



OSTBAYERISCHE  
TECHNISCHE HOCHSCHULE  
REGENSBURG

# Modulhandbuch

für den  
berufsbegleitenden  
Masterstudiengang

Informationstechnologie  
(M. Eng.)

SPO-Version: 08. März 2018

Wintersemester 2019/2020

erstellt am 10.12.2019

Fakultät Informatik und Mathematik

# Modulliste

Big Data.....	3
Big Data.....	4
Cloud Computing.....	8
Cloud Computing.....	9
Datensicherheit.....	12
Datensicherheit.....	13
IT Projektmanagement.....	17
IT Projektmanagement.....	18
Management in der Informationstechnologie.....	21
Management in der Informationstechnologie.....	22
Masterarbeit (Thesis).....	24
Masterarbeit (Thesis).....	25
Moderne Informatik.....	26
Moderne Informatik.....	27
Wissenschaftliches Seminar.....	31
Wissenschaftliches Seminar.....	32

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Big Data		BD
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Wolfgang Mauerer	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	10

Empfohlene Vorkenntnisse
<ul style="list-style-type: none"> <li>• Grundlegende Kenntnisse der Linearen Algebra, Analysis und Wahrscheinlichkeitstheorie</li> <li>• Fortgeschrittene Kenntnisse in OO-Programmierung (Java, Python)</li> <li>• Grundkenntnisse in theoretischer Informatik</li> <li>• Algorithmen und Programmierung (im Modul Moderne Informatik)</li> <li>• Grundkenntnisse in der Programmierung</li> <li>• Grundkenntnisse in mindestens zwei der folgenden oder angrenzenden Gebiete: Maschinelles Lernen, Data Mining, Informationsvisualisierung, Relationale Datenbanksysteme, Software Engineering</li> <li>• Datenaufbereitung (vorher im gleichen Modul), endliche Automaten</li> </ul>

Lernziele/Lernergebnisse/Kompetenzen
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

Zugeordnete Lehrveranstaltungen:

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Big Data	70 UE	10

<b>Teilmodul</b>		<b>TM-Kurzbezeichnung</b>	
Big Data		BD	
<b>Verantwortliche/r</b>		<b>Fakultät</b>	
Prof. Dr. Wolfgang Mauerer		Informatik und Mathematik	
<b>Lehrende/r / Dozierende/r</b>		<b>Angebotsfrequenz</b>	
Prof. Dr. Jan Dünneweber Prof. Dr. Jürgen Frikel Prof. Dr. Carsten Kern Prof. Dr. Wolfgang Mauerer			
<b>Lehrform</b>			
Seminaristischer Unterricht mit Übungen Problemorientiertes Lernen (Case Studies), Forschendes Lernen (Gruppenprojekt)			

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	70 UE	deutsch	10

**Zeitaufwand:**

Präsenzstudium	Eigenstudium
52,5	247,5

Studien- und Prüfungsleistung
schrP 150 Min

## Inhalte

### 1. Machine Learning and Tools

- Lineare Regression
- Neuronale Netze
- Support Vector Machines
- Hidden Markov Model
- Kernbasierte Methoden
- Diskriminanzanalyse
- Bayes Netze

### 2. Informationsvisualisierung

- Wahrnehmung und Mapping
- Interaktionstechniken
- Visualisierungstechniken
- Visualisierungstechnologien

### 3. Practical Data Analysis

- Datenanalysewerkzeuge (Rapidminer, R, R Studio, KNIME) und Anwendung typischer Operatoren
- Umsetzung ausgewählter Fallbeispiele und eines selbstgewählten Datenanalyseprojekts mit Hilfe der Werkzeuge
- Problemadäquate Auswahl und zielgerichteter Einsatz verschiedener Algorithmen zur praktischen Datenanalyse

### 4. Natural Language Processing / Text analytics

- Tokenisierung
- Stemming
- Part-of-speech tagging
- lexikalische, syntaktische Analyse
- Parsing-Verfahren
- Semantik
- maschinelle Übersetzung

### 5. High-Performance Computing

- Grundlegende Konzepte der parallelen Datenverarbeitung
- Programmierung von Shared-Memory und Distributed-Memory
- Aktuelle Methoden zur Lösung Performance-kritischer Probleme (Simulation etc.)
- Computergestützte Verifikation und Profilierung des Laufzeitverhaltens paralleler Software

## Lernziele: Fachkompetenz

- Kennenlernen grundlegender Fragestellungen der Datenanalyse (großer Datensätze) und ihrer praktischen Relevanz
- Fähigkeit, praktische Probleme der Datenanalyse (großer Datensätze) mathematisch zu formulieren

