieren. Darüber hinaus beherrschen sie Methoden für die Modellierung und Implementierung von Datenmodellen, Datenflüssen, sowie Datenpersistenz und -konsistenz zwischen und innerhalb von Services.</p> <p>Sie können die Begriffe der SOA-Terminologie und der Enterprisearchitektur erläutern.</p> <p>Sie begreifen Lösungsmuster (-Patterns) für die jeweiligen Standardanforderungen und können einzelne davon umsetzen.</p> <p>Sie sind zur Entscheidungsfindung in der Lage, komplexe technische und organisatorische Probleme kritisch und methodisch zu analysieren und zu strukturieren.</p> <p>Sie sind offen dafür technische, organisationsübergreifende Lösungen auch in interkulturellen Teams zu erarbeiten.</p> <p>Darüber verstehen sie die historische Entwicklung der Ziele und Paradigmen Serviceorientierter Architekturen (WS*) vom Ende des 20. Jahrhunderts bis heute (REST, Microservices).</p>		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Serviceorientierte Architekturen		Vorlesung		2 SWS
b) Übung Serviceorientierte Architekturen		Übung		2 SWS
<b>Empfohlenes Fachsemester</b>		5. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 4 SWS / 60 Stunden		<b>Selbststudium (h)</b> 90	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		<p>a) Nach der Einführung der grundlegenden Begriffe der Enterprise-Architektur (u.a. Architektur, Architekturschichten, Service, Modell, Standard, Pattern) wird zunächst eine betrieblich-organisatorische Perspektive zur Identifikation und Modellierung von Services, die Integration betrieblicher Anwendungssysteme und die Komposition von Services im Sinne einer Orchestrierung (ESB) bzw. Choreographie herangeführt. Dabei werden die jeweiligen Ziele, Architekturprinzipien und Lösungsebenen von klassischen als auch von cloudnativen Webservices diskutiert.</p> <p>Aus einer technischen Perspektive werden anschließend Komponententechnologien (insbesondere klassische und aktuelle Web Services' Technologien) eingeführt und Ausprägungen serviceorientierter Architekturen (im Sinne einer Schnittstellen-, Nachrichten- bzw. Ressourcen-Orientierung) anhand konkreter Softwareprodukte vorgestellt. Betrachtungen zur Granularität von Services, zu Sicherheit, Qualität und Qualitätssicherung, sowie zu aktuellen Trends runden diesen Teil der Veranstaltung ab. Darüber hinaus werden erprobte Vorgehensweisen zu Dokumentation von Services (Service-Vertrag) vermittelt und in den Übungen ausprobiert.</p>		



