

Modulhandbuch

Bachelor-Verbund-Studiengang

Angewandte Informatik

FPO März 2020

Stand: Sommersemester 2024

Studienverlaufsplan

Semester	Module					
1	Mathematik für Informatiker 1	Studiertechniken und wissenschaftliches Schreiben	Grundlagen der Programmierung			
2	Mathematik für Informatiker 2	Rhetorik und Präsentationstechniken	Algorithmen und Datenstrukturen	Imperative und Prozedurale Programmierung		
3	Datenbanken 1	Rechnernetze 1	Wahlpflichtblock BWL 1 / Grundlagen des Grafikdesign / Datenschutz	Grundlagen der Objektorientierten Programmierung		
4	Software-Engineering	Rechnerarchitektur	Wahlpflichtblock BWL 2 / Social Media / IT-Recht /	<i>Studienrichtung Anwendungsentwicklung</i>	<i>Studienrichtung Künstliche Intelligenz</i>	<i>Studienrichtung Systemintegration</i>
				Datenbanken 2	Datenbanken 2	Rechnernetze 2
5	Webentwicklung Frontend	Grundlagen Betriebssysteme	Wahlpflichtblock BWL 3 / Mediendesign / Cloud-, Big-Data- & KI-Recht	Design Pattern in der OOP	Skriptsprachen	Skriptsprachen
6	Webentwicklung Backend	IT-Projektmanagement	Graphen, Codes, und Automaten	Mobile Applikationen	Einführung Machine Learning	Unix-artige Betriebssysteme
7	Softwarequalität	Konferenzseminar	Wahlpflichtmodul 1	Java Enterprise Edition	Deep Learning	Microsoft Betriebssysteme
8	Projekt		Wahlpflichtmodul 2	3D- Grafikprogrammierung	Natural Language Processing	Virtualisierung
9	Bachelorseminar	Bachelorarbeit und Kolloquium				

Inhaltsverzeichnis

Container	7
Pflichtmodule des 1. Semesters	8
Mathematik für Informatiker 1	8
Grundlagen der Programmierung	10
Studiertechniken und wissenschaftliches Arbeiten (Studier- und Präsentationstechniken).....	12
Pflichtmodule des 2. Semesters	14
Mathematik für Informatiker 2	14
Algorithmen und Datenstrukturen	16
Imperative und prozedurale Programmierung	18
Rhetorik und Präsentationstechniken (Rhetorik und wissenschaftliches Schreiben)	20
Pflichtmodule des 3. Semesters	23
Datenbanken 1	23
Rechnernetze 1.....	25
Grundlagen der objektorientierten Programmierung	27
Pflichtmodule des 4. Semesters	29
Software Engineering	29
Rechnerarchitektur.....	31
Studienrichtung Anwendungsentwicklung und Künstliche Intelligenz	33
Datenbanken 2	33
Studienrichtung Systemintegration	35
Rechnernetze 2.....	35
Pflichtmodule des 5. Semesters	37
Webentwicklung Frontend.....	37
Grundlagen Betriebssysteme	39
Studienrichtung Anwendungsentwicklung	41
Design Patterns in der objektorientierten Programmierung.....	41
Studienrichtung Systemintegration und Künstliche Intelligenz	43
Skriptsprachen	43
Pflichtmodule des 6. Semesters	45
Webentwicklung Backend	45
IT-Projektmanagement	47
Graphen, Codes und Automaten	49
Studienrichtung Anwendungsentwicklung	51
Mobile Applikationen.....	51

Studienrichtung Systemintegration	53
Unix-artige Betriebssysteme.....	53
Studienrichtung Künstliche Intelligenz.....	55
Einführung Machine Learning.....	55
Pflichtmodule des 7. Semesters.....	57
Softwarequalität.....	57
Konferenzseminar.....	59
Studienrichtung Anwendungsentwicklung	61
Java Enterprise Edition	61
Studienrichtung Systemintegration	63
Microsoft-Betriebssysteme	63
Studienrichtung Künstliche Intelligenz.....	65
Deep Learning.....	65
Pflichtmodule des 8. Semesters.....	67
Projekt.....	67
Studienrichtung Anwendungsentwicklung	68
3D-Grafikprogrammierung.....	68
Studienrichtung Systemintegration	70
Virtualisierung	70
Studienrichtung Künstliche Intelligenz.....	72
Natural Language Processing	72
Pflichtmodule des 9. Semesters.....	74
Bachelorseminar.....	74
Bachelorarbeit	76
Kolloquium	77
Wahlpflichtblock Betriebswirtschaftslehre.....	78
Betriebswirtschaftslehre 1.....	78
Betriebswirtschaftslehre 2.....	80
Betriebswirtschaftslehre 3.....	82
Wahlpflichtblock Mediendesign.....	84
Grundlagen des Grafikdesigns.....	84
Social Media (Digital and Social Media Marketing).....	86
Mediendesign (Grundlagen Audiovisuelles Mediendesign).....	88
Wahlpflichtblock Recht	90
Datenschutz.....	90
IT-Recht.....	92
Big-Data- Cloud und KI-Recht.....	94
Containermodule	96
Container Algorithmen	96

Praktische Anwendungen von Algorithmen (Container Algorithmen)	96
Container Anwendungsgebiete der Informatik	99
Grundlagen des Grafikdesigns (Container Anwendungsgebiete der Informatik)	99
Mediendesign	99
Social Media (Container Anwendungsgebiete der Informatik)	99
Container Betriebswirtschaft	100
Betriebswirtschaftslehre 1 (Container Betriebswirtschaft)	100
Betriebswirtschaftslehre 2 (Container Betriebswirtschaft)	100
Betriebswirtschaftslehre 3 (Container Betriebswirtschaft)	100
Container Datenanalyse	101
Container Datenbanksysteme	102
Datenbanken 2 (Container Datenbanksysteme)	102
Container Gesellschaftliche Themen	103
Container Grafische Datenverarbeitung	104
Container Internet der Dinge	105
Container Künstliche Intelligenz	106
Deep Learning (Container Künstliche Intelligenz)	106
Einführung Machine Learning (Container Künstliche Intelligenz)	106
Natural Language Processing (Container Künstliche Intelligenz)	106
Container Programmiersysteme	107
3D-Grafikprogrammierung (Container Programmiersysteme)	107
Design Pattern in der OOP (Container Programmiersysteme)	107
Java Enterprise Edition (Container Programmiersysteme)	107
Mobile Applikationen (Container Programmiersysteme)	107
Skriptsprachen (Container Programmiersysteme)	107
Container Quantencomputing	108
Container Recht und Datenschutz	109
Datenschutz (Container Recht und Datenschutz)	109
IT-Recht (Container Recht und Datenschutz)	109
Big-Data-Cloud und KI-Recht (Container Recht und Datenschutz)	109
Container Spezielle Gebiete Softwareengineering	110
Container Theoretische Informatik	111
Container Verteilte Systeme und Betriebssysteme	112
Microsoft Betriebssysteme (Container Verteilte Systeme und Betriebssysteme)	112
Rechnernetze 2 (Container Verteilte Systeme und Betriebssysteme)	112
Unix-artige Betriebssysteme (Container Verteilte Systeme und Betriebssysteme)	112
Virtualisierung (Container Verteilte Systeme und Betriebssysteme)	112
Container Vorgehensmodelle / Projektmanagement	113

Container

Folgende Container sind diesem Studiengang zugeordnet:

- Algorithmen
- Anwendungsgebiete der Informatik
- Betriebswirtschaft
- Datenanalyse
- Datenbanksysteme
- Gesellschaftliche Themen
- Grafische Datenverarbeitung
- Internet der Dinge
- Künstliche Intelligenz
- Programmiersysteme
- Quantencomputing
- Recht und Datenschutz
- Spezielle Gebiete Softwareengineering
- Theoretische Informatik
- Verteilte Systeme und Betriebssysteme
- Vorgehensmodelle / IT-Projektmanagement

Die Module, die den einzelnen Containern zugeordnet sind, finden Sie im Bereich „Containermodule“ dieses Modulhandbuchs.

Pflichtmodule des 1. Semesters

Mathematik für Informatiker 1					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	1. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Übung	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Übung: 30 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden lernen die grundlegenden mathematischen Methoden zur Lösung ingenieurwissenschaftlicher Aufgabenstellungen kennen und können sie anwenden. Sie sind in der Lage, Terme und einfache Gleichungen sicher umzuformen, können die Lösungsmenge von Ungleichungen bestimmen und mit komplexen Zahlen rechnen. Die Methoden der Kombinatorik zum Abzählen endlicher Mengen können benutzt und die Genauigkeit von Rechenergebnissen beurteilt werden.</p> <p>Die Studierenden erlangen die Kompetenz, mit Zahlenfolgen und unendlichen Reihen umzugehen, reelle Funktionen und ihre charakteristischen Eigenschaften zu untersuchen und diese zu differenzieren sowie eine Kurvendiskussion durchzuführen.</p> <p>Nach Abschluss des Moduls sind die Studierenden mit Vokabular und Methodik der Analysis in einem Umfang vertraut, der sie in die Lage versetzt, mathematische Teilprobleme (aus der Analysis) in der Informatik zu erkennen, geeignete Hilfsmittel für die Beschreibung und Lösung des Problems zu erkennen und diese Problemlösung schließlich praktisch – aber mathematisch korrekt – umzusetzen.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Allgemeine Grundlagen Aussagen und logische Verknüpfungen, Mengen, Relationen und Abbildungen, Gleichungen und Ungleichungen, Kombinatorik, numerisches Rechnen und elementare Fehlerrechnung • Komplexe Zahlen Imaginäre Einheit, Real- und Imaginärteil, Gaußsche Zahlenebene, Polar- und Exponentialform einer komplexen Zahl, Umrechnung der Darstellungsformen, Rechnen mit komplexen Zahlen, Potenzieren, Radizieren und Logarithmieren von komplexen Zahlen • Folgen und Reihen Begriff der Zahlenfolge, Eigenschaften von Folgen, Grenzwert einer Folge, Begriff der unendlichen Reihe, Konvergenzkriterien • Reelle Funktionen Definition und Darstellung einer reellen Funktion, Rechnen mit reellen Funktionen, Eigenschaften reeller Funktionen, Grenzwert und Stetigkeit von reellen Funktionen • Spezielle Funktionen Ganzrationale Funktionen, gebrochenrationale Funktionen, irrationale Funktionen, Exponentialfunktionen, Logarithmusfunktionen, trigonometrische Funktionen • Differentialrechnung Differenzierbarkeit, Ableitungsregeln, Differentiation nach Logarithmieren, Ableitung der Umkehrfunktion, höhere Ableitungen, die Regeln von de L'Hospital, Monotonie- und Krümmungsverhalten reeller Funktionen, Extrema, Kurvendiskussion 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzveranstaltungen in Form von seminaristischem Unterricht und Übungen
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen

Grundlagen der Programmierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	250 h	10 ECTS	1. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 6 SWS Tutorium 2 SWS Praktikum	Kontaktzeit 36 h	Selbststudium 214 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden beherrschen die mathematisch-physikalischen und die informations-technologischen Grundlagen der Programmierung von Computern der Von-Neumann-Architektur, insbesondere die abstrakte Funktionsweise derartiger Computer sowie die für die effektive Nutzung erforderlichen Digitalisierungen und Binärkodierungen von Kommunikationsinhalten (Zahlen, Texte, Bilder, Sprache, Musik, Videos).</p> <p>Sie verstehen die interdisziplinäre Verankerung der Programmierung von Computern der Von-Neumann-Architektur, können Strukturen und Regeln in Problemstellungen erkennen und resultierende Problemlösungen algorithmisch formulieren. Sie haben einen Überblick über die verschiedenen Möglichkeiten der Überführung algorithmischer Formulierungen in Programme für Computer der Von-Neumann-Architektur und verstehen den praktischen Wert von präzisen algorithmischen Formulierungen. Insbesondere können Sie derartige Programme im Hinblick auf Effektivität und Effizienz beurteilen.</p>				
3	<p>Inhalte</p> <p>Im Tutorium werden nachfolgende Themengebiete des Lehrbriefs vertieft:</p> <ul style="list-style-type: none"> • Einführung in die Begriffswelt der Programmierung • Architektonische und linguistische Grundlagen der Programmierung <ul style="list-style-type: none"> • Prinzipien und Gewerke der Von-Neumann-Architektur • Programme für Computer der Von-Neumann-Architektur • Programmerstellung für Computer der Von-Neumann-Architektur (Pseudocode, Flussdiagramme, Programmiersprachen) • Mathematische Grundlagen der Programmierung <ul style="list-style-type: none"> • Mengen und Relationen • Aussagenlogik • Zahlentheorie (Darstellungen von Zahlen, Primzahlen, Zufallszahlen) • Kombinatorik (Abzähltechniken, Häufigkeiten, Wahrscheinlichkeiten) • Physikalische Grundlagen der Programmierung <ul style="list-style-type: none"> • Eigenschaften akustischer und elektromagnetischer Wellen • Diskretisierung und Quantisierung von Wellen • Pulse Code Modulation (PCM, ADPCM) • Anthropologische Grundlagen der Programmierung <ul style="list-style-type: none"> • Menschliche Empfindung • Menschliche Wahrnehmung • Informations-technologische Grundlagen der Programmierung <ul style="list-style-type: none"> • Elemente der Kodierungstheorie (Binärkodierungen, Stromkodierungen, Blockkodierungen, Entropiekodierungen (Shannon, Huffman), Quellkodierungen, Hybridkodierungen) 				

3	<ul style="list-style-type: none"> • Praktisch relevante Digitalisierungen und Binärkodierungen für <ul style="list-style-type: none"> • Zahlen • Symbole/Glyphen • Texte • Sprache • Musik • Bilder <ul style="list-style-type: none"> • Vektorgrafik • Rastergrafik • Videos <p>Im Praktikum werden begleitend Lösungen für Aufgabenstellungen der jeweiligen Themengebiete erarbeitet, in Programme für Computer der Von-Neumann-Architektur überführt (vorwiegend mit Hilfe von Flussdiagrammen) und hinsichtlich Effektivität und Effizienz analysiert.</p>
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Tutorium und Praktikum</p>
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: –</p>
6	<p>Prüfungsformen</p> <p>Klausur</p>
7	<p>Prüfungsvorleistung</p> <p>Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert</p>
8	<p>Voraussetzungen für die Vergabe von Kreditpunkten</p> <p>bestandene Modulprüfung</p>
9	<p>Verwendung des Moduls (in anderen Studiengängen)</p> <p>–</p>
10	<p>Stellenwert der Note für die Endnote</p> <p>10/180</p>
11	<p>Modulbeauftragte*r und hauptamtlich Lehrende</p> <p>Prof. Dr. Rainald Schöneberg, Dipl.-Inform. (FH) Uwe Gogolin</p>
12	<p>Sonstige Informationen</p> <p>Literaturauswahl (jeweils in der aktuellen Auflage):</p> <p>Kenneth H. Rosen: Discrete Mathematics and Its Applications, McGraw-Hill</p> <p>David A. Patterson and John L. Hennessy: Computer Organization and Design, Morgan Kaufmann Publishers</p> <p>Weiterführende Literaturangaben zu den einzelnen Themengebieten befinden sich im Lehrbrief</p>