- Kennenlernen und Verstehen weiterführender mathematischer Konzepte und Methoden des maschinellen Lernens
- Kennenlernen und Verstehen wichtiger Algorithmen des maschinellen Lernens, samt ihrer Stärken und Schwächen
- Fähigkeit, zu entscheiden welche Algorithmen für welche Problemklasse angewendet werden können
- Fähigkeit, die Algorithmen auf Probleme anzuwenden
- Fähigkeit, selbständig weiterführende Literatur und Fachartikel zu lesen, zu verstehen und ihre Methoden anzuwenden
- Studierende können die Theorie unterschiedlicher Methoden des maschinellen Lernens definieren und verstehen diese.
- Studierende sind in der Lage, mit Hilfe von Methoden des maschinellen Lernens vorgegebene Probleme zu erklären und selbständig zu lösen.
- Studierende sind in der Lage einem gegebenen Problem eine geeignete Methode zuzuordnen und diese anzuwenden.
- Studierende wenden bestehende Tools an, können diese bewerten und bei Bedarf zur Lösung von Problemstellungen im Bereich des maschinellen Lernens modifizieren.
- Parallelrechner-Architektur (Flynn's Kategorisierung, (N)UMA, Multiprozessor und Multicore Systeme, GPUs, Caches und Cache-Kohärenz), Benchmarking (LINPACK und TOP500) )
- Konsistenzbegriffe (Sequentielle Konsistenz und Linearisierbarkeit nach Lamport), Sperrenfreie Synchronisation und verklemmungsfreies Locking
- GPU Programmierung: CUDA, Thrust, OpenCL und OpenACC
- Multithreading in C++17 und Java 9 (Functors & Lambdas, Future & Promise, Mutex, Guards & Streams, Parallel STL)
- Verfahren zur Lösung linearer Gleichungssysteme (Jacobi, Gauss-Seidel, SOR), Kriterien für die Konsistenz und Konvergenz approximativer Algorithmen
- Automatische Schleifenparallelisierung (Polytope Model, Schedule & Placement), PRAM Model, Typen von Datenabhängigkeiten
- OpenMP und Message-Passing Interface (MPI)
- Der Kurs ist eine Einführung in Methoden zur Informationsvisualisierung sowie deren praktische Anwendung.
- Der Kurs illustriert an Fallbeispielen grundlegende Methoden und Eigenschaften von Informationsvisualisierungsverfahren und -systemen.
- Es soll die Fähigkeit erworben werden, abstrakte Daten mithilfe von Layout-Algorithmen zu visualisieren.
- Studierende sind in der Lage, abgegrenzte Problemstellungen der Datenanalyse mit Hilfe geeigneter IT-Werkzeuge selbständig zu lösen (methodische Kompetenz)
- Studierende beherrschen eines der vorgestellten DA-Werkzeuge und können es in den verschiedenen Phasen eines DA-Projekts zielgerichtet zum Einsatz bringen (fachliche Kompetenz)
- Studierende wissen um die Herausforderungen kooperativer Teamarbeit bei der Durchführung von DA-Projekten und entwickeln adäquate Handlungsstrategien (persönliche Kompetenz)
- Motivation und Verständnis der Problematik der Analyse sog. unstrukturierter und semistrukturierter Daten wie Texte und Webseiten.
- Wissen über grundlegende Funktionsweise von Analyseverfahren für natürliche Sprache.
- Kenntnis der Anwendungsgebiete Text Analytics, Opinion Mining / Sentiment Analysis für CRM, textuelle Suche, Informationsextraktion, semantische Suche, Wissensmanagement, automatische Fragebeantwortung, Textklassifikation.
- Kenntnis regelbasierter, statistischer und hybrider Verfahren zur Textanalyse, N-gram-Modelle, Maschinelles Lernverfahren, Textcorpora und Evaluation.
- Kenntnis von Standard-Software / Frameworks wie UIMA, NLTK, usw.

- Kritische Auseinandersetzung mit den Möglichkeiten und Grenzen der Textanalyse.

#### Lehrmedien

Beamer, Folien, Tafel und Übungen, Code-Beispiele, Übungen am PC, Bücher, Videos, Beispiel-Code, Werkzeug-Tutorials

#### Literatur

- C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
- G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Springer, 2013
- Machine Learning: Hands-On for Developers and Technical Professionals, J. Bell, 1. Auflage, Wiley, 2014
- Data Mining: Practical Machine Learning Tools and Techniques, I. Witten et al., 3. Auflage, Morgan Kaufmann, 2011
- Thomas Rauber & Gudula Rünger: Parallele Programmierung, Springer Verlag GmbH, 2013.
- Michael J. Quinn: Parallel Programming in C with MPI and OpenMP, McGrawHill 2008
- S.K. Card; Mackinlay, J. & Shneiderman, B.: Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann Publishers, San Francisco, 1999
- North, M.: Data Mining for the Masses, Washington, 2012.
- Bali, R; Sarkar, D.: What you need to know about R - Kick-start your journey with R, Packt Publishing, Birmingham, 2016.
- M. Klose & D. Wrigley: Einführung in Apache Solr, O'Reilly Verlag GmbH, 2014.
- J. Ernesti, P. Kaiser: Python 3 # Das umfassende Handbuch. Sprachgrundlagen, Objektorientierung, Modularisierung. Galileo Computing

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Cloud Computing		CLOUD
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Daniel Jobst	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	10

<b>Verpflichtende Voraussetzungen</b>
keine
<b>Empfohlene Vorkenntnisse</b>
<ul style="list-style-type: none"> <li>• Angewandte Kenntnisse in einer aktuellen, objektorientierten Programmiersprache</li> <li>• Grundkenntnisse in Software Engineering, Datenbanken, Internetprotokollen und Betriebssystemen</li> <li>• Grundkenntnisse in Verteilten Systemen und Internetprotokollen</li> <li>• Grundkenntnisse in Analyse- und Designmethoden des Software Engineering</li> <li>• Kenntnisse in einer aktuellen, objektorientierten Programmiersprache</li> <li>• Grundkenntnisse zu Internetprotokollen</li> <li>• Grundkenntnisse in Datenbanken bzw. Datenbank-Management-Systemen</li> <li>• Grundkenntnisse in einem modernen Netzwerkbetriebssystem</li> <li>• Grundkenntnisse in Algorithmen und Datenstrukturen</li> </ul>

<b>Lernziele/Lernergebnisse/Kompetenzen</b>
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Cloud Computing	70 UE	10



<b>Teilmodul</b>		<b>TM-Kurzbezeichnung</b>
Cloud Computing		CLOUD
<b>Verantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Daniel Jobst	Informatik und Mathematik	
<b>Lehrende/r / Dozierende/r</b>	<b>Angebotsfrequenz</b>	
Lehrbeauftragte der Fakultät Informatik und Mathematik (LB) Prof. Dr. Markus Heckner Prof. Dr. Daniel Jobst Prof. Dr. Klaus Volbert		
<b>Lehrform</b>		
Seminaristischer Unterricht mit Übungen		