	<p>In den Übungen werden die in der Vorlesung vermittelten Inhalte anhand ausgewählter Beispiele vertieft.</p> <p>b) Die Übung dienen zur Veranschaulichung, Vertiefung und zur Ergänzung der in der Vorlesung theoretisch vermittelten Themen. Dies erfolgt durch Anwenden, Deployen und Konfigurieren von Services (z.B. OpenAPI/REST, WSDL/SOAP) Datenserialisierung (z.B. XML, JSON, YAML Protocol Buffers) und Datenübertragung (z.B. https, GRPC, NATS, GraphQL) mittels verschiedener Kommunikationsmuster (z.B. request/reply, publish/subscribe, streaming). Teilweise erfolgt dies auf Basis von Virtualisierung (z.B. VirtualBox, Docker/Podman, Kubernetes) Dabei sollte eine aktuelle Programmiersprache eingesetzt werden (z.B. go, python). Darüber hinaus werden gleichzeitig Grundkonzepte moderner Software-Entwicklung vermittelt (z.B. SCRUM, DevOps, Git et.al.). Abgerundet werden die Übungen durch Prozessmodellierung in BPMN und mittels eEPK.</p>
<b>Literatur</b>	<ul style="list-style-type: none"> <li>• Fowler, M., Rice, D., und Foemmel, M. Patterns of Enterprise Application Architecture, Addison-Wesley, München, 2002;</li> <li>• Krafzig, D., Banke, K., und Slama, D. "Enterprise SOA. Service Oriented Architecture Best Practices," Prentice Hall, New Jersey, 2004;</li> <li>• Starke, G., u. Tilkov, S. SOA-Expertenwissen, dpunkt, Heidelberg, 2007;</li> <li>• Arnon Rotem-Al-Goiz, SOA Patterns, Manning, New York, September 2012;</li> <li>• Thomas Erl, Next Generation SOA, PRENTICE HALL, Boston, November 2014;</li> <li>• Eberhard Wolff, Microservices Grundlagen flexibler Softwarearchitekturen, 2. Auflage, DPunkt Verlag, Heidelberg, Juli 2018;</li> <li>• Chris Richardson, Microservices Patterns, Manning, New York, October 2018;</li> <li>• José Haro Peralta, Microservice APIs, Manning, New York, ttp. Januar 2023</li> </ul>
<b>Modulkoordination</b>	Prof. Dr. Karsten Weronek
<b>Lehrende</b>	Prof. Dr. Karsten Weronek
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	<p>Inhaltlich werden die Module der objektorientierte Programmierung, des Software Engineerings, sowie Operating Systems vorausgesetzt.</p> <p>Dieses Modul ist von seiner Breite her äußerst umfangreich und greift daher Grundlagen vieler Module der vorlaufenden Semester auf. Gleichzeitig ist auf Grund der begrenzten Zeit nicht in der Lage, Serviceorientierung umfassend zu vermitteln, insbesondere die Erfahrungen bei der Umsetzung. Es dient daher dazu, die Terminologie und Grundkonzepte, soweit zu vermitteln, wie dies für die Mitarbeit in Softwareentwicklungsteams im SOA-Umfeld erforderlich ist.</p> <p>b) Als ergänzende Themen sind Informationssicherheit (z.B. Authentifizierung und Autorisierung mit OAuth) und ausgewählte Themen verteilter Datenbanken (z.B. etcd, cassandra) möglich</p>

<b>Modulnummer: 23b</b>		<b>Digitale Signalverarbeitung</b>		
<b>Modultitel (englisch)</b>		Digital Signal Processing		
<b>Dauer des Moduls</b> Ein Semester		<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Wahlpflichtmodul		<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>		Elektro- und Informationstechnik (B.Eng.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Algebra, Analysis, Diskrete Mathematik		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Keine  bestanden/nicht bestanden		
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )		Klausur (90 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Die Studierenden besitzen Grundkenntnisse der Digitalen Signalverarbeitung. Sie sind in der Lage DSP-Systeme von der Erfassung der Signale bis zu deren Auswertung zu realisieren; digitale Signale im Zeit- und im Frequenzbereich zu analysieren und zu synthetisieren, im Team mit anderen kooperativ zusammenzuarbeiten; Techniken wissenschaftlichen Arbeitens anzuwenden; technische Inhalte, Problemstellungen und Problemlösungsmöglichkeiten verständlich zu kommunizieren.		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Digitale Signalverarbeitung Vorlesung		Vorlesung		4 SWS
b) Digitale Signalverarbeitung Übung		Übung		1 SWS
<b>Empfohlenes Fachsemester</b>		5. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) • Abtastung analoger Signale und A/D-Wandlung • Diskrete lineare zeitinvariante Systeme • Signalanalyse im Zeitbereich • Diskrete Faltung • Auto- und Kreuzkorrelation • Signalanalyse im Frequenzbereich • z-Transformation • Beurteilung von Systemen anhand ihrer Pole und Nullstellen in der z-Ebene • Programmbeispiele unter Verwendung höherer Programmiersprachen b) Rechenbeispiele aus den Themenbereichen der Digitalen Signalverarbeitung		
<b>Literatur</b>		• Daniel Ch. von Grünigen: Digitale Signalverarbeitung, Fachbuchverlag Leipzig, ISBN 3-446-21445-3 • Emmanuel C. Ifeakor, Barrie W. Jervis: Digital Signal Processing – A practical Approach, Verlag Addison-Wesley, ISBN 0-201-54413-X • Paul A. Lynn, Wolfgang Fuerst: Introductory Digital Signal Processing with Computer Applications, Verlag John Wiley and Sons, ISBN 0-471-97631-8 • Jeweils in der aktuellsten Auflage. Weiterführende Literatur wird in der jeweiligen Lehrveranstaltung bekanntgegeben		
<b>Modulkoordination</b>		Prof. Dr. Manfred Jungke		
<b>Lehrende</b>		Prof. Dr. Manfred Jungke		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)				