Studiertechniken und wissenschaftliches Arbeiten (Studier- und Präsentationstechniken)					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	1. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Nach erfolgreicher Teilnahme haben die Studierenden wesentliche nicht-fachliche Kompetenzen erworben, die für das weitere Studium notwendig sind.</p> <p>Die Studierenden lernen grundlegende Techniken und Methoden für eigenständiges und teamorientiertes Lernen kennen. Dabei bilden theoretische und praktische Inhalte zum Zeit- und Arbeitsmanagement, Studiertechniken sowie Organisationskompetenz und Selbstmanagement wesentliche Bausteine.</p> <p>Die Studierenden können ihren eigenen Lernprozess beobachten, dokumentieren und reflektieren.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Organisation und Reflexion des eigenen Lernprozesses, Studientagebuch, Lern-, Konzentrations- und Entspannungstechniken, Erstellen von Wochen- und Lernplänen • Gruppendynamische Prozesse, Bildung von Lern- und Arbeitsgruppen • Lese- und Schreibetechniken • Kreativitäts- und Schreibetechniken, Postererstellung / Präsentationstechnik • Umgang mit LaTeX inkl. praktischer Übungen • Anforderungen an wissenschaftliche Ausarbeitungen, Typen von wissenschaftlichen Arbeiten • Formale Richtlinien • Literaturverwaltung- und Recherche • Zitate und Schreiben von Texten • Zeitplanung • Wie finde ich mein Thema • Exkurs: Vorarbeit und Durchführung der Bachelor-Arbeit am Ende des Studiums 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als seminaristischer Unterricht</p> <p>Theoretischer und praktischer Diskurs in den Präsenzveranstaltungen</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: Termingerechte Anmeldung für die Ausarbeitung und das Referat, Einhalten der gesetzlichen Fristen</p>				

	Inhaltlich: –
6	Prüfungsformen Kombinationsprüfung aus semesterbegleitender schriftlicher Ausarbeitung und einem Referat
7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) --
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen:

Pflichtmodule des 2. Semesters

Mathematik für Informatiker 2					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	2. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Übung	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Übung: 30 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden kennen die grundlegenden mathematischen Methoden zur Lösung ingenieurwissenschaftlicher Aufgabenstellungen und können sie anwenden. Sie sind in der Lage, die Potenzreihenentwicklung einer Funktion zu berechnen und bei der Approximation zu benutzen. Sie können reelle Funktionen mit Hilfe der behandelten Techniken integrieren und mit Vektoren und Matrizen umgehen, insbesondere bei Anwendungen in der analytischen Geometrie.</p> <p>Die Studierenden verstehen und üben Standardverfahren der linearen Algebra praktisch ein. Sie sind in der Lage, lineare Gleichungssysteme mit Hilfe des Gauß-Algorithmus zu lösen und die Determinante einer Matrix zu berechnen.</p> <p>Nach Abschluss der beiden Mathematik-Module sind die Studierenden mit fortgeschrittenem Vokabular und Methodik der Analysis und der linearen Algebra in einem Umfang vertraut, der sie in die Lage versetzt, mathematische Teilprobleme (aus beiden Gebieten der Mathematik) in der Informatik zu erkennen, geeignete Hilfsmittel für die Beschreibung und Lösung des Problems zu bestimmen und diese Problemlösung schließlich praktisch – aber mathematisch korrekt – umzusetzen.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Integralrechnung das bestimmte Integral, das Flächenproblem, allgemeine Definition des bestimmten Integrals, allgemeine Integrationsregeln und Eigenschaften des bestimmten Integrals, der Hauptsatz der Differential- und Integralrechnung, Grund- oder Stammintegrale, Integrationsmethoden, partielle Integration, Integration durch Substitution, Integration gebrochenrationaler Funktionen, uneigentliche Integrale • Vektorrechnung Skalare und vektorielle Größen, Vektor als Abbildung, dreidimensionaler Vektorraum, Vektoraddition und Multiplikation mit einem Skalar, Skalarprodukt, n-dimensionaler Vektorraum, lineare Abhängigkeit und Unabhängigkeit, Vektor- und Spatprodukt, analytische Geometrie • Matrizen und lineare Gleichungssysteme Definition einer Matrix, Rechnen mit Matrizen, Matrizen als lineare Abbildungen, lineare Gleichungssysteme, Koeffizientenmatrix eines linearen Gleichungssystems, Zeilennormalform einer Matrix, Gauß-Jordan-Verfahren, Lösbarkeit linearer Gleichungssysteme, Berechnung der inversen Matrix, Determinanten 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzveranstaltungen in Form von seminaristischem Unterricht und Übungen
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: Beherrschung des Stoffes aus Mathematik für Informatiker*innen 1
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen

Algorithmen und Datenstrukturen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	2. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden lernen fundamentale Paradigmen, Fakten und Methoden der angewandten Informatik kennen und üben die Methoden praktisch ein. Sie verstehen das Zusammenspiel von Algorithmen und Computertechnik. Anhand exemplarischer Algorithmen für grundlegende Probleme der Informatik machen sie sich mit „algorithmischem Denken“ sowie einfachen Methoden für den Entwurf und die Analyse effizienter Algorithmen vertraut.</p> <p>Die Arbeit mit Algorithmen führt automatisch und natürlich zur Beschäftigung mit grundlegenden Datenstrukturen: Studierende lernen deren Eigenschaften kennen und verstehen, und sie erlangen die Kompetenz, geeignete Datenstrukturen für typische Einsatzszenarien auszuwählen.</p> <p>Grundlegende Überlegungen zur Effizienz von Algorithmen versetzen die Studierenden in die Lage, bei Verfügbarkeit alternativer Lösungsansätze den effizientesten auszuwählen und diese Auswahl mathematisch korrekt zu begründen.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Darstellungsformen für Algorithmen • Einführung in Datenstrukturen (Stacks, Queues, verkettete Listen, Suchbäume) • Einführung in effiziente Algorithmen und Datenstrukturen (Bäume, Hashverfahren) • Effizienz und Korrektheit von Algorithmen • Worst-case-, Average-case- und Best-case-Betrachtung • Algorithmen für Suchen und Sortieren 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: – Inhaltlich: –</p>				
6	<p>Prüfungsformen</p> <p>Klausur</p>				
7	<p>Prüfungsvorleistung</p> <p>Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert</p>				
8	<p>Voraussetzungen für die Vergabe von Kreditpunkten</p> <p>bestandene Modulprüfung</p>				

9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Cormen, Leiserson, Rivest, Stein: <i>Algorithmen – Eine Einführung</i> , Oldenbourg Verlag, München Ernst: <i>Grundkurs Informatik</i> , Vieweg+Teubner Verlag, Wiesbaden Herold, Lurz, Wohlrab: <i>Grundlagen der Informatik</i> , Pearson Deutschland, München Gumm, Sommer: <i>Einführung in die Informatik</i> , Oldenbourg Verlag, München

Imperative und prozedurale Programmierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	2. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden können Entwicklungsumgebungen einsetzen, um Programme zu erstellen. Sie kennen die Werkzeuge der einzelnen Arbeitsschritte zur Programmerstellung sowie die benötigten Systemkomponenten und sind somit in der Lage, für spezielle informationstechnische Problemstellungen Programme zu erstellen.</p> <p>Die Studierenden können die Grundprinzipien der imperativen und prozeduralen Programmierung in eigenen Programmen anwenden. Sie sind in der Lage, informationstechnische Problemstellungen zu modellieren und einen algorithmischen Lösungsansatz in einer imperativen und prozeduralen Programmiersprache angemessen und im Detail zu realisieren. Sie können auftretende Probleme beim Entwurf und bei der Programmierung im Team gemeinsam lösen, die Ergebnisse erläutern, demonstrieren und verteidigen. Sie sind in der Lage, erhaltene Hinweise in ihre Lösungen einzuarbeiten.</p> <p>Die Studierenden erlangen die Kompetenzen der Analyse- und Synthesefähigkeit. Sie können Wissen selbständig erarbeiten, aufbereiten und weitergeben. Die kommunikativen Kompetenzen werden durch Teamarbeit und Forenkommunikation gestärkt.</p>				
3	<p>Inhalte</p> <p>Unter Verwendung der Programmiersprache Java werden folgende Inhalte vermittelt, demonstriert und erprobt:</p> <ul style="list-style-type: none"> • Quelltexte der imperativen und prozeduralen Programmierung <ul style="list-style-type: none"> • Aufbau und Strukturkonventionen • Elemente und Namenskonventionen • Imperative Programmierung <ul style="list-style-type: none"> • Standarddatentypen, Variablen, Konstanten, Literale • Verbünde, Felder (ein- und mehrdimensional) und Zeichenketten • Ausdrücke und Operatoren • Kontrollstrukturen (if, switch, for, while, do-while) • Prozedurale Programmierung <ul style="list-style-type: none"> • Spezifikation, Signatur und Nutzung einer Methode • Überladen von Methoden (engl. overloading) • Methoden mit variabler Anzahl von Parametern (engl. var-args) • Methoden für den Zufall • Generische, rekursive und anonyme Methoden • Fehler- und Ausnahmebehandlung entlang der Aufrufkette von Methoden • Nebenläufige und verkettete Ausführung von Methoden 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Modul „Grundlagen der Programmierung“
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Rainald Schöneberg
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): D. Abts: <i>Grundkurs JAVA</i> C. Ullenboom: <i>Java ist auch eine Insel</i> , Rheinwerk Computing (http://openbook.rheinwerk-verlag.de/javainsel/) Oracle Java Dokumentationen (https://docs.oracle.com/en/java/javase)

Rhetorik und Präsentationstechniken (Rhetorik und wissenschaftliches Schreiben)					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	2. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden lernen Grundlagen des wissenschaftlichen Arbeitens kennen. Sie erwerben Kenntnisse über die Möglichkeiten der Literaturrecherche und -verwaltung. Sie sind in der Lage, zielgruppengerecht Präsentationen zu planen, zu erstellen und vor einer Gruppe vorzutragen. Sie setzen rhetorische Mittel selbstbewusst, reflektiert und zielgerichtet ein. Sie erläutern und hinterfragen kritisch anhand von Beispielen ihre Einstellungen, Fähigkeiten und Methoden im Umgang mit sich selbst und mit anderen Menschen, insbesondere im Team. Die Studierenden erweitern somit insbesondere ihre kommunikativen und instrumentellen Kompetenzen. Sie üben die Planung, die Gliederung und das Schreiben wissenschaftlicher Ausarbeitungen ein und beachten dabei die in der Informatik üblichen Zitierkonventionen und die Verbote diverser Plagiatformen. Im Bereich des zivilgesellschaftlichen Engagements (Bedeutung der Informatik für die Gesellschaft) haben sie sich erstmals mit der Verantwortung der Informatiker*innen auseinandergesetzt.				
3	Inhalte <ul style="list-style-type: none"> • Rhetorik und Körpersprache, Eigen- und Fremdwahrnehmung, Gruppen- und Teamarbeiten, Problemlösungen, Kreativitätstechniken • Techniken der Bibliotheksrecherche & Quellenbeschaffung, -bewertung, -auswertung, mit Anwendung auf IT-Fragestellungen • Lese- und Schreibtechniken • Korrektes Zitieren, Zitierstile verschiedener Disziplinen, Plagiat-Typen • Professionelle Textverarbeitung und Literaturverwaltung mit LaTeX und BibTeX • Verantwortung als Wissenschaftler*in, Auswirkungen wissenschaftlichen Fehlverhaltens 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als seminaristischer Unterricht				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Klausur				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				

8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage) R. Müller, J. Plieninger, Chr. Rapp: <i>Recherche 2.0</i> J. Knappen: <i>Schnell ans Ziel mit LaTeX</i>

Pflichtmodule des 3. Semesters

Datenbanken 1					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	3. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden verstehen, wie sich Daten in einer relationalen Datenbank speichern lassen, und sie sind in der Lage, eine solche Datenbank selbst zu entwerfen – zunächst formal und dann praktisch durch die Angabe der nötigen SQL-Befehle. Sie kennen wichtige Eigenschaften von Tabellen (etwa die Relevanz von Primär- und Fremdschlüsseln), und sie können Anfragen an eine Datenbank formulieren, um Informationen zu extrahieren und zu kombinieren.</p> <p>Durch den Einsatz von Selbststudienmaterial (Lehrbrief) erwerben die Studierenden eine zusätzliche Lernkompetenz: das selbstständige Erarbeiten von Themen aus einer schriftlichen Abhandlung.</p>				
3	Inhalte <ul style="list-style-type: none"> • Einführung in die Datenmodellierung • kurze Einführung ins Relationenmodell • Vorgehensmodell zur Herleitung einer Datenbankstruktur • Einführung in SQL, insbes. Tabellenstrukturanweisungen, Abfragen einzelner Tabellen, einfache Verbundanweisungen, Anzeigebereitungen • Prinzipieller Aufbau einer relationalen Datenbank 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: Grundlagen der Programmierung, Imperative und Prozedurale Programmierung				

6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Uwe Klug
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): R. Elmasri, S. B. Navathe, <i>Grundlagen von Datenbanksystemen</i> , Addison-Wesley Heuer, G. Saake, <i>Datenbanken: Konzepte und Sprachen</i> , Internat. Thomson Publ. M. Klettke, H. Meyer; <i>XML & Datenbanken</i> ; dpunkt.verlag U. Klug; <i>Datenbankanwendungen entwerfen & programmieren</i> G.Lausen; <i>Datenbanken</i> ; Spektrum Akademischer Verlag G.Saake, K.-U. Sattler; <i>Datenbanken & Java</i> ; dpunkt.Verlag