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	70 UE	deutsch	10

**Zeitaufwand:**

Präsenzstudium	Eigenstudium
52,5	247,5

Studien- und Prüfungsleistung
schrP 150 Min

## Inhalte

- Systematisierung des Konzepts „Cloud#Computing“
- Service-Modelle des Cloud-Computing
- Cloud#Computing und Verteilte Systeme: Ausgewählte Problemstellungen, Protokolle, Algorithmen und Lösungskonzepte
- IT-Management-Aspekte des Cloud#Computing
- Grundlagen, Problemstellungen und Lösungsstrategien verteilten Funktionsaufrufs
- Servicebegriff
- Architekturmodelle verteilter, Cloud-basierter Anwendungen
- Aktuelle Implementierungs-Standards und -Frameworks (z. B. Webservice basiert) und verwandter Technologien
- Serialisierung von Daten
- Microservices und deren Bedeutung für das Cloud Computing
- Cloud#Plattformen des Service-Computing
- Datenmodelle und Daten-Speichertechniken (relationale und nicht-relationale Modelle)
- Konsistenzmodelle
- Ausgewählte verteilte Datenhaltungssysteme
- Data-Grids und Caching
- Ausgewählte Algorithmen und Auswertung auf verteilte Daten
- Grundsätze funktionaler, objektorientierter Programmierung
- Design verteilter Anwendungen mit dem Aktorenmodell
- Asynchrone, Message-orientierte Kommunikation und Futures
- „Event Sourcing“ als Persistenzmodell
- Resilienz und Fehlertoleranz
- Elastizität und Clustering
- Deployment von Cloud-Computing-Anwendungen
- Ausgewählte Anbieter von Service-Modellen des Cloud-Computing
- Virtualisierungstechnologien (z. B. Container-Virtualisierung)
- Public, private und hybride Clouds
- Cloud-Operation-Systeme und #Frameworks
- Ausgewählte Beispiele moderner Cloud-Computing-Anwendungen
- Moderne algorithmische und programmiertechnische Lösungsansätze verteilter Anwendungen in der Cloud (z. B. Peer-to-Peer, Streaming, Cloud-Algorithmen u. ä.)

## Lernziele: Fachkompetenz

- Studierende können das Konzept „Cloud#Computing“ systematisch erklären und deren Teilaspekte in die Bereiche der IT einordnen
- Studierende verstehen die zugrunde liegenden Problemstellungen, Theorien und Konzepte verteilter und paralleler Systeme
- Studierende können diese auf beispielhafte Anwendungsfälle (ggf. aus der eigenen Berufspraxis) anwenden
- Studierende kennen die zentrale Bedeutung von Daten in verteilten Anwendungen und können diese diskutieren

- Studierende kennen relationale und nicht-relationale Datenmodelle und sind in der Lage zu entscheiden, welche Datenmodelle für ein gegebenes Problem geeigneter sind als andere
- Studierende lösen beispielhafte Probleme der Datenmodellierung, #speicherung und #auswertung selbstständig und können mithilfe von ausgewählten Datenhaltungssystemen beispielhaft implementieren
- Studierende sind in der Lage, die erlernten Konzepte mit dem Service#Computing praktisch zu verwenden
- Studierende kennen das Aktorenmodell
- Studierende können einfache Anwendungen beispielhaft mit Hilfe von Aktoren in einer reaktiven Plattform implementieren
- Studierende kennen die Konzepte zur Umsetzung von Resilienz und Elastizität und können diese beispielhaft umsetzen
- Studierende nutzen bestehende private, hybride oder öffentliche Cloud-Anbieter zum Deployment von kleineren Cloud#Anwendungen
- Studierende kennen die Möglichkeiten der Virtualisierung und können diese auf bisher erstellte Beispielprogramme anwenden
- Studierende kennen Aspekte ausgehälter, moderner Problemstellungen im Cloud-Computing und können diese erklären
- Studierende können beispielhafte Problemstellungen konkret designen und implementieren

#### Lehrmedien

Beamer-Präsentation, Einzel- und Gruppenübungen, Lösen von Fallbeispielen, Lehrvideos, Programmieraufgaben

#### Literatur

- Bengel, Günther (2014). Grundkurs Verteilte Systeme, 4. Aufl. Wiesbaden: Springer Vieweg
- Bengel, Günther (2015). Masterkurs Parallele und Verteilte Systeme. 2., erw. u. akt. Aufl. Wiesbaden: Springer
- Wagenknecht, Christian (2016). Programmierparadigmen, 2. Aufl. Wiesbaden: Springer Fachmedien
- Wolff, Eberhard (2016). Microservices. Heidelberg: dpunkt-Verlag
- Schicker, Edwin (2014). Datenbanken und SQL, 4. überarb. Aufl. Wiesbaden: Springer Fachmedien
- Edlich, Stefan (2010). NoSQL. München: Hanser
- Roestenburg, Raymond et al. (2017). Akka in Action. Shelter Island: Manning
- Vaugn Vernon (2016). Reactive Messaging Patterns with the Actor Model. New York et al.: Addison-Wesley
- Beitter, Tilmann et al. (2014). IaaS mit OpenStack. Heidelberg: dpunkt-Verlag
- Raj, Pethuru (2015). Learning Docker. Birmingham: Packt Publishing

Die jeweiligen Literaturhinweise zu den Detailthemen werden über die Online-Lehrplattform zur Verfügung gestellt

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Datensicherheit		DS
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Christoph Skornia	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	10

<b>Verpflichtende Voraussetzungen</b>
keine
<b>Empfohlene Vorkenntnisse</b>
<ul style="list-style-type: none"> <li>• Grundlagenwissen zur Informationssicherheit</li> <li>• Grundlagen der Kryptographie, Verschlüsselung</li> <li>• Programmierkenntnisse in C, C++ oder Java</li> </ul>

<b>Lernziele/Lernergebnisse/Kompetenzen</b>
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Datensicherheit	70 UE	10

<b>Teilmodul</b>		<b>TM-Kurzbezeichnung</b>	
Datensicherheit		DS	
<b>Verantwortliche/r</b>		<b>Fakultät</b>	
Prof. Dr. Christoph Skornia		Informatik und Mathematik	
<b>Lehrende/r / Dozierende/r</b>		<b>Angebotsfrequenz</b>	
Prof. Dr. Georgios Raptis Prof. Dr. Christoph Skornia			
<b>Lehrform</b>			
Seminaristischer Unterricht mit integrierten Übungen			