<b>Modulnummer: 24a</b>		<b>Human Computer Interaction</b>		
<b>Modultitel (englisch)</b>		Human Computer Interaction		
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch		
<b>Art des Moduls</b> Wahlpflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden		
<b>Verwendbarkeit des Moduls</b>		Informatik- Mobile Anwendungen (B.Sc.)		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Keine		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine		
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Projektarbeit (Bearbeitungszeit 12 Wochen) mit Präsentation (mindestens 15, höchstens 20 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		Nach Abschluss des Moduls haben Studierende ein grundlegendes Verständnis der Mensch-Computer Interaktion. Sie sind in der Lage <ul style="list-style-type: none"><li>• Modelle, Methoden und Konzepte der Mensch-Computer Interaktion ausarbeiten und anwenden. Der Schwerpunkt sind die Benutzerschnittstellen von Mobilgeräten und der menschenzentrierten Entwicklung, die sie präzisieren können.</li><li>• eine technologische Problemlösungskompetenz für die Spezifikation und Implementierung von Benutzerschnittstellen auf Basis theoretischer Grundlagen demonstrieren</li><li>• auf Basis von Benutzeranforderungen ein System mit Methoden der Mensch-Computer Interaktion quantitativ und qualitativ auswerten und die Evaluation wissenschaftlich dokumentieren sowie die gesellschaftlichen Auswirkungen dieser Technologien reflektieren und bewerten.</li></ul>		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Vorlesung Human Computer Interaction		Vorlesung		2 SWS
b) Übung Human Computer Interaction		Übung		2 SWS
c) Wissenschaftliches Arbeiten		Übung		1 SWS
<b>Empfohlenes Fachsemester</b>		5. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 5 SWS / 75 Stunden		<b>Selbststudium (h)</b> 75	<b>Praxiszeiten (h)</b> 0	
<b>Inhalte des Moduls</b>		a) 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 und Evaluation vorgestellt. Nachfolgend sind für die inhaltlichen Schwerpunkte mögliche Themen aufgelistet. Die Schwerpunkte können in unterschiedlicher Tiefe behandelt werden: <ul style="list-style-type: none"><li>• Grundlagen der Mensch-Computer Interaktion</li><li>• Menschzentrierte Entwicklungsprozesse und Modelle der Prototypentwicklung für interaktive Systeme</li><li>• Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen</li><li>• Methoden zur Realisierung und Implementierung von Benutzerschnittstellen</li><li>• Designrichtlinien und Architekturen für die Ein- und Ausgabe interaktiver Systeme, User Interface Toolkits und Schnittstellenkomponenten</li><li>• Kognitive Modelle und multimodale Interaktion mit mobilen Systemen</li></ul> b) Aufgaben und Beispiele zu den Vorlesungsthemen c) Dritte Unit zum wissenschaftlichen Arbeiten. Praktische Umsetzung wissenschaftlicher Projektdokumentation und Präsentation der Arbeit <ul style="list-style-type: none"><li>• Konzeptionieren und Verfassen wissenschaftlicher Artikel (u.a. Methodenberichte von Nutzerstudien, Nutzer-/Zielgruppenanalyse, Aufarbeitung und Systematisierung verwandter Literatur)</li><li>• Rhetorik und Aufbau überzeugender Argumentationen in wissenschaftlichen Arbeiten und Präsentationen</li><li>• Vorbereitung und Halten von Präsentationen (u.a. Themeneingrenzung, Kernbotschaften formulieren, visuelle Gestaltung)</li></ul>		
<b>Literatur</b>		• Preim, B., Dachzelt, R. (2010). Interaktive Systeme: Band 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer-Verlag.		