Rechnernetze 1					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	3. Semester	Winterssemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden erlangen die die theoretischen und praktischen Kompetenzen zur bedarfsgerechten Planung und Weiterentwicklung sowie zum Betrieb der Netzwerkinfrastruktur eines kleinen Unternehmens. Sie beherrschen hierbei insbesondere die in der Praxis relevanten Techniken und Protokolle zur Realisierung von Rechnernetzen im LAN- und WAN-Bereich, wobei der Fokus auf der Internet-Protokoll-Familie liegt. Die Studierenden können Netzwerke auf Basis des TCP/IP- und des OSI-Modells modellieren.</p> <p>Durch den Einsatz von Selbststudienmaterial (E-Learning-Kurs) erwerben die Studierenden eine zusätzliche Lernkompetenz: das selbstständige Erarbeiten von Themen durch einen interaktiven Online-Kurs.</p>				
3	Inhalte <ul style="list-style-type: none"> • Beschreibung der Funktionsweise eines Datennetzwerks (OSI- und TCP/IP-Modell, strukturierte Verkabelung, aktive Netzwerkkomponenten, LAN, WAN) • Planung und Implementierung eines kleinen geschichteten Netzwerks (Ethernet-Technologien, Zugriffsverfahren, Datenübertragung, Netzwerksegmentierung) • Planung und Implementierung eines IP-Adressen-Konzepts und der IP-Dienste für ein kleines Firmennetzwerk (IPv4, IPv6, Subnetze, NAT, DNS, DHCP) • Planung und Konfiguration eines kleinen Routernetzwerks (statische und Default-Routen, RIPv2) • WLAN-Grundkenntnisse und -Konfiguration (WLAN-Standards, SSID, Accesspoint, Kanäle, WEP, WPA, WPA2) • kurze Einführung in Grundbegriffe der Netzwerksicherheit (Bedrohungen, Vorgehen zur Entwicklung einer umfassenden Security Policy, Abwehrmaßnahmen) • Planung und Konfiguration von einfachen nicht redundanten seriellen WAN-Verbindungen 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzveranstaltungen in Form von Praktika				
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Mathematik für Informatiker*innen 1				
6	Prüfungsformen Klausur				

7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Kappes, Martin: <i>Netzwerk- und Datensicherheit</i> ; Teubner Verlag Badach, Anatol; Hoffman, Erwin: <i>Technik der IP-Netze</i> ; Hanser Washburn, Kevin; Evans, Jim: <i>TCP/IP</i> ; Addison-Wesley Barth, Wolfgang: <i>Das Firewall Buch</i> , SuSE Press RFCs nach Ankündigung in der Vorlesung

Grundlagen der objektorientierten Programmierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	3. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden können Entwicklungsumgebungen einsetzen, um Programme zu erstellen. Sie kennen die Werkzeuge der einzelnen Arbeitsschritte zur Programmerstellung sowie die benötigten Systemkomponenten und sind somit in der Lage, für spezifische Anforderungen Programme zu erstellen.</p> <p>Sie können die Grundprinzipien der objektorientierten Programmierung in eigenen Programmen anwenden und Applikationen mit gegebenen Entwurfsmustern – unter Verwendung von Klassenhierarchien und Vererbungen – entwerfen.</p> <p>Sie beherrschen das Fachvokabular der Objektorientierung und können theoretische Konzepte wie Vererbung und Sichtbarkeit erklären und begründen, welche Vorteile gegenüber rein prozeduraler Programmierung bestehen.</p> <p>Die Studierenden sind in der Lage, auftretende Probleme beim Entwurf und bei der Programmierung im Team gemeinsam zu lösen, die Ergebnisse zu erläutern, zu demonstrieren und zu verteidigen. Sie können erhaltene Hinweise in ihre Lösungen einarbeiten.</p> <p>Die Studierenden erlangen die Kompetenzen der Analyse- und Synthesefähigkeit. Sie sind in der Lage, Wissen selbständig zu erarbeiten, aufzubereiten und weiterzugeben. Die kommunikativen Kompetenzen werden durch Teamarbeit und Forenkommunikation gestärkt.</p>				
3	<p>Inhalte</p> <p>Unter Verwendung der Programmiersprache Java werden folgende Inhalte vermittelt, demonstriert und erprobt:</p> <ul style="list-style-type: none"> • Objektorientierte Modellierung <ul style="list-style-type: none"> • Akteure / Rollen / Auftraggeber-Auftragnehmer-Prinzip • Klassen / Objekte / Schnittstellen / Pakete • Modularisierung / Abstraktion / Vererbung • Vielgestaltigkeit (Polymorphismus) / Kapselung • Objektorientierte Implementierung <ul style="list-style-type: none"> • Abstrakte und geschachtelte Klassen • Schnittstellen mit Default-Implementierungen • Funktionale Schnittstellen und Lambda-Ausdrücke • Generische Klassen und Schnittstellen • Klassen und Schnittstellen für Sammlungen (Set / List / Map) • Klassen für Ausnahme- und Fehlerbehandlungen • Klassen und Schnittstellen für Nebenläufigkeit (Multithreading) • Klassen für Datentransfers 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Modul „Imperative und Prozedurale Programmierung“
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Rainald Schöneberg
12	Sonstige Informationen Literaturlauswahl (jeweils in der aktuellen Auflage): D. Abts: <i>Grundkurs JAVA</i> , Gabler C. Ullenboom: <i>Java ist auch eine Insel</i> , Rheinwerk Computing, (http://openbook.rheinwerk-verlag.de/javainsel/) Oracle Java Dokumentationen (https://docs.oracle.com/en/java/javase)

Pflichtmodule des 4. Semesters

Software Engineering					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden sind in der Lage, Softwareprodukte systematisch zu erstellen. Dazu haben sie sich theoretisch und praktisch in gängige Techniken des Software Engineering eingearbeitet. Sie können insbesondere im direkten Gespräch mit Nicht-Informatiker*innen eine erste Anforderungsanalyse erstellen. Sie beherrschen die Modellierung in UML und die Umsetzung mit einer objektorientierten Programmiersprache. Dazu gehört auch, unter verschiedenen Software-Architekturmodellen das oder die passenden Modelle begründet auszuwählen.</p> <p>Über die technischen Fähigkeiten hinaus haben die Studierenden Projektmanagement-Kompetenzen (u.a. klassisch und agil) erworben und sich auch mit den Themen Dokumentation und Systemtest befasst.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Überblick über Historie der Softwareentwicklung: Spiralmodell, V-Modell und Prototyping-Ansätze • Erstellung objektorientierter Software • Anwendung der Unified Modeling Language • Patterns • Entwicklungstechniken (Entscheidungsbäume, -tabellen) • Architekturmodelle für Software • Dokumentation • Systematisches Testen • Vorgehensmodell mit <ul style="list-style-type: none"> • Anforderungsanalyse • Problembereichsanalyse • iterativ inkrementelle Komponentenentwicklung • Systemtest • Einführung Projektmanagement 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum</p>				

5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Grundlagen der Programmierung, Imperative und Prozedurale Programmierung, Grundlagen oder objektorientierten Programmierung, Datenbanken 1
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Doga Arinir
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Balzert, Helmut: <i>Lehrbuch der Software-Technik, Entwurf, Implementierung, Installation und Betrieb</i> ; Spektrum Akademischer Verlag Balzert, Helmut: <i>Lehrbuch des Software-Technik, Basiskonzepte und Requirements Engineering</i> , Spektrum-Verlag Heide Balzert; <i>Lehrbuch der Objektmodellierung</i> ; Spektrum Verlag Joachim Goll: <i>Methoden und Architekturen der Softwaretechnik</i> , Vieweg + Teubner Joachim Goll: <i>Architektur- und Entwurfsmuster der Softwaretechnik</i> , Springer Vieweg Ian Sommerville; <i>Softwareengineering</i> ; Pearson Studium Chris Rupp; <i>Requirements Engineering</i> ; Hanser Fachbuch

Rechnerarchitektur					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden entwickeln ein Verständnis für die Vorgänge, die in einem einfachen Prozessor mit skalarer oder superskalarer Pipeline bei der Verarbeitung von Maschinencode ablaufen. Zu diesem Zweck haben sie auch Grundkenntnisse in Assemblerprogrammierung für eine Maschinenarchitektur erworben.</p> <p>In einer virtuellen Maschine haben die Studierenden die Tool-Chain für die Assembler-Entwicklung und die Integration von Assembler- und C-Code an praktischen Beispielen eingeübt.</p> <p>Für einfache Architekturen und konkrete Beispielprogramme können die Studierenden angeben, welche Abhängigkeiten zwischen Instruktionen der Prozessor bei der Ausführung beachten muss und welche Auswirkungen auf die Performance das hat.</p> <p>Ein über den Kurs hinausgehendes Kompetenzziel ist, das Bewusstsein dafür zu entwickeln, dass Anwendungen, die in einer Hochsprache entwickelt wurden, letztlich in Form von Maschinencode durch die CPU verarbeitet werden.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Rechneraufbau: Computer-Kategorien, Hauptplatine, Busse für Erweiterungskarten, Prozessor, Chipsatz, Beispiele (Ansteuerung von Tastatur und Festplatte), Geschichte (von-Neumann- und Harvard-Architektur) • Assembler-Programmierung (praktisch): x86-Assembler, Register, Arithmetik und Logik, Speicherzugriff, Variablen, Sprungbefehle, Stack, Unterprogramme, Integration C/Assembler • Assembler-Sprachen (theoretisch): Adressierungsarten, Befehlssatzarchitektur • Performance-Steigerung: Pipelining, RISC/CISC, Pipeline-Hemmnisse, Superskalare Architekturen, Reorder Buffer • Parallel Computing: Cluster, Message Passing, Shared Memory, Grid Computing, Cloud, * as a Service (*aaS), verteilte Dateisysteme 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: –</p>				

6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Hennessy, John L., Patterson, David A.: <i>Computer Architecture: A Quantitative Approach</i> , Morgan Kaufmann Publishers Böttcher, Axel: <i>Rechneraufbau und Rechnerarchitektur</i> , Springer examen.press

Studienrichtung Anwendungsentwicklung und Künstliche Intelligenz

Datenbanken 2					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Studierende haben ihre im Modul „Datenbanken 1“ erworbenen Kenntnisse erweitert und beherrschen nun die Datenmodellierung, insbesondere den Einsatz von Verbundanweisungen und Unterabfragen; in SQL formulieren sie anspruchsvolle (z. B. mengenbasierte) Anweisungen.</p> <p>Sie haben erste Erfahrungen in der Nutzer-/ Rollen- und Rechteverwaltung eines Datenbanksystems und können Sichten (Views) erstellen.</p> <p>Die Studierenden haben außerdem die Programmierung innerhalb von Datenbanksystemen anhand von Stored Procedures, Triggern und Transaktionen praktisch eingeübt.</p>				
3	Inhalte <ul style="list-style-type: none"> • Modellierung von Datenbanken • Normalisierung • mengenalgebraische Grundlagen und Verbundanweisungen • Unterabfragen • skalare Unterabfragen, Listenabfragen • Unterabfragen in Ergebnisspalten • Unterabfragen als Datenquellen • Zusammenhang Unterabfragen und Gruppierungen • Unterabfragen zur Wertebereichseinschränkung • mengenmäßige Weiterverarbeitung von Selektionsergebnissen • Umgang mit Benutzerrechten in Form von Einzelberechtigungen und Benutzerrollen • Benutzersichten auf Datenbestände • Umgang mit Stored Procedures • Verwendung von Triggern 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				

5	Teilnahmevoraussetzungen Formal: grundlegende Kenntnisse relationaler Datenbanken und SQL nachgewiesen durch eine Prüfungsleistung, z.B. eine Klausur Inhaltlich: Grundkenntnisse in Java
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Uwe Klug
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): R. Elmasri, S. B. Navathe, <i>Grundlagen von Datenbanksystemen</i> , Addison-Wesley Heuer, G. Saake, <i>Datenbanken: Konzepte und Sprachen</i> , Internat. Thomson Publ. M. Klettke, H. Meyer; <i>XML & Datenbanken</i> ; dpunkt.verlag U. Klug; <i>Datenbankanwendungen entwerfen & programmieren</i> G.Lausen; <i>Datenbanken</i> ; Spektrum Akademischer Verlag G.Saake, K.-U. Sattler; <i>Datenbanken & Java</i> ; dpunkt.Verlag

Studienrichtung Systemintegration

Rechnernetze 2					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Aufbauend auf dem Modul „Rechnernetze 1“ haben die Studierenden im Modul „Rechnernetze 2“ die theoretischen und praktischen Kompetenzen erworben, die nötig sind, um die Netzwerkinfrastruktur eines mittelständischen Unternehmens mit mehreren weltweit verteilten Standorten bedarfsgerecht zu planen, zu betreiben und weiterzuentwickeln.</p> <p>Sie können geroutete Netzwerke planen und konfigurieren, und sie haben an praktischen Beispielen die Möglichkeiten zur Fehlersuche und -behebung eingeübt. Sie sind auch für Sicherheitsaspekte in diesem Umfeld sensibilisiert.</p> <p>Für die Anbindung entfernter Standorte haben die Studierenden sich praktisch mit WAN- und VPN-Verbindungen nach verschiedenen Protokollen beschäftigt. Sie können nach Analyse der konkreten Anforderungen in einem Beispielunternehmen begründete Vorschläge für den Technologieeinsatz machen.</p>				
3	Inhalte <ul style="list-style-type: none"> • Konfiguration von Switchen und der Kommunikation zwischen Switchen (VLAN, Trunk, Port Security, Spanning Tree) • Entwicklung und Umsetzung des IP-Adressen-Konzepts für ein mittelständisches Unternehmen (VLSM, CIDR, Route-Aggregation, IPv4, IPv6) • Planung, Konfiguration und Fehlersuche in gerouteten Netzwerken (OSPF) • Planung, Implementierung und Fehlersuche von NAT und Access Control Listen für einen Unternehmensstandort 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzveranstaltungen als Praktika				
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Rechnernetze 1				
6	Prüfungsformen Klausur				

7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen

Pflichtmodule des 5. Semesters

Webentwicklung Frontend					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden können auf der Basis clientseitiger Technologien (HTML, CSS, Javascript) einfache Webanwendungen erstellen und dabei existierende, einfache REST-APIs einbinden.</p> <p>Sie verstehen die Unterschiede zwischen einer Webanwendung und einer klassischen Anwendung und können die Vor- und Nachteile einer Webanwendung für einen konkreten Anwendungsfall beurteilen.</p> <p>Die Studierenden kennen die Funktionsweise von Single-Page-Anwendungen und verstehen das dabei notwendige Zusammenwirken von Front- und Backend. Darüber hinaus können sie Tools für das Testen und das Debugging von Frontendkomponenten verwenden.</p>				
3	Inhalte <ul style="list-style-type: none"> • HTML • Cascading Style Sheets • Javascript • Frameworks wie z.B. jQuery, bootstrap, Angular, React • Authentisierungsverfahren in Webanwendungen • Anbindung von REST-APIs per http 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: -				
6	Prüfungsformen Praktische Prüfungen				
7	Prüfungsvorleistung -				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				

9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Duckett, John, HTML & CSS: <i>Erfolgreich Websites gestalten und programmieren</i> , Weinheim SELFHTML-Redaktion. SELFHTML - HTML-Dateien selbst erstellen, http://de.selfhtml.org