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	70 UE	deutsch	10

Zeitaufwand:

Präsenzstudium	Eigenstudium
52,5h	247,5h

Studien- und Prüfungsleistung
schrP 150 Min

## Inhalte

- 1-Faktor, 2-Faktor Authentifizierungsverfahren
- Biometrische Authentifizierung
- Starke Authentifizierung mit kryptographischen Verfahren, Zero-Knowledge Protokolle
- Code Signing
- Sichere E-Mail: S/MIME, DE-Mail
- Chipkarten, RFID-Systeme
- Identity Provider, ServiceProvider, Single-Sign-On, gängige Protokolle, Policies
- Machine-to-machine Kommunikation und ID-Management, Webservices, DNSSec
- Public Key Infrastrukturen, Artefakte, Protokolle und zugehörige Dienste, LDAP, OCSP
- X.509-Zertifikate, Lifecycle
- eID-Funktion des Personalausweises
- Für die Informatik relevante, rechtliche Grundlagen des Datenschutzes
- Bedrohungen des Datenschutzes
- Privacy by Design, Privacy by Default, Best Practices
- Begriffe, Rechte und Maßnahmen: Datenerhebung, Speicherung, Verarbeitung, Kontrolle, Weitergabe, Löschung, Sperrung, Zweckbindung, Datensparsamkeit
- Verfahrensverzeichnis, Vorab-Kontrolle, Datenschutzbeauftragte/r, Auftragsdatenverarbeitung
- Neue Begriffe und Regelungen der EU-Datenschutzgrundverordnung: Rechenschaftspflicht ("Accountability"), Verzeichnis von Verarbeitungstätigkeiten, Sicherheit der Verarbeitung, Meldepflichten bei Verstößen, Datenschutz-Folgenabschätzung, Auftragsverarbeiter, Bußgelder
- Datenschutzmanagement
- Anforderungen und Techniken zur sicheren Pseudonymisierung und Anonymisierung
- Anforderungen für den rechtskonformen und sicheren Einsatz von Cloud Diensten
- Kryptographische Hashfunktionen: Konstruktion, Eigenschaften, Anwendungen
- Integrität und Authentizität von Netzwerkdaten: Message Authentication Codes, HMAC
- Elektronische Signaturen: kryptographische und technische Grundlagen
- Qualifizierte elektronische Signaturen, Zeitstempel, Siegel
- Anforderungen an die Archivierung elektronisch signierter Dokumente, relevante Vertrauensdienste
- Sicherheitsbezogene Anforderungsanalyse für die Softwareentwicklung
- Implementierung ausgewählter Sicherheitsfunktionalitäten aus den Bereichen Authentifizierung und Verschlüsselung
- Analyse von Softwareschwachstellen mittels statischer und dynamischer Analyse
- Strategien zur Vermeidung häufiger Sicherheitsprobleme in Software
- Verfahrensmodelle für sicheres Softwaredesign
- Historische Verschlüsselungsverfahren
- Aktuelle symmetrische und asymmetrische Verschlüsselungsverfahren
- Block- und Stromchiffren
- Schlüsselaustauschverfahren
- Public-Key Verschlüsselung
- Public-Key Infrastruktur
- Einsatz von Penetration Testing zur Erhöhung der IT-Sicherheit
- Phasenmodell im Penetration Testing
- Kill-Chain-Methodologie
- Schwachstellen
- Einsatz von Werkzeugen im Penetration Testing am Beispiel von Metasploit und Kali Linux
- Angriffsoberfläche und Angriffsvektoren

- Sicherheitsanalyse von Informationssystemen
- Lokale Gefahrenquellen
- Gefahren aus dem Netzwerk
- Schutzziele und Sicherheitsmodelle
- Technische Maßnahmen der lokalen Systemsicherheit
- Technische Maßnahmen der Netzwerksicherheit

#### Lernziele: Fachkompetenz

- Die Studierenden beherrschen die mathematischen, organisatorischen und technischen Grundlagen der Informationssicherheit und des Datenschutzes.
- Sie haben fundierte Kenntnisse zu den Themenfeldern der Authentifizierung, des Identity Managements, der Verschlüsselungs- und Signaturverfahren und des Datenschutzes in der Informationstechnologie erworben.
- Die Studierenden haben fundierte Kenntnisse zum Thema Authentifizierung erworben und beherrschen die Methoden und Werkzeuge, um moderne Authentifizierungsverfahren in verschiedenen Sicherheitsniveaus einzubinden.
- Sie können das Potential sowie die Vor- und Nachteile verschiedener Authentifizierungsverfahren im Kontext einer Anwendung evaluieren.
- Die Studierenden haben Kenntnisse der Funktionsweise und der Sicherheitsleistung fortgeschrittener Authentifizierungswerkzeuge, wie Chipkarten und RFID-Systeme erworben.
- Die Studierenden haben fundierte Kenntnisse zum Thema Identity Management erworben.
- Sie beherrschen die Methoden und Werkzeuge, um ID-Management Technologien in Anwendungen einzubinden.
- Die Studierenden verstehen die Grundlagen, Protokolle und Dienste von Public-Key Infrastrukturen, die Funktionsweise verschiedener ID-Management-Systeme und können sie in verschiedenen Szenarien einsetzen.
- Die Studierenden beherrschen die Grundlagen des Datenschutzes in der Informationstechnologie und können datenschutzkonforme IT-Systeme entwerfen, betreiben und evaluieren.
- Die Studierenden haben Kenntnisse zum Datenschutzmanagement und zur rechtskonformen Auslagerung von IT-Diensten erworben und beherrschen die rechtlichen und technischen Anforderungen für den datenschutzkonformen Einsatz von Cloud-Diensten.
- Die Studierenden verfügen über fundierte Kenntnisse zu den mathematischen, technischen und rechtlichen Grundlagen und Methoden von Signaturverfahren und Vertrauensdiensten und können diese in der Praxis evaluieren und einsetzen.
- Die Studierenden kennen die wichtigsten Sicherheitsprobleme aktueller Software und deren Ursachen sowie Strategien zu deren Vermeidung.
- Sie können ausgewählte Sicherheitsfunktionalitäten implementieren.
- Darüber hinaus sind Sie in der Lage sicherheitsorientiertes Softwaredesign in Entwicklungsprojekten umzusetzen
- Die Studierenden beherrschen die konzeptionellen Grundlagen der Verschlüsselung.