	<ul style="list-style-type: none"> <li>• Heinecke, A. M. (2011). Mensch-Computer-Interaktion: Basiswissen für Entwickler und Gestalter. Springer-Verlag.</li> <li>• Dix, A., Finlay, J., Abowd, G. D., Beale, R. (2003). Human-Computer Interaction. Pearson Education.</li> <li>• David Benyon, Designing Interactive Systems, Pearson 2010</li> <li>• Lazar, J., Feng, J. H., Hochheiser, H. (2017). Research Methods in Human-Computer Interaction. Morgan Kaufmann.</li> <li>• Field, A., Hole, G. (2002). How to design and report experiments. Sage.</li> </ul> <p>Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben</p>
<b>Modulkoordination</b>	Prof. Dr. Valentin Schwind, Prof. Dr. Matthias Deegener
<b>Lehrende</b>	Prof. Dr. Valentin Schwind, Prof. Dr. Matthias Deegener
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Inhaltlich werden die Module der objektorientierten Programmierung, des Software Engineerings und Embedded Systems vorausgesetzt.

<b>Modulnummer: 24b</b>		<b>FPGA-Schaltungsentwurf</b>	
<b>Modultitel (englisch)</b>		Logic Design with FPGAs	
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Wahlpflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Wintersemester	<b>ECTS-Punkte (CP) / Workload (h)</b> 5 CP / 150 Stunden	
<b>Verwendbarkeit des Moduls</b>			
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		Modul 5 „Digitaltechnik“	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>			
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine	
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		Projektarbeit mit schriftlicher Ausarbeitung (Bearbeitungszeit sechs Wochen) und Präsentation (mindestens 10, höchstens 20 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>		Die Studierenden vertiefen ihre Kenntnisse des Entwurfs digitaler Schaltungen mit FPGAs. Nach Abschluss des Moduls sind die Studierenden in der Lage <ul style="list-style-type: none"><li>• Schaltungen auf Register-Transfer-Ebene für reale Anwendungen zu entwickeln und als synthetisierbaren Code in einer Hardwarebeschreibungssprache zu programmieren für die praktische Realisierung auf FPGA-Boards;</li><li>• den Umgang mit gängigen FPGA-Entwicklungswerkzeugen zu beherrschen und die von ihnen entworfenen Schaltungen mithilfe von Testbenches zu simulieren und zu testen;</li><li>• spezifische FPGA-Funktionsblöcke für den Schaltungsentwurf praktisch einzusetzen;</li><li>• In einem Laborkontext an einem Projekt zu arbeiten;</li><li>• Techniken des wissenschaftlichen Arbeitens anzuwenden.</li></ul>	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>			
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) FPGA-Schaltungsentwurf Vorlesung		Seminaristischer Unterricht	2 SWS
b) FPGA-Schaltungsentwurf Labor		Labor	2 SWS
<b>Empfohlenes Fachsemester</b>		5. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 4 SWS / 60 Stunden		<b>Selbststudium (h)</b> 90	<b>Praxiszeiten (h)</b> 0
<b>Inhalte des Moduls</b>	a) • FPGA-basierter Schaltungsentwurf auf Register-Transfer-Ebene; <ul style="list-style-type: none"><li>• Einführung und Anwendung einer Hardwarebeschreibungssprache, Synthese, Simulation, Testbenches;</li><li>• Umgang mit FPGA-Entwicklungstools;</li><li>• Aufbau und Einsatz von FPGA-Funktionsblöcken;</li><li>• Clock Management</li></ul> b) • Praktische Laborarbeit mit FPGA-Boards; <ul style="list-style-type: none"><li>• Entwurfsaufgaben zu den Themenschwerpunkten der Lehrveranstaltung,</li><li>• Projektarbeit</li></ul>		
<b>Literatur</b>	<ul style="list-style-type: none"><li>• Harris, Harris: Digital Design and Computer Architecture - ARM Edition, Morgan Kaufman Verlag, 2015.</li><li>• Kesel, Bartholomä: Entwurf von digitalen Schaltungen und Systemen mit HDLs und FPGAs, Oldenbourg Verlag, 2009.</li></ul> Aktuelle Literaturhinweise werden zu Beginn des Semesters bekanntgegeben.		
<b>Modulkoordination</b>	Prof. Dr.-Ing. Heiko Hinkelmann		
<b>Lehrende</b>	Prof. Dr.-Ing. Heiko Hinkelmann, Prof. Dr. Thomas Hollstein		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Computer Architecture		

Modulnummer: 25a	Software-Projekt Mobile und verteilte Anwendungen	
Modultitel (englisch)	Software Project mobile and distributed applications	
Dauer des Moduls Ein Semester	Lehrsprache(n) des Moduls Deutsch	Prüfungssprache(n) des Moduls Deutsch
Art des Moduls Wahlpflichtmodul	Häufigkeit des Angebots Jedes Semester	ECTS-Punkte (CP) / Workload (h) 5 CP / 150 Stunden
Verwendbarkeit des Moduls		
Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung	1. Modul 4 "Einführung in die Programmierung" 2. Modul 11 "Software Engineering – Analysis“ oder Modul M17 „Software Engineering – Design“ und mindestens 80 CP aus den ersten 4 Semestern	
Voraussetzung für die Vergabe von ECTS-Punkten:		
A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)	Keine	
B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)	Projektarbeit Softwareentwicklung (Bearbeitungszeit 12 Wochen) mit Präsentation (mindestens 10, höchstens 20 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
Lernergebnisse / Kompetenzen	Nach Abschluss des Moduls sind die Studierenden in der Lage <ul style="list-style-type: none"><li>durch das Bearbeiten eines komplexen Software-Projekts aus dem Bereich mobile Systeme oder verteilte Anwendungen ihre technischen Fähigkeiten in Programmierung, Dokumentation, SW-Engineering, Präsentation und Kommunikation zu optimieren,</li><li>Aufgaben aus einem oder mehreren Gebieten des Curriculums zu lösen (z.B. Verteilte Anwendung, Mobile Devices etc.),</li><li>ein Projektziel in einer vorgegebenen Zeit zu erreichen</li><li>sich als Team sowie selbst als Mitglied eines Teams zu organisieren,</li><li>auf hohem technischen Niveau mit anderen zu kommunizieren und unerwartete Schwierigkeiten (sowohl technischer Art als auch sozialer Art) zu überwinden, (sowohl technischer Art als auch sozialer Art),</li><li>Toleranz gegenüber den Projektpartnerinnen und -partnern zu üben,</li><li>Verantwortung in einem Projektkontext zu übernehmen.</li></ul>	
Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV		
Titel der Lehrveranstaltung	Lehr- und Lernformen	SWS
a) Software-Projekt Mobile und Verteilte Anwendungen	Projektarbeit	4 SWS
Empfohlenes Fachsemester	5. Semester	
SWS des Moduls/ Kontaktzeiten des Moduls 4 SWS / 60 Stunden	Selbststudium (h) 90	Praxiszeiten (h) 0
Inhalte des Moduls	Ein Thema aus dem Bereich mobile Systeme oder verteilte Anwendungen.	
Literatur	Literatur abhängig vom aktuellen Projekt, wird zu Beginn der Veranstaltung bekannt gegeben.	
Modulkoordination	Prof. Dr. Matthias Deegener	
Lehrende	alle Professoren des Studiengangs Informatik – mobile Anwendungen	
Hinweise (insbesondere empfohlene inhaltliche Vorkenntnisse)	Module der objektorientierten Programmierung, des Software Engineerings und Echtzeit-Systeme	

<b>Module number: 25b</b>		<b>Databases</b>	
<b>Module title (English)</b>		Databases	
<b>Module duration</b> One semester		<b>Language(s) of instruction for this module</b> English	<b>Language(s) of examination for this module</b> English
<b>Module type</b> Elective module		<b>Availability</b> Every winter semester	<b>ECTS points (CP) / workload (h)</b> 5 CP / 150 h
<b>Module relevance</b>		Informatik (B. Sc.), Informatik- Mobile Anwendungen (B.Sc.)	
<b>Admission requirement(s) for the module and module examination</b>		None	
<b>Requirements for awarding ECTS points:</b>			
<b>A) Prerequisite assessments</b> for admission to the module examination ( <b>type, workload, duration and grading scheme</b> )		None	
<b>B) Final module assessment</b> ( <b>type, workload, duration and grading scheme</b> )		Written examination (120 minutes) <u>Grading scheme:</u> graded as per Section 15 AB BA/MA FRA UAS	
<b>Learning outcomes / skills</b>		Upon completion of the module the student is able to <ul style="list-style-type: none"><li>• Outline the relational data model and apply it practically</li><li>• Master the standard database language SQL by using a specific database management system</li><li>• The students solve a given problem in a structured manner and have to develop their creative skills.</li></ul>	
<b>Courses, learning and teaching formats, and SCH of the course:</b>			
<b>Course title</b>		<b>Learning and teaching formats</b>	<b>Course SCH</b>
a) Lecture Databases		Lecture	4 SWS
b) Exercise Databases		Exercise	2 SWS
<b>Recommended semester of study</b>		Semester 5 (für Studierende der Vertiefungsrichtung Technische Informatik)	
<b>SCH / contact hours of the module</b> 6 SCH/ 90 h		<b>Independent study (h)</b> 60	<b>Practice hours (h)</b> 0
<b>Module content</b>		a) <ul style="list-style-type: none"><li>• Introduction: database system concepts, database architecture, data models</li><li>• The relational data model, relational database constraints, relational algebra</li><li>• Data modeling and relational database schema: Entity-Relationship model, ER-to-relational mapping, normalization</li><li>• 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).</li><li>• Introduction to SQL Programming techniques: database functions, procedures, trigger, interfaces</li></ul> Transaction processing concepts, user administration, data dictionary views b) 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.	
<b>Literature</b>		<ul style="list-style-type: none"><li>• Elmasri, R. and S. Navathe. Fundamentals of Database Systems, Addison Wesley Publishing Company</li><li>• Date, C.J., An Introduction to Database Systems. AddisonWesley</li><li>• Garcia-Molina, H., J. D. Ullman and J. D. Widom, Database Systems: The Complete Book, Prentice Hall.</li><li>• Härder, Theo; Rahm, Erhard: Datenbanksysteme - Konzepte und Techniken der Implementierung, Springer</li><li>• Kemper, A. und A. Eickler. Datenbanksysteme, Oldenbourg.</li><li>• Kifer, M., A. Bernstein and P.M. Lewis. Database Systems: An Application-Oriented Approach. Pearson International</li><li>• Ramakrishnan, R. and J. Gehrke, Database Management Systems, McGraw-Hill</li><li>• Silberschatz, A., H.F. Korth, S. Sundershan, Database System Concepts, McGraw Hill.</li></ul>	
<b>Module coordination</b>		Prof. Dr. Christian Rich	
<b>Teachers</b>		Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich	

<b>Notes</b> (In particular, recommended prior 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.
--------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## Modul 26: Wahlpflichtmodul

Modultitel	Wahlpflichtmodul
Modulnummer	26

Die für den Studiengang vorgesehenen Wahlpflichtmodule werden jedes Semester aus einem bestehenden Modulpool im Fachbereichsrat beschlossen.

## Modul 27: Wahlpflichtmodul

Modultitel	Wahlpflichtmodul
Modulnummer	27

Die für den Studiengang vorgesehenen Wahlpflichtmodule werden jedes Semester aus einem bestehenden Modulpool im Fachbereichsrat beschlossen.

## Modul 28: Interdisziplinäres Studium Generale

Modultitel	Interdisziplinäres Studium Generale
Modulnummer	28
Studiengang	Alle Bachelor-Studiengänge der Frankfurt University of Applied Sciences.
Verwendbarkeit des Moduls	Alle Bachelor-Studiengänge der Frankfurt University of Applied Sciences
Dauer des Moduls	Ein Semester
Empfohlenes Semester im Studienverlauf	5. Semester
Art des Moduls	Pflichtmodul
ECTS-Punkte (CP) / Workload (h)	5 CP / 150 Stunden
Empfohlene inhaltliche Vorkenntnisse	Empfohlene Voraussetzungen: 60 ECTS-Punkte im Fachstudium
Voraussetzungen für die Teilnahme am Modul und an der Modulprüfung	<b>Keine</b>
Voraussetzung für die Vergabe von Leistungspunkten: a. Vorleistung b. Modulprüfung	<b>a. Keine</b> <b>b. Projektarbeit</b> (Bearbeitungszeit: Variabel, je nach Modulexemplar) mit Präsentation (Variabel je nach Modulexemplar)
Lernergebnisse und Kompetenzen	<p>Die Studierenden erweitern die fachspezifischen Denkweisen (Theorien und Methoden) durch Einblicke in Fachwissen, Methodenkenntnisse und Denkweisen anderer Disziplinen.</p> <p>Die Studierenden sind in der Lage:</p> <ul style="list-style-type: none"> <li>• interdisziplinär zu denken und unterschiedliche Aspekte eines Querschnittsthemas zu erkennen, diese gegeneinander abzuwägen und ganzheitlich zu reflektieren,</li> <li>• Zusammenhänge ihres künftigen Berufsfelds im Raum unterschiedlicher Disziplinen sowie gesellschaftlicher Interessen verständlich zu machen und diese Zusammenhänge fachlich versiert darzustellen und argumentativ zu vertreten,</li> <li>• die Wirkungen und Folgen ihrer beruflichen und gesellschaftlichen Tätigkeit zu reflektieren und daraus Konsequenzen für ihr eigenes Handeln abzuleiten,</li> <li>• anhand konkreter interdisziplinärer Aufgabenstellungen Verständnis für die fachfremden Denkweisen zu entwickeln und kooperativ im Umgang mit verschiedenen Kulturen und Wertesystemen zu handeln.</li> </ul> <p>Die Studierenden lernen neue Methoden und inhaltliche Kenntnisse auf konkrete Problemstellungen anzuwenden (je nach Modulexemplar).</p>
Inhalte des Moduls	Ein Querschnittsthema unter Beteiligung von mindestens zwei Fachbereichen und drei Fachdisziplinen der Frankfurt University of Applied Sciences. <i>Gemäß der aktuellen Ankündigungen auf der Studium Generale-Webseite.</i>
Lehrformen des Moduls	Projekt
Sprache	Variabel, je nach Modulexemplar
Häufigkeit des Angebots von Modulen	In jedem Semester
Modulkoordination	Variabel, je nach Modulexemplar - Gemäß der aktuellen Ankündigungen auf der Studium Generale-Webseite.
Hinweise	<p>Die Hinweise zu Anforderungen, Projektthemen und Besonderheiten (Blockveranstaltung, Englische Sprache, Blended Learning, Virtuelles Klassenzimmer, Technische Voraussetzungen, Semesterplan) sind für jedes Modulexemplar in den konkreten Unitbeschreibungen zu finden.</p> <p>Regulärer Termin der Veranstaltung jeweils Mittwochnachmittag (in der Regel 4. und 5. Block).</p>

<b>Modulnummer: 29</b>		<b>Praxisphase</b>		
<b>Modultitel (englisch)</b>				
<b>Dauer des Moduls</b> Ein Semester		<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch	
<b>Art des Moduls</b> Pflichtmodul		<b>Häufigkeit des Angebots</b> Jedes Semester	<b>ECTS-Punkte (CP) / Workload (h)</b> 18 CP / 540 Stunden	
<b>Verwendbarkeit des Moduls</b>		verwendbar in anderen Informatik-Bachelor-Studiengängen		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>		120 CP aus den ersten 5 Semestern		
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>				
<b>A) Vorleistungen als Modulprüfungsvoraussetzung (Art, Umfang, Dauer und Art der Benotung)</b>		Keine		
<b>B) Modulabschließende Prüfungsleistung (Art, Umfang, Dauer und Art der Benotung)</b>		b. Praxisbericht (Bearbeitungszeit 12 Wochen) mit mündlichem Vortrag (mindestens 15, höchstens 20 Minuten)  Weitere Voraussetzung gem. Praktikumsverordnung: Nachweis über die Durchführung des berufspraktischen Zeitraums durch die Praxisstelle  <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS		
<b>Lernergebnisse / Kompetenzen</b>		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 Teilnehmerinnen und Teilnehmern ausgetauscht. 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. Neben der fachlichen Projektarbeit sind 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. Sie haben Einblick in wichtige Anwendungsfelder der Informatik und Verständnis gewonnen, besitzen die Fähigkeit zur Beurteilung von 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.		
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>				
<b>Titel der Lehrveranstaltung</b>		<b>Lehr- und Lernformen</b>		<b>SWS</b>
a) Seminar zur Praxisphase		Seminar		2 SWS
b) Betreutes Praxisprojekt		Projekt in einem Betrieb		6 SWS
<b>Empfohlenes Fachsemester</b>		6. Semester		
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 2 SWS / 30 Stunden		<b>Selbststudium (h)</b> 0	<b>Praxiszeiten (h)</b> 510	
<b>Inhalte des Moduls</b>		Zeitgemäße Präsentation der Praxisprojekte in einem 20-minütigen Vortrag mit <ul style="list-style-type: none"><li>• Diskussion zum eigenen Vortrag und Beteiligung an der Diskussion zu anderen Vorträgen.</li><li>• Layout von Präsentationsseiten (Folien/Laptop/Beamer)</li><li>• 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.</li></ul>		
<b>Literatur</b>				
<b>Modulkoordination</b>		Prof. Dr. Jens Liebehenschel		
<b>Lehrende</b>		Prof. Dr. Jens Liebehenschel		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)		Programmierung, Software Engineering und Kenntnisse aus den vertiefenden Vorlesungen der vorangehenden Semester Zur Vorbereitung der Praxisphase erfolgt in vorangehenden Semestern eine einführende Veranstaltung, deren Termin jeweils per Aushang mitgeteilt wird.		

<b>Modulnummer: 30</b>	<b>Bachelor-Arbeit mit Kolloquium</b>	
<b>Modultitel (englisch)</b>	Bachelor-Thesis with Colloquium	
<b>Dauer des Moduls</b> Ein Semester	<b>Lehrsprache(n) des Moduls</b> Deutsch	<b>Prüfungssprache(n) des Moduls</b> Deutsch
<b>Art des Moduls</b> Pflichtmodul	<b>Häufigkeit des Angebots</b> Jedes Semester	<b>ECTS-Punkte (CP) / Workload (h)</b> 12 CP / 360 Stunden
<b>Verwendbarkeit des Moduls</b>		
<b>Teilnahmevoraussetzung(en) am Modul und an der Modulprüfung</b>	Erfolgreicher Abschluss aller Module M1 bis M28 sowie der Nachweis des Beginns des Moduls M29 Praxisphase in Form eines unterzeichneten Praktikumsvertrages.	
<b>Voraussetzung für die Vergabe von ECTS-Punkten:</b>		
<b>A) Vorleistungen</b> als Modulprüfungsvoraussetzung ( <b>Art, Umfang, Dauer und Art der Benotung</b> )	Keine	
<b>B) Modulabschließende Prüfungsleistung</b> ( <b>Art, Umfang, Dauer und Art der Benotung</b> )	b. Bachelor-Arbeit (Bearbeitungszeit 9 Wochen) mit Kolloquium (mindestens 30, höchstens 60 Minuten) <u>Art der Benotung:</u> benotet gemäß § 15 AB BA/MA FRA UAS	
<b>Lernergebnisse / Kompetenzen</b>	Nach Abschluss des Moduls sind die Studierenden in der Lage - selbständig ein Thema ihres Fachs zu bearbeiten - eine einfachere, praxisbezogene Aufgabe mit wissenschaftlichen Methoden und Verfahren des Fachs sicher zu bearbeiten - eine wissenschaftlich fundierte, schriftliche Ausarbeitung zu erstellen - eigene Ideen und Ergebnisse gegenüber fachlicher Kritik argumentativ zu vertreten  Die in diesem Modul erworbenen Kompetenzen helfen Absolventinnen und Absolventen nach Abschluss ihres Studiums in allen avisierten Einsatzfeldern.	
<b>Lehrveranstaltungen (LV), Lehr- und Lernformen und SWS der LV</b>		
<b>Titel der Lehrveranstaltung</b>	<b>Lehr- und Lernformen</b>	<b>SWS</b>
a) Bachelor-Arbeit mit Kolloquium	selbstständiges Arbeiten	1 SWS
<b>Empfohlenes Fachsemester</b>	6. Semester	
<b>SWS des Moduls/ Kontaktzeiten des Moduls</b> 1 SWS / 15 Stunden	<b>Selbststudium (h)</b> 0	<b>Praxiszeiten (h)</b> 345
<b>Inhalte des Moduls</b>		
<b>Literatur</b>		
<b>Modulkoordination</b>	Prof. Dr. Jörg Schäfer	
<b>Lehrende</b>		
<b>Hinweise</b> (insbesondere empfohlene inhaltliche Vorkenntnisse)	Alle Module der ersten 5 Studiensemester Das Kolloquium findet am Ende der Arbeit mit den ReferentInnen statt. Im Kolloquium zur Bachelorarbeit soll die oder der Studierende ihre oder seine Bachelorarbeit gegenüber kritischen Fragen verteidigen.	