Grundlagen Betriebssysteme					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden lernen die theoretischen Grundlagen der Betriebssysteme kennen und erkennen, dass verschiedene Teilkomponenten eines Betriebssystems Algorithmen und Strategien umsetzen, wodurch Betriebssysteme für verschiedene Anwendungsfelder besser oder schlechter geeignet sind.</p> <p>Studierende können für Prozesse und Threads die wichtigsten Standardzustände erklären und können erläutern, wann und wie ein Zustandswechsel erfolgt. Sie können Beispiele für Synchronisationsprobleme konstruieren und die gängigen Synchronisationsmechanismen (Mutexe, Semaphore) theoretisch und praktisch einsetzen, um diese Probleme zu beheben. Im praktischen Bereich haben die Studierenden Erfahrungen mit Prozess- und Thread-Erzeugung sowie Synchronisation in C-Programmen unter Linux gesammelt.</p> <p>Die Studierenden können verschiedene Scheduler (genauer: Scheduling-Strategien) unterscheiden und erklären, wie diese funktionieren. Sie können grob erklären, welche Schritte der Scheduler beim „Umschalten“ von einem Prozess zum anderen (dem so genannten context switch) abarbeiten muss.</p> <p>Durch praktische Übungen werden die theoretischen Konzepte als unmittelbar für den Alltag eines Software-Entwicklers relevant erfahren; die Studierenden können erläutern, wie der Einsatz von Standardmethoden (etwa für Dateizugriff) über verschiedene Ebenen von Bibliotheken und System Calls zu einer Nutzung von Betriebssystemdiensten führt.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Geschichte und Grundlagen: BIOS-Funktionen, Speichern auf Datenträgern, Multitasking • Prozesse: Prozessbegriff, Prozesskontrollblock, Kapselung, Speichernutzung, Prozessbäume, Parent-Child-Beziehung zwischen Prozessen, Zustandsübergänge, Prozessverwaltung (Unix), Priorität, User Mode vs. Kernel Mode, System Calls • Scheduling: Anforderungen an Scheduler, kooperatives und unterbrechendes (präemptives) Scheduling, Verfahren (First Come First Served, Shortest Job First, Shortest Remaining Time, Round Robin, Prioritäten-Scheduler, Lotterie-Scheduler, Multi-Level-Verfahren), Context Switch • Multi-Threading: Kernel- vs. User-Level-Threads, Thread-Programmierung (Java, C++, POSIX/C) • Synchronisation: Race Conditions, kritische Bereiche, gegenseitiger Ausschluss, Locking, Test-and-set-lock, Mutexe, Semaphore, Barrieren, Prozesskommunikation (Pipes, Shared Memory, Signale) • Deadlocks: Ressourcentypen, Deadlock-Bedingungen, Erkennung, Vermeidung • Dateisysteme: Geräte, Partitionen, Interrupt-Behandlung, virtuelles Dateisystem, Treiberhierarchie, Standard-System-Calls (Unix) • Speicherverwaltung: feste und dynamische Partitionierung, Segmentierung, Relokation, Paging, Shared Memory, Copy-on-Write, Auslagerungsspeicher 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturlauswahl (jeweils in der aktuellen Auflage): Tanenbaum, Bos: <i>Moderne Betriebssysteme</i> , Pearson Studium Ehes et al.: <i>Systemprogrammierung in UNIX/Linux - Grundlegende Betriebssystemkonzepte und praxisorientierte Anwendungen</i> , Vieweg+Teubner Verlag Stallings: <i>Operating Systems: Internals and Design Principles</i> , Prentice Hall Silberschatz, Galvin, Gagne: <i>Operating System Concepts</i> , Wiley Eßer, Hans-Georg und Freiling, Felix: <i>The Design and Implementation of the Ulix Operating System</i>

Studienrichtung Anwendungsentwicklung

Design Patterns in der objektorientierten Programmierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden verstehen die Bedeutung von Design Patterns als Hilfsmittel, mit denen sich häufig auftretende Problemstellungen in der Softwareentwicklung lösen lassen, und sie können wichtige, in der objektorientierten Programmierung häufig verwendete Design Patterns in ihrer Struktur und Logik wiedergeben. Im Praktikum lernen sie diese Design Patterns an konkreten Beispielen kennen.</p> <p>Sie können für die selbständige Lösung von konkreten Problemstellungen für die Auswahl eines oder mehrerer Design Patterns argumentieren und diese im Anschluss in der Programmiersprache Java einsetzen und implementieren.</p> <p>Nach Abschluss der Veranstaltung verfügen die Studierenden über ein „Vokabular“ an Design Patterns, so dass sie in Standardsituationen schnell darauf zurückgreifen können. Damit steht ihnen mehr Zeit für die Lösung der neuartigen Probleme in einem größeren Projekt zur Verfügung.</p>				
3	Inhalte <ul style="list-style-type: none"> • Wiederkehrende Problemstellungen in der Objektorientierten Programmierung • Klassendiagramme als Mittel zur Darstellung eines Entwurfsmusters • Erzeugungsmuster (Factory Pattern, Singleton) • Strukturmuster (Adapter, Facade, Decorator, Proxy) • Verhaltensmuster (Iterator, Observer, Command, Memento) • Umsetzung von Design Patterns in Java 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: Grundlagen der Objektorientierten Programmierung				
6	Prüfungsformen Hausarbeit mit Fachvortrag				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): E. Gamma, R. Helm, R. Johnson, J. Vlissides: <i>Design Patterns – Elements of Reusable Object-Oriented Software</i> . Addison-Wesley, Reading MA u. a.

Studienrichtung Systemintegration und Künstliche Intelligenz

Skriptsprachen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden beherrschen die grundlegenden imperativen und objektorientierten Konstrukte der Programmiersprache Python und können damit eigenständig Lösungen für einfache Automatisierungsaufgaben entwickeln. Ergänzend haben sie auch neuere Programmierparadigmen (insbesondere funktionale und aspektorientierte Programmierung) kennengelernt.</p> <p>Sie haben den Umgang mit regulären Ausdrücken eingeübt, können einfache Textfilter erstellen und haben gelernt, die objektorientierte Methodik – etwa für die Nutzung von Bibliotheksmodulen – einzusetzen.</p> <p>Sie kennen ausgewählte Bibliotheken der Standardbibliothek, können komplexere Anwendungsbeispiele nachvollziehen und erlangen die Kompetenz diese auf neue Problemstellungen zu übertragen. Dazu gehört auch der Zugriff auf Datenbanken und das Erstellen von CGI-Skripten.</p> <p>Die Praktikumssteilnahme befähigt zur selbständigen Durchführung einer schriftlichen Ausarbeitung und zur Erarbeitung weiterer Programmierkenntnisse.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Grundlagen von Skriptsprachen, Unterschiede zu kompilierten Sprachen • Python Crashkurs • Programmierumgebungen und Werkzeuge • Zusammengesetzte Datentypen • Kontrollstrukturen und Funktionen • Objektorientierung • Funktionale und aspektorientierte Programmierung • Modularisierung • Ausgewählte Themen aus der Standardbibliothek 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: Grundlagen der imperativen und objektorientierten Programmierung</p>				

6	Prüfungsformen Semesterabschließende schriftliche Ausarbeitung
7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Heiner Giefers
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Johannes Ernesti und Peter Kaiser: <i>Python 3 – Das umfassende Handbuch</i> , Rheinwerk Bernd Klein: <i>Einführung in Python 3</i> , Hansa Python Software Foundation: The Python Language Reference, [Online]¶ https://docs.python.org/3/reference/ (Stand 15.05.2019)

Pflichtmodule des 6. Semesters

Webentwicklung Backend					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden können das Backend einer einfachen Webanwendung erstellen und dabei sowohl die funktionalen Aspekte (Umsetzung von Use-Cases, Datenpersistenz) als auch nicht-funktionale Aspekte (Skalierbarkeit, Schutz vor gängigen Angriffen wie XSS und XSRF) angemessen berücksichtigen.</p> <p>Sie verstehen den Einfluss der Anwendungsarchitektur auf Skalierbarkeit und Sicherheit einer Webanwendung und die Bedeutung eines Zonenmodells, können mithilfe von Werkzeugen wie OpenAPI zu einem gegebenen Anwendungsfall ein API entwerfen und es implementieren.</p> <p>Die Studierenden können geeignete Authentisierungsverfahren für eine Webanwendung auswählen und implementieren, und sie können geeignete Strategien für den Test und das Deployment einer Webanwendung entwickeln.</p>				
3	Inhalte <ul style="list-style-type: none"> • Serverseitige Frameworks wie z.B. NodeJS oder PHP • Anbindung von Datenbanken in Webanwendungen • Spezifikation von REST-APIs • Authentisierungsverfahren in Webanwendungen • Test von REST-APIs 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: Webentwicklung Frontend Inhaltlich: –				
6	Prüfungsformen Praktische Prüfung				
7	Prüfungsvorleistung -				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				

9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Springer, Sebastian: Node.js: <i>das Praxisbuch</i> , Bonn Bojinov, Valentin: RESTful web API design with Node.js, Birmingham

IT-Projektmanagement					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden kennen die grundlegenden Aufgaben und Methoden des IT-Projektmanagements. Sie besitzen die theoretischen und praktischen Kompetenzen um sich in einem Projekt zu orientieren und können konstruktiv in einem Projekt mitarbeiten. Darüber hinaus haben die Studierenden das theoretische Wissen um eine Projektleitung auszuüben. Die Studierenden kennen die wesentlichen Elemente einer Projektinfrastruktur zur Softwareentwicklung, könnend selbige bedarfsgerecht planen, installieren, konfigurieren und administrieren.</p> <p>In den Übungen sammeln die Studierenden erste praktische Erfahrungen mit dem Einsatz der Techniken und Methoden des Projektmanagements bei der Lösung von Fallbeispielen und Übungsaufgaben.</p> <p>Im Focus des Praktikums stehen der Aufbau und die Erweiterung der Fähigkeiten und Fertigkeiten der Studierenden zur Planung, Installation, Konfiguration und Administration einer Projektinfrastruktur zur Softwareentwicklung (Versionsverwaltung, Konfigurations- und Build-Management, Coding-Standards, Tests, Testabdeckung, Continuous Integration, Bugtracking/ Trouble-Ticket-System, Projektplanungstools zur Term- und Kostenplanung).</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Grundbegriffe Projekt, Projektphasen, Aufgaben des Projektmanagements, Projektmanagementzyklus, Methoden, Instrumente und Werkzeuge des Projektmanagements, Erfolgsfaktoren des Projektmanagements • Besondere Aspekte beim Management von Softwareprojekten • Klassische und agile Vorgehensmodelle zur Abwicklung von Softwareprojekten • Qualitätssicherung in Softwareprojekten • Definitionsphase Machbarkeitsstudie, Projektziele, Lasten- und Pflichtenheft, Projektorganisation, Kick-Off, Anforderungsmanagement, Schätzverfahren für Aufwand und Laufzeit • Planungsphase Arbeitspakete, Projektstrukturplan, Ablaufplan Terminplan, Ressourcenplan Kostenplan, Qualitätsplan • Durchführungsphase Projektcontrolling, Teamführung, Qualitätssicherung, Dokumentation • Projektabschlussphase Abschluss-Präsentation, Abschluss-Bericht, Abschluss-Besprechung, Regelung der Projektnachbetreuung, Auflösung der Projektorganisation, Reflexion • Arbeitsumgebung zur Softwareentwicklung Versionsverwaltung mit Subversion, Build Management mit Maven, Coding Standards, Tests, Testabdeckung, Continuous Integration Bug Tracking 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: – Grundlagen der imperativen und der objektorientierten Programmierung
6	Prüfungsformen Hausarbeit mit Fachvortrag
7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Doga Arinir
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Balzert, Helmut: <i>Lehrbuch der Software-Technik, Band 2</i> ; Spektrum Akademischer Verlag Sommerville, Ian: <i>Software Engineering</i> ; Pearson Beiderwieden, Arndt; Pürling, Elvira: <i>Projektmanagement für IT-Berufe</i> ; Stam Verlag; Troisdorf Heilmann, Heidi; Etzel; Hans-Joachim; Richter, Reinhard: <i>IT-Projektmanagement - Fallstricke und Erfolgsfaktoren</i> ; Dpunkt-Verlag; Heidelberg Mangold, Pascal: <i>IT-Projektmanagement kompakt</i> ; Spektrum Akademischer Verlag; Heidelberg Stoyan, Robert: <i>Management von Webprojekten</i> ; Springer; Berlin Kerzner, Harold; Grau, Nino: <i>Projektmanagement</i> ; Mitp-Verlag; Bonn Patzak, Gerold; Rattay, Günter: <i>Projektmanagement</i> ; Linde; Wien Schreckeneder, Berta: <i>Projektcontrolling</i> ; Haufe; Freiburg im Breisgau DeMarco, Tom, Lister, Timothy: <i>Wien wartet auf Dich. Der Faktor Mensch im DV-Management</i> , Peopleware; Hanser DeMarco, Tom: <i>Der Termin</i> ; Hanser Popp, Gunther: <i>Konfigurationsmanagement mit Subversion</i> , Ant und Maven; dPunkt Verlag (Auszüge als eBook verfügbar) Collins-Sussman, Ben et al: <i>Versionskontrolle mit Subversion</i> ; Veröffentlicht (TBA) (eBook verfügbar unter http://svnbook.red-bean.com/) Casey, John et al: <i>Better builds with Maven, How-to Guide for Maven 2.0</i> ; exist global Library Press (eBook verfügbar) O'Brien, Tim et al: <i>Maven: The Definitive Guide</i> ; Sonatype Inc. (eBook verfügbar)

Graphen, Codes und Automaten					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden erwerben Grundlagenwissen aus den Bereichen Graphentheorie, Kodierung (Kompression und Kryptographie) und Automaten / formale Sprachen. Sie üben den Umgang mit den geläufigen Graphentypen ein, was das Durchführen gängiger Graphenalgorithm (etwa: Breiten- und Tiefensuche, Suche nach kürzesten Wegen) einschließt.</p> <p>Die Studierenden erfahren im Themengebiet Kodierung – auch durch praktische Übungen – u. a., dass und warum kein Code beliebige Datensätze komprimieren kann und warum für eine nicht brechbare Verschlüsselung aufwendige Verfahren und ein tieferer Einstieg in die Mathematik notwendig sind. Sie können für unterschiedliche Kommunikationssituationen begründet ein symmetrisches oder asymmetrisches Verschlüsselungsverfahren auswählen.</p> <p>Formale Sprachen sind über Grammatiken oder über erkennende Automaten definierbar. Die Studierenden lernen beide Darstellungsformen (und deren Äquivalenz) für zwei Sprachklassen kennen und üben den praktischen Umgang damit ein – u. a. auch durch die Arbeit mit regulären Ausdrücken, die ein beliebtes Werkzeug in vielen Programmier- und Skriptsprachen sind.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> Graphentheorie: Gerichtete und ungerichtete Graphen, mit und ohne Gewicht, Inzidenz, Adjazenz, Breitensuche, Tiefensuche, topologisches Sortieren, minimale Spannbäume, Algorithmen von Kruskal und Prim, kürzeste Pfade, Relaxierung, Algorithmen von Bellman-Ford und Dijkstra Datenkompression: Präfixfreie Codes, Shannon-Kodierung, Shannon-Fano-Kodierung, Huffman-Kodierung, erweiterte Huffman-Kodierung, arithmetische Codes, Wörterbuchtechniken, LZW Kryptographie: Grundlagen der Zahlentheorie, Restklassenring $\mathbb{Z}/n\mathbb{Z}$, GGT-Berechnung, modulare lineare Gleichungen, RSA-Verfahren, Signieren und verifizieren Automatentheorie und formale Sprachen: endliche deterministische / nicht-deterministische Automaten, reguläre Ausdrücke, Typ-3-Grammatik, kontextfreie Sprachen, Chomsky-Normalform, Cocke-Younger-Kasami-Algorithmus für das Wortproblem, Kellerautomaten 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: –</p>				
6	<p>Prüfungsformen</p> <p>Klausur</p>				

7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): T. H. Cormen, C. E. Leiserson, R. Rivest, C. Stein: <i>Algorithmen – Eine Einführung</i> , Oldenbourg Verlag, München G. Vossen, K.-U. Witt: <i>Grundkurs Theoretische Informatik</i> , Springer Vieweg R. Socher: <i>Theoretische Grundlagen der Informatik</i> , Carl Hanser Verlag

Studienrichtung Anwendungsentwicklung

Mobile Applikationen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden sind in der Lage, mobile Applikationen auf Basis der gängigen vier Entwicklungsparadigmen zu entwickeln: Mobile Web-Apps, hybride Apps, Cross-Plattform-Apps sowie native Apps.</p> <p>Sie haben insbesondere vertiefte Kenntnisse über die Architektur der Android-Plattform erworben, und sie beherrschen den Umgang mit der Entwicklungsumgebung Android Studio. Den Einsatz von Android-App-spezifischen Elementen wie etwa Intents haben sie eingeübt, und sie können bei Konzeption und Implementierung die Besonderheiten von mobilen Anwendungen berücksichtigen.</p> <p>Im Zuge der Arbeit mit Activity Layout Designs haben die Studierenden auch den Umgang mit der Auszeichnungssprache XML vertieft.</p>				
3	Inhalte <ul style="list-style-type: none"> • Paradigmen für die Entwicklung mobiler Endgeräte • Mobile Web-Anwendungen auf Basis des Ionic- und Vue.js-Frameworks • Native Entwicklung mobiler Anwendungen mit Android <ul style="list-style-type: none"> • Architektur der Android-Plattform • Activities und ihr Lebenszyklus • Views und Layouts • Intents • Menüs und Navigation • Maps (Google und Open Street Map) • Sensoren • Bar- und QR-Codes 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhalt: Grundlagen der Objektorientierten Programmierung				

6	Prüfungsformen Semesterabschließende schriftliche Ausarbeitung
7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Doga Arinir
12	Sonstige Informationen -

Studienrichtung Systemintegration

Unix-artige Betriebssysteme					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden erwerben praktische Fertigkeiten im Bereich der Systemadministration (insbesondere Prozessverwaltung und Dateisysteme) und der Shell-Programmierung auf Unix-Systemen. Sie haben im Rahmen der Lösung konkreter Verwaltungsaufgaben u. a. prozedurale Standardelemente der Programmierung (Variablen, Schleifen, Fallunterscheidung, Unterprogramme) wiederholt und im Zusammenhang von Shell-Skripten eingeübt.</p> <p>Damit verbunden ist auch, sich über die unterschiedlichen Ansätze bei der Anwendungsentwicklung und der Shell-Programmierung bewusst zu werden und z. B. „einen schnellen Hack“ im Umfeld der Systemadministration anders als bei der Anwendungsentwicklung zu bewerten.</p> <p>Studierende sind sicher im Umgang mit der Shell auf einem Linux-System (Debian) und können fehlende Software nachinstallieren. Sie kennen diverse Standardtools der Linux-/Unix-Welt und sind mit dem Baukasten-Paradigma vertraut, nach dem spezialisierte, monothematische Werkzeuge durch geschickte Kombination (Shell-Skripte, Pipelines) zu einer Problemlösung zusammengebaut werden.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Tools für die Systemadministration unter Linux und Grundlagen der Linux-Standard-Shell (Bash) • Die Inhalte werden anhand von konkreten Problemen und Problemlösungen vorgestellt, wobei sich neue syntaktische Elemente (etwa: Schleifen) und der Einsatz neuer Tools jeweils aus der Aufgabenstellung ergeben • Shell-Grundlagen: Interaktive Shell-Sitzung vs. Shell-Skript, Datei- und Verzeichnisoperationen, History, Shell- und Umgebungsvariablen, Umleitungen, Pipes, Manpages, Editor vi, Administrator root • Shell-Programmierung: Strings, Arrays, Dictionaries, Wildcards, Schleifen, Vergleichsoperationen, Fallunterscheidungen, reguläre Ausdrücke, Funktionen, Parameter, Auswertung von Rückgabewerten, FIFO-Dateien • Dienste und wichtige Tools: SSH, cron, grep, sed, md5sum, awk, inotifywait, apt (Debian) • Unix-Dateisysteme: Benutzer, Gruppen, Zugriffsrechte (symbolische und Oktalnotation), SUID- und GUIT-Bits, ACLs, Partitionierung, Formatierung, Einbinden (Mounten), Journaling, Image- und Swap-Dateien, Verschlüsselung (blockgerät- und verzeichnisbasiert), Zugriff auf fremde Dateisysteme • Prozesse unter Unix: Jobkontrolle, Vorder- und Hintergrund, Loslösen eines Jobs von der Shell, Prozessliste, Baumstruktur, Speicherbelegung, Linux-/proc-Dateisystem, Signale, Prozessgruppen und Sessions 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				

5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Kofler, Michael: <i>Linux – das umfassende Handbuch</i> , Rheinwerk Computing Eßer, Hans-Georg und Dölle, Mirko: <i>Linux-Grundlagenbuch</i> , Data Becker

Studienrichtung Künstliche Intelligenz

Einführung Machine Learning					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	6. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden haben sich intensiv mit mathematischen Grundlagen der Statistik beschäftigt und können insbesondere die lineare Regression zur Regressionsanalyse erläutern und praktisch einsetzen.</p> <p>Sie kennen die grundlegenden Machine-Learning-Methoden zur Regression und Klassifikation und können diese im Gebiet der künstlichen Intelligenz einordnen.</p> <p>Für einfache Problemstellungen können sie geeignete Machine-Learning-Verfahren auswählen, Lösungsansätze entwickeln und diese mithilfe der Programmiersprache Python sowie entsprechender Bibliotheken umsetzen und evaluieren. Hierzu gehört auch, dass sie ein gutes Überblickswissen zur Verfügbarkeit von ML-Bibliotheken für Python erworben haben und abhängig von den Anforderungen begründet eine Auswahl der einzusetzenden Bibliotheken vornehmen können.</p>				
3	Inhalte <ul style="list-style-type: none"> • Übersicht über das Themenfeld KI und Einordnung der Methoden des Machine Learnings • Python-Bibliotheken für numerisches Rechnen, Verarbeitung strukturierter Daten, Maschinelles Lernen und Visualisierung • Grundlagen der Statistik • Lineare Regression • Trainieren von Machine-Learning-Modellen • Binäre und multinomiale logistische Regression • Entscheidungsbäume • Neuronale Netzwerke 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: Skriptsprachen (Python)				
6	Prüfungsformen Klausur				

7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Heiner Giefers
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): <i>Aurélien Géron: Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow. O'Reilly</i> <i>Sebastian Raschka, Vahid Mirjalili: Machine Learning mit Python und Scikit-Learn und TensorFlow. mitp</i> <i>Jörg Frochte: Maschinelles Lernen: Grundlagen und Algorithmen in Python, Hansa</i> <i>Tariq Rashid: Neuronale Netze selbst programmieren: Ein verständlicher Einstieg mit Python, O'Reilly</i>

Pflichtmodule des 7. Semesters

Softwarequalität					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden kennen verschiedene Verfahren zur Beurteilung und Sicherung der Qualität von Software und können bewerten, welche Verfahren in einem konkreten Fall effizient eingesetzt werden können.</p> <p>Sie können argumentieren, was aus Sicht der Qualitätssicherung beachtet werden muss, um ein Software-Entwicklungsprojekt erfolgreich durchzuführen, und welcher Testansatz unter gegebenen Randbedingungen am erfolgversprechendsten für ein Projekt ist.</p> <p>Sie kennen unterschiedliche Techniken und algorithmische Ansätze für die Planung von Qualitätssicherungsmaßnahmen und die Gestaltung, Ausführung und Auswertung von Tests. Das beinhaltet Funktions-, Performance- und Lasttests und auch den Einsatz von Testautomatisierungstools.</p> <p>Sie sind in der Lage, für verschiedene Arten von Software eine passende Art der Qualitätssicherung und die einzusetzenden Werkzeuge auszuwählen und zu nutzen.</p>				
3	Inhalte <ul style="list-style-type: none"> • Qualität von Software • Testebenen und -stufen im Softwareentwicklungsprozess • Werkzeuge für Unit-Tests • Überdeckungsmaße • Mocking • Test von Nutzungsoberflächen • Performance- und Lasttests • Testautomatisierung • Organisation des QS-Prozesses in IT-Projekten 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Software Engineering				
6	Prüfungsformen Portfolio				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Hoffmann, Dirk; <i>Software-Qualität</i> , Berlin Liggesmeyer, Peter; <i>Software-Qualität : Testen, Analysieren und Verifizieren von Software</i> , Heidelberg

Konferenzseminar					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen Aufaktveranstaltungen: 4 h Seminar: 4 h	Kontaktzeit 8 h	Selbststudium 117 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden arbeiten sich selbständig anhand wissenschaftlicher Literatur in ein vorgeschlagenes oder nach Absprache frei gewähltes Thema aus einem von Semester zu Semester wechselnden Teilgebiet der Informatik ein, erstellen dazu eine schriftliche Ausarbeitung, bereiten einen Seminarvortrag vor und halten ihn. Sie stärken somit u.a. neben ihrer kommunikativen auch ihre Selbstkompetenz.</p> <p>Zu den hier zu erwerbenden Kompetenzen zählen die Literaturrecherche, korrektes Zitieren, die Fähigkeit auszuwählen, welche Aspekte in der Ausarbeitung und im Vortrag behandelt werden, zielgruppengerechtes Schreiben sowie der Umgang mit dem Textsatzsystem LaTeX.</p> <p>Das Seminar findet als „Konferenzseminar“ statt: Die Studierenden üben auch das Prozedere ein, das beim Einreichen einer wissenschaftlichen Arbeit bei einer Konferenz üblich ist: Unter anderem lernen sie, die Arbeiten anderer Personen im Review-Prozess zu beurteilen und Kritik und Verbesserungsempfehlungen auszusprechen sowie für die eigene Arbeit anzunehmen und umzusetzen.</p> <p>Im Bereich des zivilgesellschaftlichen Engagements (Bedeutung der Informatik für die Gesellschaft) haben die Studierenden zu den bearbeiteten Forschungsthemen auch gesellschaftliche Risiken diskutiert, die mit dem Einsatz bekannter und neuer Verfahren der Informatik verbunden sind.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Wissenschaftliches Arbeiten, Zitieren, Plagiat • LaTeX-Grundlagen • Abläufe einer wissenschaftlichen Konferenz • Aktuelle Themen der Informatik (mit wechselndem Themenschwerpunkt) • Chancen und Risiken der behandelten Technologien 				
4	<p>Lehr- und Lernformen</p> <p>Seminar</p> <p>Selbststudium (Recherche und Lektüre wissenschaftlicher Literatur)</p> <p>Präsenzlehre als seminaristischer Unterricht</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: –</p>				
6	<p>Prüfungsformen</p> <p>Portfolio oder Klausur</p>				
7	<p>Prüfungsvorleistung</p> <p>-</p>				

8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literatur (jeweils in der aktuellen Auflage): Balzert, Helmut et al.: <i>Wissenschaftliches Arbeiten: Ethik, Inhalt & Form wiss. Arbeiten, Handwerkszeug, Quellen, Projektmanagement, Präsentation</i> , Springer Nature

Studienrichtung Anwendungsentwicklung

Java Enterprise Edition					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden erwerben fortgeschrittene Kenntnisse zur Erstellung professioneller Unternehmensanwendungen mit Java. Sie verstehen insbesondere, welche besonderen Anforderungen an Unternehmensanwendungen gestellt und wie diese mit Hilfe von gängigen Frameworks umgesetzt werden. Die Veranstaltung vermittelt wichtige Entwurfsmuster aus der Jakarta EE sowie Spring/Spring Boot-Spezifikation und geht auf bestimmte Aspekte, wie zum Beispiel die Datenhaltung über die Jakarta Persistence API (JPA), genauer ein. Darauf aufbauend werden REST-API und Mikroservice-Architekturen eingeführt und eine rudimentäre Umsetzung von Single-Page-Web-Anwendungen diskutiert.</p> <p>Teilnehmer*innen sollten am Ende dieser Veranstaltung mehrschichtige Unternehmensanwendungen (Präsentationsschicht, Serviceschicht, Logik und Datenhaltung) unter Verwendung aktueller Frameworks realisieren können.</p>				
3	Inhalte <ul style="list-style-type: none"> • Aufbau von Unternehmensanwendungen • Einführung Spring und Spring Boot • Das MVC-Entwurfsmuster • Dependency Injection • Jakarta Persistence API • REST-APIs • Spring Security 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: Grundlagen der Objektorientierten Programmierung, Webentwicklung Frontend, Datenbanken 1				
6	Prüfungsformen Semesterabschließende schriftliche Ausarbeitung				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Doga Arinir
12	Sonstige Informationen -

Studienrichtung Systemintegration

Microsoft-Betriebssysteme					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden sind in der Lage, Netzwerklösungen für kleine, mittelständische und große Unternehmen zu entwerfen. Sie können dazu geeignete Microsoft Server auswählen. Sie installieren oder aktualisieren Microsoft-Server und können diese als Datei- und Druck-Server einrichten.</p> <p>Zu den fortgeschrittenen administrativen Aufgaben, die die Studierenden erlernen, gehört das Konfigurieren von Microsoft-Servern als Domaincontroller sowie das Verwalten und Überwachen von Microsoft-Servern und das Absichern eines Netzwerks mit Microsoft-Servern.</p> <p>Die Aufgaben erledigen die Studierenden mit der Microsoft Management Console (MMC), mit deren Bedienung sie durch umfangreiche praktische Übungen vertraut werden.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Installation von Microsoft-Servern, Umgang mit der MMC • Benutzerverwaltung, Computerverwaltung • DHCP, DNS • ADS, Gruppenrichtlinien • Internet- und Remote-Verbindungen • Lizenzverwaltung • Verwalten von Clientcomputern, Benutzer-Gruppen und Remote-Arbeitsplätzen • Datei- und Druckserver • Domaincontroller • Serversicherheit: Internetzugriff mittels NAT, Remotezugriff, VPN, Absichern von Datei-, Ordner- und Druckobjekten, Virenschutzmaßnahmen, Schutz vor Datenverlusten • Domain, Forest • Design einer Messaging-Infrastruktur • Exchange-Server verwalten und konfigurieren 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Hans-Georg Eßer
12	Sonstige Informationen Literaturlauswahl (jeweils in der aktuellen Auflage): Craig Zacker: <i>Installing and Configuring Windows Server 2012 R2 Exam 70-410</i> , Microsoft Official Academic Course, Wiley

Studienrichtung Künstliche Intelligenz

Deep Learning					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden kennen die theoretischen Grundlagen künstlicher neuronaler Netze und verstehen, wie man künstliche neuronale Netze in der Praxis verwendet.</p> <p>Sie sind in der Lage, Aufgaben wie die Erkennung von Ziffern oder die Klassifikation von Bildern mithilfe von Deep-Learning Bibliotheken selbstständig zu lösen. Hierzu gehört auch die Fähigkeit, die passenden Methoden und Parameter auszuwählen und diese zu optimieren.</p> <p>Außerdem können die Studierenden die Güte von Modellen bewerten und unterschiedliche potenzielle Qualitätsprobleme benennen sowie deren Ursachen erläutern.</p> <p>In praktischen Übungen setzen die Studierenden u. a. die MNIST-Datenbank mit handgeschriebenen Ziffern und die ImageNet-Datenbank ein, um die Leistungsfähigkeit von Deep Learning nachvollziehen zu können. Dies ermöglicht es ihnen, begründet für oder gegen den Einsatz von Deep Learning in einem Softwareprojekt zu argumentieren.</p>				
3	Inhalte <ul style="list-style-type: none"> • Deep Learning und Künstliche neuronale Netze (KNN) • Trainingsalgorithmen für KNN • Bewertungsverfahren für die Modellgüte • Spezielle Typen von KNN: Convolutional Neural Networks, Recurrent Neural Networks • Training von CNNs • Training von RNNs mit LSTM • Werkzeuge für Deep Learning • Praktische Beispiele: MNIST-Datensatz, ImageNet 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Machine Learning, Skriptsprachen (Python-Kenntnisse)				
6	Prüfungsformen Semesterabschließende schriftliche Ausarbeitung				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) im Verbund M.Sc.-Studiengang Angewandte Künstliche Intelligenz
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Patterson, Josh; <i>Deep learning: a practitioner's approach</i> , O'Reilly Chollet, Francois; <i>Deep Learning with Python</i> Wartala, Ramon; <i>Praxiseinstieg Deep Learning: mit Python</i> Caffe, Tensor; <i>Flow und Spark eigene Deep-Learning-Anwendungen erstellen</i> , O'Reilly

Pflichtmodule des 8. Semesters

Projekt					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	250 h	10 ECTS	ab 8. Semester	Sommersemester und Wintersemester	1 Semester
1	Lehrveranstaltungen Seminar und Vor-Ort Betreuung	Kontaktzeit 16 h	Selbststudium 234 h	geplante Gruppengröße i.d.R. 1 Studierende*r	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden sammeln Erfahrungen in der Anwendung von Kenntnissen aus dem vorangehenden Studium. Die Aufgabenstellung geht dabei deutlich über den Umfang von Praktikumsaufgaben hinaus und bereitet die Studierenden auf die Bachelorarbeit vor. Das Projekt wird in der Regel in einem Unternehmen durchgeführt. Die Studierenden vertiefen ihre Kompetenz im Bereich der Präsentationstechnik.				
3	Inhalte Die Themenauswahl soll sich an der gewählten Studienrichtung orientieren, also im Bereich Anwendungsentwicklung, Systemintegration oder Künstliche Intelligenz liegen. Die Themen legen die Dozenten unter Berücksichtigung der Vorschläge der Studierenden fest.				
4	Lehr- und Lernformen Selbststudium und selbständige praktische Durchführung des Projektes. Im begleitenden Seminar halten die Studierenden mehreren Präsentationen zum Projektstatus.				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Projektarbeit (Vorträge zum Projektstatus und semesterbegleitende Ausarbeitung (siehe FPO)).				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 10/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Klug / Prof. Dr. Rübsam				
12	Sonstige Informationen -				

Studienrichtung Anwendungsentwicklung

3D-Grafikprogrammierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	8. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden kennen den Aufbau der Rendering Pipeline. Sie verstehen die Notwendigkeit für homogene Koordinaten in der 3D-Grafikprogrammierung und sind in der Lage, die mathematischen Transformationen zur Überführung der Koordinaten einer 3D-Szene in die 2D-Bildschirmkoordinaten nachzuvollziehen und auch selbst durchzuführen.</p> <p>Sie kennen einige wesentliche Algorithmen zur Rasterung von Grafikprimitiven sowie die Vorteile des Einsatzes von Szenengraphen und kennen die grundlegenden Beleuchtungsmodelle. Sie können diese hinsichtlich ihrer Implementierungsweise, Vor- und Nachteilen unterscheiden.</p> <p>Die Studierenden beherrschen die Berechnung von Schatten in einer 3D-Szene und die Projektion von Texturen auf Geometrien.</p> <p>Sie können die theoretischen Grundlagen der 3D-Grafikprogrammierung unter Einsatz der OpenGL selbst anwenden.</p>				
3	Inhalte <ul style="list-style-type: none"> • Homogene Koordinaten und lineare Abbildungen • Koordinatensysteme und Transformationen in der Rendering Pipeline • Algorithmen in der Rendering Pipeline, Szenengraphen • Beleuchtungsmodelle • Texturierung von Objekten • Schattenberechnung • Grundlagen der OpenGL 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				

5	Teilnahmevoraussetzungen Formal: - Inhaltlich: Mathematik für Informatiker*innen 1, Mathematik für Informatiker*innen 2 oder die Mathematik vergleichbarer ingenieurwissenschaftlicher Studiengänge, Grundlagen Objektorientierte Programmierung
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Andreas Steins
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Nischwitz, Fischer, Haberäcker, Socher: <i>Computergrafik und Bildverarbeitung – Band 1</i> . Vieweg+Teubner Verlag, Wiesbaden Zeppenfeld: <i>Lehrbuch der Grafikprogrammierung – Grundlagen, Programmierung, Anwendung</i> , Spektrum Verlag, Heidelberg, Berlin Virag: <i>Grundlagen der 3D-Programmierung – Mathematik und Praxis mit OpenGL</i> , Open Source Press, München

Studienrichtung Systemintegration

Virtualisierung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	8. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Den Studierenden sind die verschiedenen Konzepte der Virtualisierung bekannt, und sie können deren Einsatzgebiete sowie die jeweiligen Vor- und Nachteile benennen und damit anforderungsbasiert eine begründete Entscheidung für die passende Technologie treffen.</p> <p>Sie kennen die verschiedenen Systemkomponenten einer in Rechenzentren führenden Virtualisierungslösung und können diese einsetzen, um einfache virtuelle Infrastrukturen aufzubauen, und die Konzepte auf vergleichbare Frameworks anderer Hersteller übertragen.</p> <p>Softwareverteilung mit Hilfe der Containervirtualisierung haben die Studierenden im praktischen Betrieb erprobt, und sie können Vor- und Nachteile im Vergleich zu klassischen Virtualisierungstechniken erläutern.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Motivation und Grundlagen der Virtualisierung • Hypervisor-Technologien • Server-Virtualisierung • Virtualisierung im Rechenzentrum • Hochverfügbarkeit und Fehlertoleranz • Hardwareunterstützung für Virtualisierung • Betriebssystem-Virtualisierung • Virtuelle Maschinen für Programmiersprachen 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: UNIX-artige Betriebssysteme</p>				
6	<p>Prüfungsformen</p> <p>Mündliche Prüfung</p>				
7	<p>Prüfungsvorleistung</p> <p>Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert</p>				

8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Heiner Giefers
12	Sonstige Informationen Literaturlauswahl (jeweils in der aktuellen Auflage): Jim Smith und Ravi Nair: <i>Virtual Machines: Versatile Platforms for Systems and Processes</i> . Morgan Kaufmann Publishers Edouard Bugnion, Jason Nieh, und Dan Tsafir: <i>Hardware and Software Support for Virtualization</i> , Morgan & Claypool Publishers Ralph Göpel: <i>Praxishandbuch VMware vSphere 6.7</i> , O'Reilly VMware, Inc.: <i>vCenter Server und Hostverwaltung</i> VMware, Inc.: <i>vSphere Networking</i> VMware, Inc.: <i>Handbuch zur Verfügbarkeit in vSphere</i> Bernd Öggl, Michael Kofler: <i>Docker – Das Praxisbuch für Entwickler und DevOps-Teams</i> , Rheinwerk

Studienrichtung Künstliche Intelligenz

Natural Language Processing					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	8. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden kennen die Grundlagen des Natural Language Processing (NLP) und die wichtigsten Anwendungsgebiete. Sie können mit großen Textkörpern umgehen, geeignete NLP-Tools auswählen und diese einsetzen.</p> <p>Sie sind in der Lage, NLP-Modelle zu erstellen und deren Resultate quantitativ zu analysieren, zu bewerten und zu interpretieren.</p> <p>Die Studierenden reflektieren die Mehrdeutigkeit und Ungenauigkeit von Text und haben ein Bewusstsein für die Grenzen der Möglichkeiten aktueller NLP-Verfahren (etwa beim Erkennen von Ironie oder rhetorischen Fragen).</p>				
3	<p>Inhalte</p> <p>Moderne Computersysteme verarbeiten zunehmend Daten in natürlicher Sprache, etwa</p> <ul style="list-style-type: none"> • bei der Suche nach Texten im Internet, • in Dialogsysteme mit virtuellen Agenten, • bei der automatischen Übersetzung oder • bei der Informationsextraktion aus Nachrichtenmeldungen oder Beiträgen in sozialen Medien. <p>In diesem Modul werden anhand von praktischen Beispielen ausgewählte aktuelle Themen des Natural Language Processing behandelt, insbesondere aus den Bereichen:</p> <ul style="list-style-type: none"> • Textklassifikation, • Erkennung von Entitäten, • Chatbots, • Topic Modelling und • Word Embeddings 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Praktikum</p>				
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: -</p> <p>Inhaltlich: Machine Learning, Deep Learning, Modul Skriptsprachen (Python-Kenntnisse)</p>				

6	Prüfungsformen Kombinationsprüfung
7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Kurdi, Mohamed; <i>Natural language processing and computational linguistics</i> , Wiley Goldberg, Yoa; <i>Neural Network Methods in Natural Language Processing</i> , Morgan & Claypool Bengfort, Benjamin; <i>Applied Text Analysis with Python</i> , O'Reilly

Pflichtmodule des 9. Semesters

Bachelorseminar					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	ab 9. Semester	Wintersemester und Sommersemester	1 Semester
1	Lehrveranstaltungen Selbststudium zum Thema der Bachelorarbeit (3 SWS) Seminar (1 SWS)	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden erarbeiten selbständig ein (ingenieur-)wissenschaftliches Thema. Sie extrahieren die für die Bachelorarbeit relevanten Informationen und stellen sie dar. Sie transferieren die gewonnenen Erkenntnisse auf das konkrete Thema der Bachelorarbeit und wenden sie darauf an. Die Kompetenz zur Projektplanung und zur Projektsteuerung der Studierenden wird somit vertieft und praktisch erprobt, und das methodische Vorgehen wird reflektiert.				
3	Inhalte Ziel der Lehrveranstaltung ist es die Studierenden bei der strukturierten Durchführung ihrer Bachelorarbeit zu begleiten und zu unterstützen. Im Austausch mit anderen Bachelorstudierenden werden in den regelmäßigen Seminarveranstaltungen folgende Aspekte behandelt: <ul style="list-style-type: none"> • Themenfindung und ableiten der Forschungsfragen/Projektziele • Projektplanung und Zeitmanagement • Informationsbeschaffung / Recherche • Entwicklung des Lösungskonzeptes • Auswahl der Umsetzungsvariante • Umsetzung und Test • Schriftliche Ausarbeitung und Abschlusspräsentation 				
4	Lehr- und Lernformen Seminar				
5	Teilnahmevoraussetzungen Formal: mindestens 153 ECTS gemäß den Anlagen 1 und 2 der FPO, wovon mindestens 60 ECTS in den ersten drei Semestern erworben wurden Inhaltlich: -				
6	Prüfungsformen Portfolio				

7	Prüfungsvorleistung Regelmäßige Teilnahme am Seminar
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) -
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Uwe Klug / Prof. Dr. Michael Rübsam
12	Sonstige Informationen -

Bachelorarbeit					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	300 h	12 ECTS	ab 9. Semester	Wintersemester und Sommersemester	1 Semester
1	Lehrveranstaltungen	Kontaktzeit	Selbststudium	geplante Gruppengröße	
		-	300 h	i.d.R. 1 Studierende*r	
2	Lernergebnisse (learning outcomes) / Kompetenzen				
	<p>Die Kandidatin oder der Kandidat ist befähigt, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus dem Bereich der Anwendungsentwicklung, der Systemintegration oder der Künstlichen Intelligenz selbstständig mit den in der Anwendung erprobten wissenschaftlichen und fachpraktischen Methoden zu bearbeiten und in fachübergreifende Zusammenhänge zu stellen.</p> <p>Durch das erfolgreiche Erstellen der Abschlussarbeit demonstrieren sie die Kompetenzen, die im Laufe des Studiums erworben oder vertieft wurden.</p>				
3	Inhalte				
	<p>Es soll ein in der Regel ein praxisorientiertes Problem aus dem Bereich der Anwendungsentwicklung, der Systemintegration oder der Künstlichen Intelligenz mit den im Studium erlernten wissenschaftlichen und fachpraktischen Methoden in begrenzter Zeit unter Anleitung eines erfahrenen Betreuers gelöst werden. Die Bachelorarbeit ist entweder eine eigenständige Ausarbeitung oder betrachtet ein bekanntes Thema unter neuen Aspekten.</p>				
4	Lehr- und Lernformen				
	Eigenständige Literaturstudien, eigene experimentelle Arbeiten und Untersuchungen, persönliche Beratung durch den/die beteiligte/n Professor*in				
5	Teilnahmevoraussetzungen				
	<p>Formal: mindestens 153 ECTS gemäß den Anlagen 1 und 2 der BPO, wovon mindestens 60 ECTS aus den ersten drei Semestern sein müssen</p> <p>Inhaltlich: -</p>				
6	Prüfungsformen				
	Bachelorarbeit (schriftliche Ausarbeitung)				
7	Prüfungsvorleistung				
	-				
8	Voraussetzungen für die Vergabe von Kreditpunkten				
	bestandene Bachelorarbeit				
9	Verwendung des Moduls (in anderen Studiengängen)				
	-				
10	Stellenwert der Note für die Endnote				
	12/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
	Ein/e betreuender/r Professor*in der Fachhochschule Südwestfalen				
12	Sonstige Informationen				

Kolloquium					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	75 h	3 ECTS	ab 9. Sem.	nach Bedarf	-
1	Lehrveranstaltungen	Kontaktzeit	Selbststudium	geplante Gruppengröße	
		1 h	74 h	i.d.R. 1 Studierende*r	
2	Lernergebnisse (learning outcomes) / Kompetenzen				
	Die Studierenden sind befähigt, die Ergebnisse der Bachelorarbeit, ihre fachlichen Grundlagen, ihre fachübergreifenden Zusammenhänge und ihre außerfachlichen Bezüge mündlich darzustellen und selbstständig zu begründen sowie ihre Bedeutung für die Praxis einzuschätzen.				
3	Inhalte				
	Zunächst wird der Inhalt der Bachelorarbeit aus dem Bereich der Anwendungsentwicklung, Systemintegration oder Künstlichen Intelligenz im Rahmen eines Vortrages präsentiert. Anschließend sollen in einer Diskussion Fragen zum Vortrag und zur Bachelorarbeit beantwortet werden.				
4	Lehr- und Lernformen				
	-				
5	Teilnahmevoraussetzungen				
	<ul style="list-style-type: none"> • min. 153 ECTS in den Pflichtmodulen gemäß Anlage 1 (FPO) • min 12 ECTS in den Wahlpflichtmodulen gemäß Anlage 2 (FPO) • 12 ECTS in der Bachelorarbeit 				
6	Prüfungsformen				
	mündliche Prüfung				
7	Prüfungsvorleistung				
	bestandene Bachelorarbeit				
8	Voraussetzungen für die Vergabe von Kreditpunkten				
	Bestehen der mündlichen Prüfung des Kolloquiums				
9	Verwendung des Moduls (in anderen Studiengängen)				
	-				
10	Stellenwert der Note für die Endnote				
	3/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
	Ein/e betreuende/r Professor*in der Fachhochschule Südwestfalen				
12	Sonstige Informationen				
	-				

Wahlpflichtblock Betriebswirtschaftslehre

Betriebswirtschaftslehre 1					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	3. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden erhalten die betriebswirtschaftlichen Grundkenntnisse, um im Fortgang der weiteren Semester sowohl die Fertigkeit der Entwicklung eines Produktes/Dienstleistung als auch die Umsetzung eines Start-ups durch Ausarbeitung eines Businessplanes zu erlangen				
3	Inhalte <ul style="list-style-type: none"> • Firma, Betrieb, Mittelstand, Hidden champions • Ökonomisches Prinzip • Shareholder vs. Stakeholder Ansatz • Unternehmensführung • Unternehmenskultur • Corporate behavior • Unternehmensstrategie • Rechtsformwahl, Unternehmensverfassung • Standortwahl • Organisation als betriebliche Aufgabe <p>Die Studierenden erhalten die Kompetenz, Unternehmen im wirtschaftlichen Kontext besser zu verstehen und Sachverhalte zu beurteilen.</p>				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Seminar				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Klausur				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				

9	Verwendung des Moduls (in anderen Studiengängen) im B.Sc.-Studiengang Informatik
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende N.N.
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Jung, H.: <i>Allgemeine Betriebswirtschaftslehre</i> , München/Wien Schierenbeck, H.: <i>Grundzüge der Betriebswirtschaftslehre</i> , München/Wien Thommen, J.-P./Achleitner, A.-K.: <i>Allgemeine Betriebswirtschaftslehre. Umfassende Einführung aus managementorientierter Sicht</i> , Wiesbaden Wöhe, G.: <i>Einführung in die Allgemeine Betriebswirtschaftslehre</i> , München

Betriebswirtschaftslehre 2					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden vertiefen die Grundlagen der Betriebswirtschaftslehre in den Bereichen Personalwirtschaft, Produktion, Absatz (Marketing) und Finanzierung, entwickeln/wählen eine Geschäftsidee und machen erste Schritte bei der Konzeption des eigenen Business.				
3	Inhalte <ul style="list-style-type: none"> • Personalwirtschaft • Betriebswirtschaftlicher Blick auf das Thema Produktion • Absatz/Marketing Entwicklung einer Geschäftsidee • Investitionsplanung • Finanzierung • Erste Schritte im Rechnungswesen • Die Studierenden erhalten die Kompetenz, weitergehend beurteilen zu können „wie hängt alles zusammen“ 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Seminar				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Klausur				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				

9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende N.N.
12	Sonstige Informationen Literaturlauswahl (jeweils in der aktuellen Auflage): Haberstock, L.: <i>Kostenrechnung I</i> , Berlin Kloock, J./Sieben, G./Schildbach, Th.: <i>Kosten- und Leistungsrechnung</i> , Düsseldorf Schierenbeck, H.: <i>Grundzüge der Betriebswirtschaftslehre</i> , München/Wien Schweitzer, M./Hettich, O./Küpper, H.-U.: <i>Systeme der Kosten- und Leistungsrechnung</i> , München Thommen, J.-P./Achleitner, A.-K.: <i>Allgemeine Betriebswirtschaftslehre. Umfassende Einführung aus managementorientierter Sicht</i> , Wiesbaden Wöhe, G.: <i>Einführung in die Allgemeine Betriebswirtschaftslehre</i> , München

Betriebswirtschaftslehre 3					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden lernen die Grundlagen des externen Rechnungswesens insbesondere in den Themenbereichen Bilanz und Gewinn und Verlustrechnung.</p> <ul style="list-style-type: none"> • Vom Inventar zur Bilanz • Ertrag/Aufwand; Einzahlungen/Auszahlungen; Kosten/Erlöse • Wie hängt GuV und Bilanz zusammen? • Gestalterische Maßnahmen beim Jahresabschluß • Bewertungsfragen • Abschreibung • Bilanzanalyse <p>Die Studierenden erhalten die Kompetenz, eine Planbilanz und Plan GuV für das erste Geschäftsjahr für ihr Start-up zu entwickeln sowie für das Folgejahr</p> <p>Die Studierenden erlernen darüber hinaus Grundlagen des internen Rechnungswesens, insbesondere in den Bereichen Kalkulation, Rentabilitätsrechnung und Liquiditätsrechnung.</p> <ul style="list-style-type: none"> • Kostenfaktoren/ Kostenarten • Skaleneffekt • Mindestgewinnberechnung <p>Die Studierenden erhalten die Kompetenz nunmehr einen kompletten Businessplan zu erstellen und dazu die Beurteilungsfähigkeit, ob das Gründungsvorhaben tragfähig ist, um den eigenen Lebensunterhalt zu decken. Sie erhalten Hilfestellung und Korrekturen, um zu einem ansprechenden Ergebnis zu kommen.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Vertiefung der Istkostenrechnung auf Vollkostenbasis <ul style="list-style-type: none"> • weitere Kostenarten • weitere Verfahren der innerbetrieblichen Leistungsverrechnung • Deckungsbeitragsrechnung <ul style="list-style-type: none"> • Programmplanung ohne und mit Engpässen • Eigenfertigung und Fremdbezug • Wahl des optimalen Produktionsverfahrens • Plankostenrechnung <ul style="list-style-type: none"> • starre Plankostenrechnung • flexible Plankostenrechnung auf Voll- und Teilkostenbasis <p>...</p>				

	<p>...</p> <ul style="list-style-type: none"> • Neue Instrumente des Kostenmanagements <ul style="list-style-type: none"> • Prozesskostenmanagement • Target Costing
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als Seminar</p>
5	<p>Teilnahmevoraussetzungen</p> <p>Formal: –</p> <p>Inhaltlich: –</p>
6	<p>Prüfungsformen</p> <p>Hausarbeit mit Fachvortrag</p>
7	<p>Prüfungsvorleistung</p> <p>-</p>
8	<p>Voraussetzungen für die Vergabe von Kreditpunkten</p> <p>bestandene Modulprüfung</p>
9	<p>Verwendung des Moduls (in anderen Studiengängen)</p> <p>–</p>
10	<p>Stellenwert der Note für die Endnote</p> <p>5/180</p>
11	<p>Modulbeauftragte*r und hauptamtlich Lehrende</p> <p>N.N.</p>
12	<p>Sonstige Informationen</p> <p>Literaturauswahl (jeweils in der aktuellen Auflage):</p> <p>Haberstock, L.: <i>Kostenrechnung I</i>, Berlin</p> <p>Haberstock, L.: <i>Kostenrechnung II</i>, Berlin</p> <p>Kilger, W.: <i>Flexible Plankostenrechnung und Deckungsbeitragsrechnung</i>, Wiesbaden</p> <p>Kloock, J./Sieben, G./Schildbach, Th.: <i>Kosten- und Leistungsrechnung</i>, Düsseldorf</p> <p>Schierenbeck, H.: <i>Grundzüge der Betriebswirtschaftslehre</i>, München/Wien</p> <p>Schweitzer, M./Hettich, O./Küpper, H.-U.: <i>Systeme der Kostenrechnung- und Leistungsrechnung</i>, München</p> <p>Thommen, J.-P./Achleitner, A.-K.: <i>Allgemeine Betriebswirtschaftslehre. Umfassende Einführung aus managementorientierter Sicht</i></p> <p>Wöhe, G.: <i>Einführung in die Allgemeine Betriebswirtschaftslehre</i>, München</p>

Wahlpflichtblock Mediendesign

Grundlagen des Grafikdesigns					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	3. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar und Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar und Praktika: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden erlernen die Grundlagen der visuellen Wahrnehmung und des Grafikdesigns. Sie verstehen die wesentlichen Kriterien der Gestaltung visueller medialer Produkte. Sie kennen das Vorgehen zur Entwicklung professionell gestalteter visueller medialer Produkte und haben erste Erfahrung in der Umsetzung gewonnen.				
3	Inhalte <ul style="list-style-type: none"> • Grundlagen Wahrnehmung • Grundlagen Grafik-Design • Grundlagen Typografie, Icons, Logoentwicklung • Grundlagen Layout und Print • Grundlagen Plakatgestaltung • Grundlagen Webdesign • Designprozess • Branding und Zielgruppendefinition • Design Thinking 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Seminar und Praktikum				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Kombinationsprüfungen				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen)
10	Stellenwert der Note für die Endnote 6/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): M. Wäger: Grafik und Gestaltung, Galileo Press 2010 B. Reiter, E. Ruhland: Gute Gestaltung, Addison Wesley Verlag 2011 F. Koschembar: Grafik für Nicht-Grafiker, Westend Verlag 2008 M. Diefenbach: Workbook visuelles Denken: Ideen, Generieren, Kundenskizzen Anfertigen, Scribbles schnell gestalten, Books on Demand, 2013 P. Renner: Die Kunst der Typographie, Maro 2003 G. Schweiger: Praxishandbuch Werbung, UVK Verlagsgesellschaft, 2013 M. Hahn: Webdesign, Galileo Press, 2015

Social Media (Digital and Social Media Marketing)					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	4. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar und Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar und Praktika: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen <p>Die Studierenden erwerben Grundlagenwissen und praxisorientiertes Wissen zum Digital Marketing.</p> <p>Sie eignen sich die Grundlagen der Strategieentwicklung in Online Marketing, Mobile Marketing und Social Media an und verstehen diese.</p> <p>Sie verstehen die wichtigsten Tools im Digital Marketing und deren Aufbau und Funktionsweise.</p> <p>Sie kennen die Formen, Anwendungsmöglichkeiten und Analyse-Methoden von Social Media-Plattformen.</p> <p>Sie verstehen die Herausforderungen und Chancen im Mobile Marketing.</p> <p>Sie können Best von Worst Practice im Digital Marketing und Omni-Channel Marketing unterscheiden.</p> <p>Sie kennen die wichtigsten Do's and Don'ts im digitalen Marketing.</p>				
3	Inhalte <p>Digital Marketing Basics</p> <ul style="list-style-type: none"> • Grundlagen der Marketinginstrumente • Grundlagen und Formen des Digital Marketing • Quantifizierung und KPIs im Digital Marketing • Kundengewinnung, Kundenbewertung und Kundenbindung <p>Online Marketing</p> <ul style="list-style-type: none"> • Search EngineAdvertising (SEA) • Search Engine Optimiziation (SEO) • Affiliate Marketing <p>Mobile Marketing</p> <ul style="list-style-type: none"> • Formen des Mobile Marketings • Social-Local-Mobile • Technologien und Anwendungen <p>Social Media Marketing</p> <ul style="list-style-type: none"> • Nutzungsszenarien von Social Media • Einbettung in die Marketing-Strategie • Operativer Marketing-Prozess • Influencer <p>Omni-Channel Marketing</p> <ul style="list-style-type: none"> • Cross-Channel-und Omni-Channel-Marketing • Marketing-Kooperationen • Best-Practices im Omni Channel-Marketing 				

4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Seminar und Praktikum
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine
6	Prüfungsformen Mündliche Prüfung
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen)
10	Stellenwert der Note für die Endnote 6/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): CHAFFEY, D., 2015. Digital Marketing –Strategy, Implementation and Practice. ISBN 978-1292077611 KOLLMANN, T., 2016. E-Business –Grundlagen elektronischer Geschäftsprozesse in der Net Economy. 6. Auflage. ISBN 978-3-658-07669-6 KREUTZER, R., 2014. Praxisorientiertes Online-Marketing: Konzepte -Instrumente -Checklisten. 2. Auflage. ISBN 978-3-658-02389-8 KREUTZER, R. und K.-H. LAND, 2016. Digitaler Darwinismus: Der stille Angriff auf Ihr Geschäftsmodell und Ihre Marke. 2. Auflage. ISBN 978-3-658-11305-6 LAMMENETT, E., 2017. Praxiswissen Online-Marketing. 6. Auflage. ISBN 978-3-658-15493-6

Mediendesign (Grundlagen Audiovisuelles Mediendesign)					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar und Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar und Praktika: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden erlernen die Grundlagen audiovisuellen Wahrnehmung. Die Studierenden erwerben grundlegende Fähigkeiten, Kenntnisse und Fertigkeiten zu fotografischen und audiovisuellen Gestaltung digitaler multimedialer Medien.				
3	Inhalte <ul style="list-style-type: none"> • Grundlagen Wahrnehmung • Grundlagen Bildaufbau / Komposition • Grundlagen Fotografie Einführung in die Kamera- und Objektivtechnik, Labor- und Bildbearbeitungstechnik Einführung in lichtbildnerisches Gestalten Einführung in die Bildbearbeitungssoftware Photoshop • Die Grundprinzipien der Animation • Grundlagen Motion Design • Grundlagen Filmgestaltung / Filmkonzeption • Projekt zur Umsetzung von kreativen Ideen in audiovisuelle Konzepte am Beispiel eines Social Clip: Ideen, Pitching, Treatment, Expose, Licht, Ton, Videodreh, Schnitt + Vertonung in Adobe Premiere oder Final CutPro / DVD Finishing Sound-Design: Tonkonzept zum Videoclip/ Layout des Tonkonzepts in Audacity oder Reaper/ Steromix und Finishing eines vertonten QT-Videos 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Seminar und Praktikum				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: keine				
6	Prüfungsformen Hausarbeiten				

7	Prüfungsvorleistung -
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) Im Verbund B.Sc.-Studiengang Angewandte Informatik
10	Stellenwert der Note für die Endnote 6/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Michael Rübsam
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): P. Kandorfer: Lehrbuch der Filmgestaltung, Schiele und Schön, 2010 W. Faulstich: Grundkurs Filmanalyse, UTB Verlag 2008 J. Monaco: Film verstehen, Rowohlt-Taschenbuch-Verl., 2001 S. D. Katz -Die richtige Einstellung. Shot by shot, Zweitausendeins, 1999 W. Kamp: AV-Mediengestaltung Grundwissen, Europa-Lehrmittel Verlag 2010 S. Drate, D. Robbins, J. Salavetz: Motion by Design, Laurence King Publishing, 2006 M. Betancourt: The History of Motion Graphics : from Avant-Garde to Industry in The United States, Wildside Press, 2013. J. J. Marchesi, Handbuch der Fotografie, Band 1-3, Verlag Photographie, 2006 R. E. Williams, The Animator's Survival Kit, Faber and Faber, 2009 M. Wäger: Grafik und Gestaltung, Galileo Press 2010 M. Diefenbach: Workbook visuelles Denken: Ideen, Generieren, Kundenskizzen Anfertigen, Scribbles schnell gestalten, Books on Demand, 2013

Wahlpflichtblock Recht

Datenschutz					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	3./7. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Der Kurs vermittelt den Studierenden rechtliche Kenntnisse in wesentlichen Bereichen des privaten Datenschutzes. Die Studierenden lernen, in der gesetzlich vorgeschriebenen Weise mit personenbezogenen Daten umzugehen. Gleichzeitig stellt das Modul für sie eine Einführung in die IT-Sicherheit dar. Im Bereich des zivilgesellschaftlichen Engagements (Bedeutung der Informatik für die Gesellschaft) sind die Studierenden für Fragen des Datenschutzes sensibilisiert.				
3	Inhalte <ul style="list-style-type: none"> • einschlägige Gesetze DSGVO, Bundesdatenschutzgesetz, aber auch Sondergesetze wie zum Beispiel das Telemediengesetz, TTDSG und verwandte Gesetze • Grundzüge des Datenschutzes Definitionen, Datensparsamkeit, Datenvermeidung • Übermittlung von Daten zwischen verschiedenen Stellen, Übermittlung von Daten ins Ausland, Umgang mit Daten im Internet • Erlaubnistatbestände der Datenerhebung, -verarbeitung und -nutzung (Art. 5 u.6 DSGVO); auch bei besonderen Kategorien (sensiblen Daten) • Datenschutz im Arbeitsverhältnis • Betriebliche*r Datenschutzbeauftragte*r Bestellung, Aufgaben, Abberufung und Kündigung 				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als seminaristischer Unterricht				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Klausur				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				

8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Andreas Göbel
12	Sonstige Informationen –

IT-Recht					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	4./8. Semester	Sommersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Veranstaltung gibt den Studierenden einen Überblick über die Rechte und Pflichten bei den vielfältigen Formen von Verträgen in der IT: Software- Überlassung, Lizenzverträge, EULA, Monitoring, Managed Service, Cloud Computing, usw. sowie bei der Betätigung im Internet, zum Beispiel in Bezug auf Urheberrechte oder Strafrecht.</p> <p>Die Studierenden kennen das in den Rechtswissenschaften übliche Vokabular und haben das Lesen, Verstehen und Formulieren von Vertragstexten geübt.</p> <p>Sie erfahren so, wo die besonderen Gefahren liegen und wie man sie vermeidet. Gleichzeitig erwerben sie dadurch die Kompetenz, die Gefahren von IT-Verträgen zu beurteilen und zu erkennen, wo weitere Beratung notwendig ist.</p>				
3	<p>Inhalte</p> <ul style="list-style-type: none"> • Einführung in das Recht von EDV und Internet • typische Probleme beim Kauf von Hardware und Software • typische Probleme bei der Erstellung von SW und der Durchführung von Dienstleistungen • Allgemeine Geschäftsbedingungen: Vereinbarung und zulässige Inhalte • Das EDV-Projekt: typische Probleme und Fallen • Grundzüge des Urheberrechts: Lizenzen • EDV-Recht im Arbeitsverhältnis: Abmahnung, Kündigung, Beweislast • Vertragsschluss im Internet • Typische Verträge im Internet: Versteigerung, Power-shopping u.a. • e-commerce: online-Handel und Verbraucherschutz • Haftung für Inhalte und Links im Internet: Access- und Contentprovider • Internet und Email am Arbeitsplatz • Grundzüge des Rechts der Domains • Datenschutz I: Grundzüge • Datenschutz II: Online-Dienste, Übermittlung ins Ausland • Strafrecht und Compliance: Computerstraftaten und Terrorbekämpfung 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als seminaristischer Unterricht</p>				

5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Andreas Göbel
12	Sonstige Informationen -

Big-Data- Cloud und KI-Recht					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	5. Semester	Wintersemester	1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Seminar	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Seminar: 30 Studierende	
2	<p>Lernergebnisse (learning outcomes) / Kompetenzen</p> <p>Die Studierenden entwickeln einen Überblick über die rechtlichen Regelungen für die Bereiche Cloud, Big Data und Künstliche Intelligenz des Rechts, u. a. bei Datenschutz, IT-Sicherheit, Vertragsrecht und Arbeitsrecht.</p> <p>Sie können mit den Begrifflichkeiten umgehen und Probleme identifizieren; sie erkennen, welche Fragen sie selbst eigenverantwortlich lösen können und ab wann sie rechtliche Beratung benötigen; in größeren Unternehmen sind sie kompetente Gesprächspartner für die firmeneigene Rechtsabteilung.</p> <p>Darüber hinaus wissen die Studierenden, welche Punkte bei der Abfassung von Verträgen zu berücksichtigen sind.</p>				
3	<p>Inhalte</p> <p>Cloud/Big-Data:</p> <ul style="list-style-type: none"> • Grundlagen von Cloud-Diensten; Telekommunikationsrecht (TKG, TMG) • Datenschutz im Internet: e-PrivacyVO 2022 (Cookies, Online Behavioural Targeting, Facebook Custom Audience, Internet der Dinge) <p>Data-Mining:</p> <ul style="list-style-type: none"> • Rechtmäßigkeit durch Anonymisierung, Pseudonymisierung, Einwilligung, Interessenabwägung. • Data-Warehousing <p>IT-Sicherheit:</p> <ul style="list-style-type: none"> • Anforderungen: Gesetze zur Cybersicherheit: BSI-G, IT-Sicherheitsgesetz 2.0 • Haftung für fehlende IT-Sicherheit: Unternehmen, Geschäftsführung, Arbeitnehmer <p>KI:</p> <ul style="list-style-type: none"> • Grundfragen: Schöpfung und Schöpfer des Algorithmus und seiner Weiterentwicklungen; Recht und Ethik: Das Trolley- Problem • Rechtliche Grundlagen: Vertragsschluss – von automatisiert bis autonom; Vertretungsrecht/Vollmachten; Haftung: Kausalität und Verantwortung bei Schäden durch KI • Datenschutz: Rechtfertigungsgrundlage; rechtmäßiger Umgang mit personenbezogenen Daten; automatisierte Einzelentscheidung/Profiling; Verarbeitung durch Auftragsverarbeiter; Übermittlung ins Ausland • Datenschutz im Arbeitsrecht: Überwachungsdruck, Transparenz; Gesundheitsschutz, Arbeitsschutz; Mitbestimmungsrechte des Betriebsrats; Betriebsvereinbarung KI • Diskriminierungsrecht;; Urheberrecht 				
4	<p>Lehr- und Lernformen</p> <p>Selbststudium auf Grundlage von Lehrbriefen</p> <p>Präsenzlehre als seminaristischer Unterricht</p>				

5	Teilnahmevoraussetzungen Formal: – Inhaltlich: – Datenschutz, IT-Recht
6	Prüfungsformen Klausur
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung
9	Verwendung des Moduls (in anderen Studiengängen) –
10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Andreas Göbel
12	Sonstige Informationen -

Containermodule

Container Algorithmen

Praktische Anwendungen von Algorithmen (Container Algorithmen)					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125h	5 ECTS	7.	Wintersemester	1 Semester
1	Lehrveranstaltungen Seminar / Blockveranstaltung: 4 SWS / 45 h		Kontaktzeit 4 SWS / 45 h	Selbststudium 80 h	geplante Gruppengröße
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden kennen Strategien zur Lösung komplexer algorithmischer Probleme wie dynamische Programmierung oder die Greedy-Heuristik und können sie in der Praxis anwenden. Basis sind dabei konkrete Problemstellungen aus Wettbewerben wie etwa Google Hashcode.				
3	Inhalte Im Seminar erarbeiten die Studierenden in Dreier- oder Vierergruppen Lösungen zu Problemen aus vergangenen Wettbewerben zum Lösen algorithmischer Probleme, etwa Google Hashcode, und stellen ihre Lösungen und die verwendeten Strategien den anderen Studierenden vor. In ihren Gruppen nehmen die Studierenden an einem Wettbewerb wie Google Hashcode teil. Der konkrete Wettbewerb und der Termin wird den Studierenden zu Beginn des Moduls mitgeteilt.				
4	Lehrformen Seminar				
5	Teilnahmevoraussetzungen Formal: keine Inhaltlich: Grundlagen der Infor				
6	Prüfungsformen Portfolio				
7	Prüfungsvorleistungen keine				
8	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) in den Studiengängen Angewandte Informatik (M.Sc.) und Informatik (B.Sc.)				

10	Stellenwert der Note für die Endnote 5/180
11	Modulbeauftragte*r und hauptamtlich Lehrende Prof. Dr. Christian Gawron
12	Sonstige Informationen Literaturauswahl (jeweils in der aktuellen Auflage): Baka, Benjamin <i>Python Data Structures and Algorithms</i> David Esparza Alba: <i>Algorithms: For Competitive Programming</i>

Container Anwendungsgebiete der Informatik

Grundlagen des Grafikdesigns (Container Anwendungsgebiete der Informatik)

[siehe Modulbeschreibung Grundlagen des Grafikdesigns](#)

Mediendesign

[siehe Modulbeschreibung Mediendesign](#)

Social Media (Container Anwendungsgebiete der Informatik)

[siehe Modulbeschreibung Social Media](#)

Container Betriebswirtschaft

Betriebswirtschaftslehre 1 (Container Betriebswirtschaft)

[siehe Modulbeschreibung Betriebswirtschaftslehre 1](#)

Betriebswirtschaftslehre 2 (Container Betriebswirtschaft)

[siehe Modulbeschreibung Betriebswirtschaftslehre 2](#)

Betriebswirtschaftslehre 3 (Container Betriebswirtschaft)

[siehe Modulbeschreibung Betriebswirtschaftslehre 3](#)

Container Datenanalyse					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Datenbanksysteme

Datenbanken 2 (Container Datenbanksysteme)

[siehe Modulbeschreibung Datenbanken 2](#)

Container Gesellschaftliche Themen					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Grafische Datenverarbeitung					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Internet der Dinge					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Künstliche Intelligenz

Deep Learning (Container Künstliche Intelligenz)

[siehe Modulbeschreibung Deep Learning](#)

Einführung Machine Learning (Container Künstliche Intelligenz)

[siehe Modulbeschreibung Einführung Machine Learning](#)

Natural Language Processing (Container Künstliche Intelligenz)

[siehe Modulbeschreibung Natural Language Processing](#)

Container Programmiersysteme

3D-Grafikprogrammierung (Container Programmiersysteme)

[siehe Modulbeschreibung 3D-Grafikprogrammierung](#)

Design Pattern in der OOP (Container Programmiersysteme)

[siehe Modulbeschreibung Design Pattern in der OOP](#)

Java Enterprise Edition (Container Programmiersysteme)

[siehe Modulbeschreibung Java Enterprise Edition](#)

Mobile Applikationen (Container Programmiersysteme)

[siehe Modulbeschreibung Mobile Applikationen](#)

Skriptsprachen (Container Programmiersysteme)

[siehe Modulbeschreibung Skriptsprachen](#)

Container Quantencomputing					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Recht und Datenschutz

Datenschutz (Container Recht und Datenschutz)

[siehe Modulbeschreibung Datenschutz](#)

IT-Recht (Container Recht und Datenschutz)

[siehe Modulbeschreibung IT-Recht](#)

Big-Data-Cloud und KI-Recht (Container Recht und Datenschutz)

[siehe Modulbeschreibung Big-Data-Cloud und KI-Recht](#)

Container Spezielle Gebiete Softwareengineering					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Theoretische Informatik					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen 3 SWS Vorlesung 1 SWS Übung	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Übung: 30 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Übung				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				

Container Verteilte Systeme und Betriebssysteme

Microsoft Betriebssysteme (Container Verteilte Systeme und Betriebssysteme)

[siehe Modulbeschreibung Microsoft Betriebssysteme](#)

Rechnernetze 2 (Container Verteilte Systeme und Betriebssysteme)

[siehe Modulbeschreibung Rechnernetze 2](#)

Unix-artige Betriebssysteme (Container Verteilte Systeme und Betriebssysteme)

[siehe Modulbeschreibung Unix-artige Betriebssysteme](#)

Virtualisierung (Container Verteilte Systeme und Betriebssysteme)

[siehe Modulbeschreibung Virtualisierung](#)

Container Vorgehensmodelle / Projektmanagement					
Kennnummer	Workload	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
	125 h	5 ECTS	7./8. Semester		1 Semester
1	Lehrveranstaltungen SWS Vorlesung 1 SWS Praktikum	Kontaktzeit 18 h	Selbststudium 107 h	geplante Gruppengröße Praktikum: 15 Studierende	
2	Lernergebnisse (learning outcomes) / Kompetenzen Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
3	Inhalte Konkretisierung und Dokumentation bei der Ausgestaltung konkreter Module				
4	Lehr- und Lernformen Selbststudium auf Grundlage von Lehrbriefen Präsenzlehre als Praktikum				
5	Teilnahmevoraussetzungen Formal: – Inhaltlich: –				
6	Prüfungsformen Alle Prüfungsformen der RPO und FPO				
7	Prüfungsvorleistung Studienleistung – die genauen Modalitäten werden zu Semesterbeginn kommuniziert				
8	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung				
9	Verwendung des Moduls (in anderen Studiengängen) –				
10	Stellenwert der Note für die Endnote 5/180				
11	Modulbeauftragte*r und hauptamtlich Lehrende				
12	Sonstige Informationen				