- Sie sind in der Lage ausgewählte Algorithmen korrekt einzusetzen sowie übergeordnete Verschlüsselungsverfahren auf Ihre Eignung für ein konkretes Anwendungsszenario zu beurteilen.
- Die Studierenden verstehen den systematischen Aufbau von Penetrationstests und können diesen in der Praxis anwenden.
- Sie wissen, wie verfügbare Werkzeuge dazu zur Erhöhung der Sicherheit eingesetzt werden.
- Die Studierenden verstehen Verfahren zur Sicherheitsanalyse von Informationssystemen und können diese in der Praxis einsetzen.
- Sie sind in der Lage Schwachstellen zu identifizieren, zu bewerten und Maßnahmen zu deren Behebung vorzuschlagen.
- Sie können ihr Vorgehen dabei auf Basis gängiger Sicherheitsmodelle selbstständig planen und umsetzen.

#### Lehrmedien

Skript, Folien

#### Literatur

- C. Eckert, IT-Sicherheit: Konzepte - Verfahren # Protokolle, De Gruyter Oldenbourg, 9. Auflage, 2014
- J. Buchmann: Einführung in die Kryptographie, Springer Spektrum, 2016
- G. Borges, J. Schwenk: Daten- und Identitätsschutz in Cloud-Computing, E-Government und E-Commerce, Springer-Verlag Berlin-Heidelberg, 2012
- D. Loomans, M. Matz, M. Wiedemann: Praxisleitfaden zur Implementierung eines Datenschutzmanagementsystems. Springer-Vieweg, 2014
- Verordnung (EU) 2016/679 des Europäischen Parlaments und des Rates vom 27. April 2016 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten, zum freien Datenverkehr und zur Aufhebung der Richtlinie 95/46/EG (Datenschutz-Grundverordnung). Amtsblatt der Europäische Union, 27.04.2016
- V. Gruhn et al: Elektronische Signaturen in modernen Geschäftsprozessen, Vieweg, 2007
- Müller, Klaus-Rainer. IT-Sicherheit mit System: Sicherheitspyramide-Sicherheits-, Kontinuitäts- und Risikomanagement-Normen und Practices-SOA und Softwareentwicklung. Springer-Verlag, 2007.
- Viega, John, and Matt Messier. Secure Programming Cookbook for C and C++: Recipes for Cryptography, Authentication, Input Validation & More. " O'Reilly Media, Inc.", 2003.
- Ertel, Wolfgang. Angewandte Kryptographie. Carl Hanser Verlag GmbH Co KG, 2012.
- Eckert, Claudia. IT-Sicherheit: Konzepte-Verfahren-Protokolle. Walter de Gruyter, 2013.
- Beggs, Robert W. Mastering Kali Linux for Advanced Penetration Testing. Packt Publishing Ltd, 2014.
- Messier, Ric. "Penetration testing basics." (2016).
- Hofmann, Jürgen. "IT-Sicherheitsmanagement." Masterkurs IT-Management. Vieweg+ Teubner, 2010. 287-334.
- BSI, IT. "Sicherheitsmanagement und IT-Grundschutz-BSI-Standards zur IT-Sicherheit." (2005).

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden



Modulbezeichnung (ggf. englische Bezeichnung)		Modul-KzBez. oder Nr.
IT Projektmanagement		ITPM
Modulverantwortliche/r	Fakultät	
Prof. Dr. Markus Heckner	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	5

Verpflichtende Voraussetzungen
keine
Empfohlene Vorkenntnisse
<ul style="list-style-type: none"> <li>• Grundlagenwissen Projektmanagement</li> <li>• Grundlagen Softwareentwicklung</li> </ul>

Lernziele/Lernergebnisse/Kompetenzen
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

#### Zugeordnete Lehrveranstaltungen:

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	IT Projektmanagement	35 UE	5

<b>Teilmodul</b>		<b>TM-Kurzbezeichnung</b>	
IT Projektmanagement		ITPM	
<b>Verantwortliche/r</b>		<b>Fakultät</b>	
Prof. Dr. Markus Heckner		Informatik und Mathematik	
<b>Lehrende/r / Dozierende/r</b>		<b>Angebotsfrequenz</b>	
Prof. Dr. Markus Heckner			
<b>Lehrform</b>			
Seminaristischer Unterricht mit Übungen			

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	35 UE	deutsch	5

Zeitaufwand:

Präsenzstudium	Eigenstudium
26,25	123,75

Studien- und Prüfungsleistung
StA m.P.

## Inhalte

### Agiles Projektmanagement mit Scrum

Usability Engineering und Integration in unterschiedliche Projektmanagementmethoden mit Fokus auf Scrum

- Übergang zu agilen Projekten in Organisationen
- Individuen # Widerstände überwinden, neue Rollen in agilen Projekten
- Scrum: Sprints und Teams
- Estimation, Planung und Priorisierung
- Tools für agiles Projektmanagement
- Agiles Risikomanagement
- Agile und UX # Integration von Usability Engineering und Scrum
- Ausblick auf das Skalieren von Scrum für größere Projekte (Scrum@Scale, Scaled Agile Framework (SAFe), Large Scale Scrum LeSS)

Vermittlung der Grundprinzipien einer benutzergerechten Entwicklung von Software (User Centered Design).

Themen:

- Usability Engineering Framework
- Methoden der nutzerzentrierten Anforderungsanalyse
- Information Design und Information Architecture
- Sketching
- Paper Prototyping
- Toolbasiertes Prototyping mit Axure
- Usability Testing

Projektphase und Betreuung im Anschluss

## Lernziele: Fachkompetenz

- Die Studenten sollen ein grundlegendes Verständnis agiler Softwareentwicklung erhalten und dieses gegenüber herkömmlichen Modellen (Wasserfall, V-Modell, Rational Unified Process, ...) abgrenzen können.
- Die Studenten sollen in der Lage sein ein eigenes Projekt mit Scrum aufzusetzen, zu planen und mit einer entsprechenden Projektmanagementsoftware entsprechend zu begleiten.
- Die Studenten sollen ein grundlegendes Verständnis von Usability / User Experience und deren Auswirkungen auf Nutzer und Unternehmen erlangen.
- Die Studenten sollen den in der DIN ISO 9241-210 definierten systematischen Prozess zur Gestaltung interaktiver Systeme verstehen, der ein Prozessmodell zur Entwicklung benutzerzentrierter Software vorgibt.
- Die Studenten sollen die dazu notwendigen Methoden (z.B. Analyse des Nutzungskontexts, Prototyping, Card Sorting, Usability Testing inkl. Auswertung) selbstständig einsetzen, um das User Interface einer Anwendung so zu gestalten, dass diese effizient und effektiv benutzbar wird.

<b>Lehrmedien</b>
Beamer, Notebook
<b>Literatur</b>
<ul style="list-style-type: none"><li>• Cohn, M. (2010). Succeeding with Agile. Upper Saddle River: 2010.</li><li>• Gothelf (2016). Lean UX. Designing great products with Agile teams. O'Reilly.</li><li>• DIN EN ISO 9241-210. Human-centred design for interactive systems.</li><li>• Nodder, C. &amp; J. Nielsen (2009). Agile Usability: Best Practices for User Experience on Agile Development Projects.</li><li>• Tullis, T., &amp; Albert, B. (2008). Measuring the User Experience. Morgan Kaufmann.</li><li>• Warfel, T. Z. (2009). Prototyping: A Practitioner's Guide (1st ed.). Rosenfeld Media.</li></ul>

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Management in der Informationstechnologie		MIT
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Markus Westner	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	5

<b>Verpflichtende Voraussetzungen</b>
keine
<b>Empfohlene Vorkenntnisse</b>
keine

<b>Lernziele/Lernergebnisse/Kompetenzen</b>
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Management in der Informationstechnologie	35 UE	5

Teilmodul		TM-Kurzbezeichnung
Management in der Informationstechnologie		MIT
Verantwortliche/r	Fakultät	
Prof. Dr. Markus Westner	Informatik und Mathematik	
Lehrende/r / Dozierende/r	Angebotsfrequenz	
Prof. Dr. Markus Westner		
Lehrform		
Projektarbeit im Team		

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	35 UE	deutsch	5

Zeitaufwand:

Präsenzstudium	Eigenstudium
26,25	123,75

Studien- und Prüfungsleistung
StA m.P.

Inhalte
<p>Managementplanspiel</p> <p>Allgemein: Gruppenorganisation, Absatzplanung, Produktionsplanung, Personalplanung, Beschaffungsplanung, Finanz- und Liquiditätsplanung, Kosten- und Erfolgsplanung, Deckungsbeitragsrechnung, Jahresabschlusserstellung, Plan-Ist-Analyse, Präsentation der Abschlussergebnisse.</p> <p>Eingesetzte Planspiele:</p> <ul style="list-style-type: none"> <li>• SEED (Haptisches Planspiel)</li> <li>• TOPSIM General Management II (13.0)</li> </ul> <p>Sonderaufgaben:</p> <ul style="list-style-type: none"> <li>• Impulsreferate zu ausgewählten Managementthemen,</li> <li>• Presseartikel, Werbekonzepte, Internetauftritt, Firmenlogo-/slogan etc</li> </ul>
Lernziele: Fachkompetenz
<p>Vertiefung und Verknüpfung der Kenntnisse aus den betrieblichen Funktionen in realitätsabbildenden Unternehmensplanspielen mittels Computersimulation. Förderung der Sozialkompetenz und Teamfähigkeit. Weiterentwicklung der Präsentationsfähigkeit.</p>
Lehrmedien
Tafel, Flipchart, Notebook, Beamer, Videokamera

## Literatur

- Teilnehmerhandbuch zu den Planspielen
- Literaturhinweise zu den Referatsthemen

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Masterarbeit (Thesis)		MTH
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Vorsitzender der Prüfungskommission	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	30

<b>Verpflichtende Voraussetzungen</b>
40 Kreditpunkte aus den ersten beiden Studiensemestern
<b>Empfohlene Vorkenntnisse</b>
Alle Pflicht-Module

<b>Lernziele/Lernergebnisse/Kompetenzen</b>
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Masterarbeit (Thesis)		30



<b>Teilmodul</b>		<b>TM-Kurzbezeichnung</b>	
Masterarbeit (Thesis)		MTH	
<b>Verantwortliche/r</b>		<b>Fakultät</b>	
Vorsitzender der Prüfungskommission		Informatik und Mathematik	
<b>Lehrende/r / Dozierende/r</b>		<b>Angebotsfrequenz</b>	
alle Prüfer/innen des Masterstudiengangs Informatik			
<b>Lehrform</b>			
<ul style="list-style-type: none"> <li>• Selbständige Bearbeitung eines fachwissenschaftlichen Problems</li> <li>• Erstellen einer schriftlichen Ausarbeitung</li> <li>• Vorbereiten einer Präsentation</li> </ul>			

<b>Studiensemester gemäß Studienplan</b>	<b>Lehrumfang</b> [SWS oder UE]	<b>Lehrsprache</b>	<b>Arbeitsaufwand</b> [ECTS-Credits]
2-7		deutsch/englisch	30

Zeitaufwand:

<b>Präsenzstudium</b>	<b>Eigenstudium</b>

<b>Studien- und Prüfungsleistung</b>
Masterarbeit: Ausarbeitung

<b>Inhalte</b>
Fachwissenschaftliches Thema
<b>Lernziele: Fachkompetenz</b>
Die Studierenden können ein fachwissenschaftliches Problem selbständig bearbeiten, Lösungsansätze im Team diskutieren und die Ergebnisse in mündlicher und schriftlicher Form präsentieren.
<b>Literatur</b>
<b>Weitere Informationen zur Lehrveranstaltung</b>
Medienform: Papier, CD/DVD, PDF-Datei u.a. Zeitaufwand 900 h: 833h zur Bearbeitung und Ausarbeitung, 67h zur Vorbereitung der Präsentation

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

<b>Modulbezeichnung (ggf. englische Bezeichnung)</b>		<b>Modul-KzBez. oder Nr.</b>
Moderne Informatik		MODINF
<b>Modulverantwortliche/r</b>	<b>Fakultät</b>	
Prof. Dr. Klaus Volbert	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
1		Pflicht	10

<b>Verpflichtende Voraussetzungen</b>
keine
<b>Empfohlene Vorkenntnisse</b>
<ul style="list-style-type: none"> <li>• Grundlagen in der Algorithmik und in der objektorientierten Programmierung</li> <li>• Fortgeschrittene Kenntnisse in Software Engineering</li> <li>• Fortgeschrittene Kenntnisse in OO-Programmierung (Java)</li> <li>• Grundkenntnisse in Automatentheorie (theoretische Informatik)</li> <li>• Kenntnisse in Linearer Algebra und Analysis</li> <li>• Grundlegende Kenntnisse der linearen Algebra, Analysis und Wahrscheinlichkeitstheorie</li> </ul>

<b>Lernziele/Lernergebnisse/Kompetenzen</b>
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Moderne Informatik	70 UE	10

Teilmodul		TM-Kurzbezeichnung
Moderne Informatik		MODINF
Verantwortliche/r	Fakultät	
Prof. Dr. Klaus Volbert	Informatik und Mathematik	
Lehrende/r / Dozierende/r	Angebotsfrequenz	
Prof. Dr. Jürgen Friel Prof. Dr. Daniel Jobst Prof. Dr. Carsten Kern Prof. Dr. Stefan Körkel Prof. Dr. Klaus Volbert		
Lehrform		
Seminaristischer Unterricht und Übungen		

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
1	70 UE	deutsch	10

Zeitaufwand:

Präsenzstudium	Eigenstudium
52,5h	247,5h

Studien- und Prüfungsleistung
schrP. 150 Min

## Inhalte

- Algorithmische Grundkonzepte (Komplexität, Komplexitätsklassen, P und NP, NP-Vollständigkeit, Entwurfsmethoden)
- Basisalgorithmen mit Beispielen (Sortieren, Suchen, Hashing, Bäume, Graphen)
- Fortgeschrittene Algorithmen und Analysemethoden (Amortisierte Analyse, Approximationsalgorithmen, weitere ausgewählte Algorithmen mit Beispielen)
- Programmiersprachen-Konzepte und -Beispiele
- Konzepte der objektorientierten Programmierung
- Fortgeschrittene Programmierkonzepte

Ausgewählte Themen aus:

- Wiederholung Software Engineering 1 (Phasen im Software Engineering, agile Vorgehensmodelle)
- Modellierung mit UML (z.B. Klassendiagramme und OCL)
- Fortgeschrittene Modellierungskonzepte (z.B. EPK, BPMN, Process Engines und Webservice-Integration etc.)
- Software-Design (z.B. Analyse-, Architektur-, Design-Patterns)
- Software-Qualitätssicherung (Testtechniken und automatisiertes Testen mithilfe von Unit Tests, JUnit, TDD)
- Toolunterstützung im Software Engineering (z.B. Versionskontrolle, Buildsysteme, Trackingsysteme)

Optimierungsprobleme und Problemklassen

Modellierung von Optimierungsproblemen, Anwendungsbeispiele

Lineare Optimierung, Simplex-Algorithmus

Linear-ganzzahlige Probleme, Branch-and-Bound-Lösungsverfahren, Schnittebenenverfahren, Heuristiken

Optimalitätsbedingungen für nichtlineare Optimierungsprobleme

Verfahren von Abstiegstyp, Gradientenverfahren

Verfahren vom Newton-Typ, SQP-Verfahren

Optimierung unter Differentialgleichungsnebenbedingungen, Shooting-Verfahren

Optimale Steuerung und Parameterschätzung

Praktische Übungen mit Software und Beispielproblemen

Die Studierenden lernen grundlegende mathematische Konzepte und Algorithmen aus dem Bereich der statistischen Lerntheorie und des maschinellen Lernens. Themen sind eine Auswahl aus:

- Wiederholung Grundlagen Wahrscheinlichkeitstheorie
- Mathematische Prinzipien der (statistischen) Lerntheorie
- Überwachtes und Unüberwachtes Lernen
- K-Nearest und K-Means Klassifikation
- Bayes Klassifikation (Naive Bayes)
- Entscheidungsbäume
- Lineare Regression
- Neuronale Netze
- Support Vector Machines

### Lernziele: Fachkompetenz

- Grundlagen zur Algorithmik (Entwurfsmethoden, Komplexität, Basisalgorithmen)
- Kennenlernen der Aspekte fortgeschrittener algorithmischer Methoden
- Fähigkeit zur Charakterisierung und Analyse von Algorithmen mit mathematischen Methoden
- Die Studierenden können fortgeschrittene algorithmische Verfahren zur Lösung praktischer Probleme analysieren und implementieren.
- Die Studierenden lernen fortgeschrittene Konzepte der objektorientierten Programmierung kennen und können diese durch eigene Implementierungen anwenden.
- Die Studierenden können Ihr Wissen aus den Grundlagenveranstaltungen zu Softwareengineering benennen.
- Die Studierenden können fortgeschrittene Vorgehensmodelle, Methoden, Muster und Tools im Software Engineering erklären und diese auf konkrete Problemstellungen anwenden.
- Die Studierenden können mit fortgeschrittenen Modellierungstechniken ausführbare Modelle bewerten, optimieren und mit selbsterstellten Web Services automatisieren.
- Kennenlernen der wichtigsten Klassen von Optimierungsproblemen
- Verständnis der mathematischen Grundlagen der Lösungsalgorithmen
- Fähigkeit zur Wahl von geeigneten Lösungsverfahren
- Praktische Erfahrung im Umgang mit OptimierungssoftwareÜbung in Modellierung und Lösung von Beispielp Problemen und praktischen Anwendungsproblemen
- Kennenlernen grundlegender Fragestellungen der Datenanalyse (großer Datensätze) und ihrer praktischen Relevanz
- Fähigkeit, praktische Probleme der Datenanalyse (großer Datensätze) mathematisch zu formulieren
- Kennenlernen und Verstehen grundlegender mathematischer Konzepte und Methoden des maschinellen Lernens
- Kennenlernen und Verstehen grundlegender Algorithmen des maschinellen Lernens, samt ihrer Stärken und Schwächen
- Fähigkeit, zu entscheiden welche Algorithmen für welche Problemklasse angewendet werden können
- Fähigkeit, die Algorithmen auf Probleme (mittlerer Komplexität) anzuwenden
- Fähigkeit, selbständig weiterführende Literatur zu lesen und zu verstehen

### Lehrmedien

Beamer, Folien, Präsenzübungen und Modellierungs/Code-Beispiele  
Tafel, Übungen im Rechnerpool oder am eigenen Notebook, Skript

Literatur
<ul style="list-style-type: none"><li>• Cormen, T. H., Leisserson, C. E., Rivest, R.L., Stein, C.: Introduction to Algorithms, MIT Press, 2001</li><li>• Inden, M.: Der Weg zum Java-Profi: Konzepte und Techniken für die professionelle Java-Entwicklung, dpunkt.verlag, 3. Auflage, 2015</li><li>• UML 2 glasklar, C. Rupp, S. Queins, 4. Auflage, Carl Hanser Verlag, 2012</li><li>• OCL language specification, Object Management Group, Version 2.4, 2014</li><li>• Nocedal, J., Wright, S.: Numerical Optimization, Springer</li><li>• Jarre, F., Stoer, J.: Optimierung, Springer</li><li>• C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.</li><li>• G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Springer, 2013.</li></ul>
Weitere Informationen zur Lehrveranstaltung

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

Modulbezeichnung (ggf. englische Bezeichnung)		Modul-KzBez. oder Nr.
Wissenschaftliches Seminar		WAI
Modulverantwortliche/r	Fakultät	
Prof. Dr. Markus Heckner	Informatik und Mathematik	

Studiensemester gemäß Studienplan	Studienabschnitt	Modultyp	Arbeitsaufwand [ECTS-Credits]
2-7		Pflicht	10

Verpflichtende Voraussetzungen
keine
Empfohlene Vorkenntnisse
keine

Lernziele/Lernergebnisse/Kompetenzen
siehe Folgeseite

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden

**Zugeordnete Lehrveranstaltungen:**

Nr.	Bezeichnung der Module	Lehrumfang [SWS o. UE]	Arbeitsaufwand [ECTS-Credits]
1.	Wissenschaftliches Seminar	70 UE	10

Teilmodul		TM-Kurzbezeichnung
Wissenschaftliches Seminar		WAI
Verantwortliche/r	Fakultät	
Prof. Dr. Markus Heckner	Informatik und Mathematik	
Lehrende/r / Dozierende/r	Angebotsfrequenz	
alle Dozenten/innen der Fakultät IM		
Lehrform		
Konsultationen und Seminar		

Studiensemester gemäß Studienplan	Lehrumfang [SWS oder UE]	Lehrsprache	Arbeitsaufwand [ECTS-Credits]
2-7	70 UE	deutsch/englisch	10

Zeitaufwand:

Präsenzstudium	Eigenstudium
52,5h	247,5h

Studien- und Prüfungsleistung
StA m.P.

Inhalte
<p>Die Studierenden erarbeiten ein vorgegebenes Thema selbstständig. Grundlage bilden ein oder mehrere Fachartikel. Dabei recherchieren sie weitere relevante Fachliteratur selbstständig und ordnen das Thema und die Artikel in angrenzende Themen und verwandte Literatur ein. Mit Hilfe von Hinweisen des Dozenten erarbeiten sich die Studierenden die nötigen Spezialkenntnisse eigenständig. Fragen können in den wöchentlich angebotenen Konsultationen gestellt werden. Das Thema wird im Rahmen einer Ausarbeitung erläutert und wissenschaftlich aufbereitet. Als Blockveranstaltung werden die Themen präsentiert und sowohl Inhalt als auch Form diskutiert.</p>
Lernziele: Fachkompetenz
<ul style="list-style-type: none"> <li>• Kompetenz zur selbstständigen Einarbeitung in ein vorgegebenes eingegrenztes Spezialgebiet der Informatik mit Bezug zu den Schwerpunkten des Studiengangs</li> <li>• Kompetenz zur wissenschaftlichen Bearbeitung des Themas im Sinne des vertieften Verständnisses, der kritischen Hinterfragung und der Einordnung in die angrenzende thematischen Gebiete</li> <li>• Erfahrung mit Literaturrecherche und effizientem Umgang mit (meist englischsprachiger) Fachliteratur</li> <li>• Kompetenz zur Präsentation des Fachthemas mit geeignetem Medieneinsatz</li> <li>• Kompetenz zur fachlichen Diskussion</li> </ul>



<b>Lehrmedien</b>
Beamer, Notebook
<b>Literatur</b>
<ul style="list-style-type: none"><li>• American Psychological Association. (2010). Concise Rules of APA style. Washington, DC: American Psychological Association.</li><li>• Andermann, U., Drees, M. &amp; Grätz F. (2006). Wie verfasst man wissenschaftliche Arbeiten? Mannheim: Dudenverlag.</li><li>• Disterer, (). Studienarbeiten schreiben.</li><li>• March, S. T., &amp; Storey, V. C. (2008). Design science in the information systems discipline: An introduction to the special issue on design science research. MIS Quarterly, 32(4), 725#730.</li><li>• Reynolds, G. (2010). ZEN oder die Kunst der Präsentation. München: Addison Wesley.</li><li>• Zobel, J. (2014). Writing for Computer Science. London: Springer.</li></ul>

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden