

Modulhandbuch

M. Sc. Distributed Software Systems

Fachbereich Informatik
Technische Universität Darmstadt





TECHNISCHE
UNIVERSITÄT
DARMSTADT



Modulhandbuch

M. Sc. Distributed Software Systems

Technische Universität Darmstadt
Fachbereich Informatik
Hochschulstr. 10
64289 Darmstadt

Redaktion
Dipl.-Inform. Tim Neubacher
Jasmin Boghrat, M.A.
Stand: 07.04.2021

Inhaltsverzeichnis

Fachprüfungen

Wahlbereich Distributed Systems	4
Wahlbereich Networking and Systems Software	17
Wahlbereich Formal Methods, Programming Languages and Software Engineering	30

Studienbegleitende Leistungen

Praktika, Projektpraktika und ähnliche Veranstaltungen	59
Seminare	99
Praktikum in der Lehre	137

Masterarbeit	139
---------------------	------------

Modulhandbuch
M. Sc. Distributed Software Systems

Wahlbereich
Distributed Systems

Modulbeschreibung

Modulname Netz-, Verkehrs- und Qualitäts-Management für Internet Services					
Modul Nr. 20-00-0056	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0056-v1	Netz-, Verkehrs- und Qualitäts-Management für Internet Services	0	Vorlesung	2
2	Lerninhalt Einführung in das Management von Internet Service Provider (ISP-)Netzen zur Integration von Service Plattformen mit ihren Qualitäts- und Verkehrsprofilen				
3	Qualifikationsziele / Lernergebnisse Stoffplan: Anforderungen und Maßnahmen zur Sicherung der Quality-of-Service (QoS) - Kriterien aus Anwendungs- & Nutzer-Sicht (QoE: Quality of Experience) - QoS Architektur in IP-Netzen: Differentiated & Integrated Services - QoS Support & Auswirkung je Anwendung im IP Verkehrs-Mix (Video-Streaming, VoIP, Web Browsing, Downloads, Social Networking etc.) Qualitätssicherung für Internet Services in ISP Netzinfrastrukturen - Einfluss der Netz- und Transportebene: Routing (OSPF, BGP), Multiprotocol Label Switching (MPLS), TCP mit Absicherung gegen Fehler und Ausfälle - Messung, Monitoring, Optimierung von IP Verkehr bzgl. QoS Qualitätssicherung in Service Overlays und auf Anwendungsebene - Content Delivery Netze (CDN), Clouds und Peer-to-Peer Netze (P2P) inkl. verteilter Caches, Transportpfad-Optimierung, Skalierbarkeit - IETF Standardisierung (CDN Interconnection, ALTO: Appl. Layer Traffic Opt.)				
4	Voraussetzung für die Teilnahme Empfohlen: Vorwissen: Grundlegende Kenntnisse der Informatik und Internet-Anwendungen werden vorausgesetzt. Die Vorlesungen Kommunikationsnetze I und II sind empfohlen.				

5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0056-v1] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0056-v1] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls
9	Literatur Wird in der Vorlesung angesprochen
10	Kommentar

Modulbeschreibung

Modulname TK1: Verteilte Systeme und Algorithmen					
Modul Nr. 20-00-0065	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Wintersemester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0065-iv	TK1: Verteilte Systeme und Algorithmen	6	integrierte Lehrveranstaltung	4
2	Lerninhalt				
	<p>Lernziele:</p> <ul style="list-style-type: none"> • Umfassendes Überblickswissen über die grundlegenden Probleme und Ansätze • Tiefgehendes Methodenwissen zu klassischen verteilten Algorithmen und Programmierparadigmen • Anwendbare exemplarische Kenntnis aktueller Entwicklungen und Standards <p>Stoffplan:</p> <ul style="list-style-type: none"> • Einführung • Auffrischung und Ergänzung von Kapitel 1 der Kanonik Net-Centric Computing • Überblick über die Vorlesung • Verteilte Algorithmen <ul style="list-style-type: none"> • Elementaralgorithmen (z.B. globaler Zustand) • Basisalgorithmen (z.B. Ausschluss, Konsens, Kooperation) • Formalisierung (Eigenschaften und deren Nachweis) • Verteiltes Programmieren <ul style="list-style-type: none"> • Push-Paradigmen (z.B. IPC, RPC, DOC) • aktuelle Ansätze (z.B. Pull-Paradigmen, Objektmobilität) 				
3	Qualifikationsziele / Lernergebnisse Studierende kennen nach erfolgreichem Besuch der Veranstaltung die Grundlagen der verteilten Programmierung und verteilter Algorithmen. Sie verstehen die grundlegenden Probleme verteilter Systeme und die klassischen verteilten Algorithmen und Programmierparadigmen. Sie können klassische und aktuelle Standards verteilter Programmierung praktisch anwenden.				
4	Voraussetzung für die Teilnahme Empfohlen: „Computer-Netzwerke und verteilte Systeme“				
5	Prüfungsform Fachprüfung schriftlich/mündlich 60-120/30 min.				

6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Autonome Systeme M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik B.Sc. Informationssystemtechnik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Literaturempfehlungen werden kontinuierlich aktualisiert, Beispiele für verwendete Literatur könnten sein: <ul style="list-style-type: none"> • George Coulouris, Jean Dollimore, Tim Kindberg: Distributed Systems. Concepts and Design (Gebundene Ausgabe) 832 Seiten, Addison Wesley; Auflage: 4th (14. Juni 2005), ISBN: 0321263545 • M. Boger: Java in verteilten Systemen, 1999, dpunkt-Verlag, Heidelberg, ISBN: 3932588320 • G. Tel: Introduction to Distributed Algorithms, 2nd Ed 2001, Cambridge University Press, ISBN: 0521794838 • A. Tanenbaum, M.v.Steen, Verteilte Systeme: Grundlagen und Paradigmen, Pearson Studium 2003, ISBN: 3827370574 • A. Tanenbaum: Computernetzwerke. 4te Auflage. Pearson Studium 2003, ISBN-10: 3827370469 • J. Kurose, K. Ross: Computer Networking, 1. Ed. 2000, Adison-Wesley. ISBN: 0201477114 • L. Peterson, B. Davie, Computernetze, 1. Aufl. 2000, dpunkt Heidelberg, ISBN: 393258869X • Hammerschall, U.: Verteilte Systeme und Anwendungen. Pearson, München 2005, ISBN: 3827370965
10	Kommentar

Modulbeschreibung

Modulname Netzsicherheit					
Modul Nr. 20-00-0512	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0512-iv	Netzsicherheit	6	integrierte Lehrveranstaltung	4
2	<p>Lerninhalt</p> <p>Die integrierte Veranstaltung Netzsicherheit umfasst Sicherheits-Prinzipien und -Praxis in Telekommunikationsnetzen und dem Internet. Die grundlegenden Verfahren aus dem Bereich IT Sicherheit und Kryptographie werden auf den Bereich der Kommunikationsnetze übertragen. Hierbei verfolgen wir einen Top-down Ansatz. Beginnend mit der Anwendungsschicht erfolgt eine detaillierte Betrachtung von Prinzipien und Protokollen zur Absicherung von Netzen. Ergänzend zu etablierten Mechanismen werden ausgewählte aktuelle Entwicklungen im Bereich Netzsicherheit erläutert.</p> <p>Lerninhalte:</p> <ul style="list-style-type: none"> - Netzsicherheit: Einführung, Motivation und Herausforderungen - Grundlagen: Ein Referenzmodell für Netzsicherheit, Sicherheitsstandards für Netze und das Internet, Bedrohungen, Angriffe, Sicherheitsdienste und -mechanismen - Kryptographische Grundlagen zur Absicherung von Netzen: Symmetrische Kryptographie und deren Anwendung in Netzen, asymmetrische Kryptographie und deren Anwendung in Netzen, unterstützende Mechanismen zur Implementierung von Sicherheitslösungen - Sicherheit auf der Anwendungsschicht - Sicherheit auf der Transportschicht - Sicherheit auf der Vermittlungsschicht - Sicherheit auf der Sicherungsschicht - Sicherheit auf der Bitübertragungsschicht und physische Sicherheit - Angewandte Netzsicherheit: Firewalls, Intrusion Detection Systeme - Ausgewählte Themen der Netzsicherheit 				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an der Veranstaltung haben die Studierenden ein umfassendes Wissen auf dem Gebiet der Netzsicherheit mit dem Schwerpunkt auf Internetsicherheit. Sie können die wichtigsten Grundlagen der IT Sicherheit sowie der Kryptographie auf den Bereich Kommunikationsnetze übertragen und anwenden. Die Studierenden können die wichtigsten Basistechnologien zur Absicherung von Netzen unterscheiden. Sie weisen ein tiefgehendes Verständnis von Sicherheitsmechanismen auf</p>				

	den unterschiedlichen Protokollschichten auf (Anwendungsschicht, Transportschicht, Vermittlungsschicht, Sicherungsschicht, physikalische Schicht). Somit sind sie in der Lage, die Charakteristiken und Grundprinzipien des Problemraumes Netzsicherheit detailliert zu erläutern und weisen auf diesem Feld ein fundiertes Wissen in Praxis und Theorie auf. Darüber hinaus können sie aktuelle Entwicklungen im Bereich Netzsicherheit erläutern (z.B. Sicherheit in peer-to-peer Systemen, Sicherheit in mobilen Netzen, etc.). Die Übung vertieft das theoretische Wissen durch Literatur-, Rechen- und praktische Implementierungs-/Anwendungsübungen.
4	Voraussetzung für die Teilnahme Empfohlen: Grundlagen der IT-Sicherheit, Kryptographie und Kommunikationsnetze
5	Prüfungsform Fachprüfung schriftlich/mündlich 60-120/30 min.
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Distributed Software Systems M.Sc. Autonome Systeme M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik B.Sc. Informationssystemtechnik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Charlie Kaufman, Radia Perlman, Mike Speciner: Network Security – Private Communication in a Public World, 2nd Edition, Prentice Hall, 2002, ISBN: 978-0-14-046019-6; weiterhin ausgewählte Buchkapitel und ausgewählte wissenschaftliche Veröffentlichungen
10	Kommentar

Modulbeschreibung

Modulname					
Concepts and Technologies for Distributed Systems and Big Data Processing					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0951	3 CP	90 h	60 h	1 Semester	Jedes 2. Semester
Sprache			Modulverantwortliche Person		
Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0951-iv	Concepts and Technologies for Distributed Systems and Big Data Processing	0	Integrierte Veranstaltung	2
2	Lerninhalt				
	<p>The course provides an overview of recent advances in distributed systems for Big Data processing. The course starts presenting computational models for high throughput batch processing like MapReduce. Next, we will introduce software engineering techniques for distributed systems such as REST and component-based architectures. We will then cover low latency real time stream processing and complex event processing. Finally, we will present advanced topics in distributed data-intensive systems, such as geodistribution and security.</p> <p>The course focuses both on the fundamental concepts as well as on the concrete technologies and applications of the aforementioned techniques to real-world case studies.</p>				
3	Qualifikationsziele / Lernergebnisse				
	<ul style="list-style-type: none"> - The students are familiar with basic concepts and technologies on distributed systems and big data and are able to implement basic cloud based/distributed applications. - The students are familiar with the fundamental computational models behind recent advances in distributed systems, such as models for batch processing of massive data amounts, stream processing and complex event processing. - The students are familiar with selected advanced topics on big data, including security and geolocalization. - The students know about real-world case studies that apply the concepts and the technologies presented during the course. 				
4	Voraussetzung für die Teilnahme				
	<p>Empfohlen: This course is targeted at master students.</p>				

5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0951-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Pass exam (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0951-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Cryptocurrencies					
Modul Nr. 20-00-1010	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1010-iv	Cryptocurrencies	0	Integrierte Veranstaltung	4
2	Lerninhalt Konzepte von Kryptowährungen: - Kryptographische Bausteine: Kryptographische Hashfunktionen, Signaturen, Blinde Signaturen, Commitments - Chaum's eCash Verfahren und dessen Sicherheitseigenschaften - Verteilte System und Fehlermodelle - Broadcast- und Konsensusverfahren - Einführung in Bitcoin und dessen Konsensusverfahren - Mining Bitcoins und sicheres Speichern von Bitcoins - Anonymität in Kryptowährungen - Angriffe auf Kryptowährungen - Smart Contracts und Anwendungen - Skalierbarkeit von Kryptowährungen - Altcoins and Blockchain ecosystem				
3	Qualifikationsziele / Lernergebnisse Studierende verstehen nach erfolgreichem Besuch der Veranstaltung die technischen und theoretischen Grundkonzepte von kryptographischen Währungen. Insbesondere lernen sie: <ul style="list-style-type: none"> • Den Umgang mit kryptographischen Bausteinen und deren formale Sicherheitsanalyse mittels Beweise • Die Entwicklung kryptographischer Protokolle und verteilter Systeme • Die Grundkonzepte Blockchain-basiertere Kryptowährungen insbesondere der Konsensus Mechanismen • Mögliche Angriffe auf Bitcoin und die zugrundeliegende Technologie • Die Grundkonzepte der Entwicklung von Smart Contracts und deren Anwendung • Neue Lösungsansätze zur Verbesserung von Kryptowährungen hinsichtlich Anonymität, Skalierbarkeit und Sicherheit • Eine Übersicht über verschiedene Altcoins und deren Vorteile/Nachteile 				
4	Voraussetzung für die Teilnahme Empfohlen: Besuch der Vorlesung "Introduction to Cryptography / Einführung in die Kryptographie" bzw. entsprechende Kenntnisse aus anderen Studiengängen				

5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1010-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1010-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B. Sc. Informatik M. Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur Literaturempfehlungen werden kontinuierlich aktualisiert, ein Beispiel für verwendete Literatur könnte sein: Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller & Steven Goldfeder
10	Kommentar

Modulbeschreibung

Modulname Skalierbares Datenmanagement					
Modul Nr. 20-00-1017	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1017-iv	Skalierbares Datenmanagement	0	Integrierte Veranstaltung	4
2	Lerninhalt Diese Vorlesungen ist eine Einführung in die Basiskonzepte und die wesentlichen Paradigmen für skalierbare Datenmanagement-Systeme. Der Fokus der Vorlesung ist auf die systemorientieren Aspekten und Interna solcher Systeme gerichtet, um große Datenmengen zu speichern, zu ändern, und zu analysieren. Themen der Vorlesung sind: Database Architectures Parallel and Distributed Databases Data Warehousing MapReduce and Hadoop Spark and its Ecosystem Optional: NoSQL Databases, Stream Processing, Graph Databases, Scalable Machine Learning				
3	Qualifikationsziele / Lernergebnisse Nach dem Kurs sollen die Studierenden einen Überblick über die wichtigsten Konzepte, Algorithmen und System-Aspekte für skalierbare Datenmanagement-Systeme erworben haben. Das Hauptziel ist es, dass die Studierenden das Wissen besitzen, solche Systeme zu designen und zu entwickeln, inklusive praktischer Übungen auf Basis von bestehenden Systemen wie Spark.				
4	Voraussetzung für die Teilnahme Empfohlen: Programmierkenntnisse in C++ and Java Informationsmanagement (20-00-0015-iv) Optional: Foundations of Distributed Systems (20-00-0998-iv)				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1017-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none">• [20-00-1017-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulhandbuch
M. Sc. Distributed Software Systems

Wahlbereich
Networking and Systems Software

Modulbeschreibung

Modulname TK3: Ubiquitous / Mobile Computing					
Modul Nr. 20-00-0120	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0120-iv	TK3: Ubiquitous / Mobile Computing	6	integrierte Lehrveranstaltung	4
2	Lerninhalt				
	<ul style="list-style-type: none"> • Kenntnis technischer Grundlagen der Mobilkommunikation • Kenntnis wichtiger Herausforderungen, Thesen und Modelle des Ubiquitous Computing • Methodenwissen über aktuelle Ansätze des Ubiquitous Computing <p>Stoffplan:</p> <ul style="list-style-type: none"> • Einführung in Ubiquitous Computing <ul style="list-style-type: none"> ○ Definitionen und Bedeutung ○ Herausforderungen und Klassifikation ○ Wichtiges zur historischen Entwicklung (Mark Weiser u.a.) ○ Von Terminologie zu Taxonomie ○ Referenzarchitekture • Mobilkommunikation als 'Enabling Technology' <ul style="list-style-type: none"> ○ Einordnung und physikalische Grundlagen ○ Elementare Mehrfachzugriffs- und Modulationsverfahren ○ Zellulare Weitverkehrsnetze: von GSM bis LTE ○ Drahtlose lokale Netze: WLAN, Bluetooth und ZigBee • Internet-of-Things: RFID und Smart Items <ul style="list-style-type: none"> ○ Grundlagen von RFID-Systemen ○ EPC und Smart Items ○ NFC: Nahfeld-Kommunikation • Service Discovery und Cloudlets <ul style="list-style-type: none"> ○ Grundlagen der Skalierbarkeit im Ubiquitous Computing ○ Service Discovery: Grundlagen ○ Service Discovery: konkurrierende Ansätze ○ Cloudlets: Forschungsansätze für Ubiquitous Cloud Computing 				

	<ul style="list-style-type: none"> ● Context- und Location Aware Computing <ul style="list-style-type: none"> ○ Grundlagen der Adaptivität in Ubiquitous Computing ○ Kontext-Modelle und Ansätze für Context-Aware Computing ○ Technische Grundlagen der Ortsbestimmung und Location Awareness ● Mensch-Maschine-Interaktion für Ubiquitous Computing <ul style="list-style-type: none"> ○ Einführung: Ease-of-Use und Post-Desktop-Interaktion ○ Interaction Design und Multimediale Interaktion ○ Grundlagen von Multitouch-Systemen ○ Pen-and-Paper-Interaktion und Tangible Interaction ○ UI Design: Evaluationstechniken ○ Systematisches UI Engineering ● Privatsphäre und Vertrauen im Ubiquitous Computing <ul style="list-style-type: none"> ○ Einführung in Privacy und rechtliche Grundlagen ○ Zum Wesen personenbezogener Daten ○ Privacy-Enhancing Technologies (PETs) und Anonyme Kommunikation ○ Einführung in Vertrauen und Reputation ○ Vertrauensmodelle und Computational Trust ○ Trust-Management-Systeme
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Studierende kennen nach erfolgreichem Besuch der Veranstaltung die technische Grundlage mobiler Kommunikation. Sie verstehen die grundlegenden Herausforderungen von Ubiquitous Computing. Sie kennen aktuelle Ansätze um diese Herausforderungen zu lösen. Sie sind außerdem in der Lage ihre Kenntnisse auf aktuelle Probleme anzuwenden.</p>
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: „Computer-Netzwerke und verteilte Systeme“</p>
5	<p>Prüfungsform</p> <p>Fachprüfung schriftlich/mündlich 60-120/30 min.</p>
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Modulabschlussprüfung (100%)</p>
7	<p>Benotung</p> <p>Standard</p> <p>In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.</p>
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit</p>

	<p>M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik M.Sc. Informationssystemtechnik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur Literaturempfehlungen werden kontinuierlich aktualisiert, Beispiele für verwendete Literatur könnten sein:</p> <p>A Primärliteratur: Handbook of Research: Ubiquitous Computing Technology for Real Time Enterprises edited by Prof. Dr. Max Mühlhäuser, Dr. Iryna Gurevych, 2008, Information Science Reference, ISBN-10: 1599048329</p> <p>B Sekundärliteratur: 1. F. Adelstein, S. Gupta et al.: Fundamentals of Mobile & Pervasive Computing McGraw Hill 2004, 2. Stefan Poslad: Ubiquitous Computing, Wiley 2009, ISBN 978-0-470-03560-3 3. Kapitel Mobilkommunikation: M. Sauter: Grundkurs Mobile Kommunikationssysteme: UMTS, HSDPA und LTE, GSM, GPRS und Wireless LAN; Vieweg-Teubner Studium 2010 4. J. Krumm (Ed.): Ubiquitous Computing Fundamentals, CRC Press 2010 D. Cook, S. Das (Ed.): Smart Environments, Wiley 2005</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Mobile Netze					
Modul Nr. 20-00-0748	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0748-iv	Mobile Netze	0	Integrierte Veranstaltung	4
2	<p>Lerninhalt</p> <p>Mobilkommunikation und drahtlose Kommunikationstechniken haben sich in den letzten Jahren rapide weiterentwickelt. Die integrierte Lehrveranstaltung erläutert Charakteristiken und Grundprinzipien mobiler Netze, und praktische Lösungsansätze werden vorgestellt. Der Fokus der Veranstaltung liegt hierbei auf der Vermittlungsschicht (Netzwerkschicht). Zusätzlich zum Stand der Technik werden in der Veranstaltung aktuelle Forschungsfragen diskutiert und Methoden und Werkzeuge zur systematischen Behandlung dieser Fragen erläutert. Die Inhalte werden in Übungseinheiten vertieft.</p> <p>Lerninhalte:</p> <ul style="list-style-type: none"> - Einleitung: Drahtlose und mobile Kommunikation: Anwendungen, Geschichte, Marktchancen - Überblick über drahtlose Kommunikation: Drahtlose Übertragung, Frequenzen und Frequenzregulierung, Signale, Antennen, Signalausbreitung, Multiplex, Modulation, Spreizband-Technik, Zellulare Systeme - Medienzugriff: SDMA, FDMA, CDMA, TDMA (Feste Zuordnung, Aloha, CSMA, DAMA, PRMA, MACA, Kollisionsvermeidung, Polling) - Drahtlose Lokale Netze (Wireless LAN): IEEE 802.11 Standard inklusive Bitübertragungsschicht, Sicherungsschicht und Zugriffverfahren, Dienstgüte, Energieverwaltung - Drahtlose Stadtnetze, drahtlose Mesh Netze, IEEE 802.16 Standard inklusive Betriebsmodi, Medienzugriff, Dienstgüte, Ablaufkoordination - Mobilität auf der Netzwerkschicht: Konzepte zur Mobilitätsunterstützung, Mobile IP - Ad hoc Netze: Terminologie, Grundlagen und Applikationen, Charakteristika von Ad hoc Kommunikation, Ad hoc Routing Paradigmen und Protokolle - Leistungsbewertung von mobilen Netzen: Einführung in die Leistungsbewertung, systematischer Ansatz/häufige Fehler und wie man sie vermeiden kann, experimentelles Design und Analyse - Mobilität auf der Transportschicht: Varianten von TCP (Indirect TCP, Snoop TCP, Mobile TCP, Wireless TCP) - Mobilität auf der Anwendungsschicht: Anwendungen für mobile Netze und drahtlose Sensornetze 				

3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an der Veranstaltung haben Studierende ein umfassendes Wissen der Funktionsweise mobiler Kommunikationsnetze. Sie können die wichtigsten Grundlagen drahtloser Kommunikationstechniken erläutern. Die Studierenden können weiterhin Medienzugriffsverfahren kategorisieren und die Funktionsweise dieser Verfahren im Detail erklären. Insbesondere weisen sie ein tiefgehendes Verständnis von Verfahren auf Vermittlungsschicht und Transportschicht auf, mit Schwerpunktsetzung auf Ad hoc und Mesh Netze. Die Studierenden erlangen Wissen über die Zusammenhänge zwischen unterschiedlichen Protokollschichten und können ihr erworbenes Wissen auf die methodische Analyse von realen Kommunikationssystemen anwenden. Sie sind somit in der Lage, die Charakteristiken und Grundprinzipien des Problemraumes drahtloser und mobiler Kommunikation detailliert zu erläutern und weisen auf diesem Feld ein fundiertes Wissen in Praxis und Theorie auf. Die Übungsteile der integrierten Veranstaltung vertiefen das theoretische Wissen durch Literatur-, Rechen- und praktische Implementierungs-/Anwendungsübungen.</p>
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Grundlagen der Kommunikationsnetze</p>
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0748-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Prüfung (100%)</p>
7	<p>Benotung</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0748-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p> <p>Ausgewählte Buchkapitel und ausgewählte wissenschaftliche Veröffentlichungen</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Advanced Data Management Systems					
Modul Nr. 20-00-1039	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1039-iv	Advanced Data Management Systems	0	Integrierte Veranstaltung	4
2	<p>Lerninhalt</p> <p>Dies ist eine fortgeschrittene Vorlesung aus dem Bereich der Architektur und Implementierung moderner Datenbanksysteme mit dem speziellen Fokus auf System-orientieren Aspekten und Interna solcher Systeme. Mögliche Themengebiete die in der Vorlesung behandelt werden sind: moderne Hardwaretechnologien für das Datenbanksysteme, Optimierungen für Hauptspeicherdatenbanken, Parallelisierungsstrategien und Approximative Anfrageausführung usw.</p> <p>Es wird erwartet, dass für jede Vorlesung aktuelle Veröffentlichungen (SIGMOD, VLDB, etc.) vorher gelesen werden. Die Hauptideen ausgewählter Veröffentlichungen werden in Programmierprojekten umgesetzt. Die Endnote des Kurses basiert auf den Programmierprojekten. Es gibt keine Klausur.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreichem Abschluss der Vorlesung haben Studenten folgende Lernziele erreicht:</p> <ul style="list-style-type: none"> - Vertieftes Verständnis von aktuellen Techniken für das Design von modernen Datenbanksystemen - Diskussion von Vor- und Nachteilen dieser Techniken mit dem Fokus auf möglichen Verbesserungen - Implementierung von einzelnen Techniken und experimentelle Evaluierung dieser Techniken zum Vergleich von Designalternativen 				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Solide Programmierkenntnisse in C and C++ Skalierbares Datenmanagement (20-00-1017-iv) Informationsmanagement (20-00-0015-iv)</p>				
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-1039-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Prüfung (100%)</p>				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none">• [20-00-1039-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Kommunikationsnetze I					
Modul Nr. 18-sm-1010	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-1010-vl	Kommunikationsnetze I		Vorlesung	3
	18-sm-1010-ue	Kommunikationsnetze I		Übung	1
2	<p>Lerninhalt</p> <p>In dieser Veranstaltung werden die Technologien, die Grundlage heutiger Kommunikationsnetze sind, vorgestellt und analysiert.</p> <p>Die Vorlesung deckt grundlegendes Wissen über Kommunikationssysteme ab und betrachtet im Detail die 4 unteren Schichten des ISO-OSI-Modells: Bitübertragungsschicht, Sicherungsschicht, Vermittlungsschicht und Teile der Transportschicht.</p> <p>Die Bitübertragungsschicht, die zuständig ist für eine adäquate Übertragung über einen Kanal, wird kurz betrachtet. Danach werden fehlertolerante Kodierung, Flusskontrolle und Zugangskontrollverfahren (Medium access control) der Sicherungsschicht betrachtet. Anschließend wird die Netzwerkschicht behandelt. Der Fokus liegt hier auf Wegfindungs- und Überlastkontrollverfahren. Abschließend werden grundlegende Funktionen der Transportschicht betrachtet. Dies beinhaltet UDP und TCP- Das Internet und dessen Funktionsweise wird im Laufe der Vorlesung detailliert betrachtet.</p> <p>Themen sind:</p> <ul style="list-style-type: none"> - ISO-OSI und TCP/IP Schichtenmodelle - Aufgaben und Eigenschaften des Bitübertragungsschicht - Kodierungsverfahren der Bitübertragungsschicht - Dienste und Protokolle der Sicherungsschicht - Flußkontrolle (sliding window) - Anwendungen: LAN, MAN, High-Speed LAN, WAN - Dienste der Vermittlungsschicht - Wegfindungsalgorithmen - Broadcast- und Multicastwegfindung - Überlastbehandlung - Adressierung - Internet Protokoll (IP) 				

	<ul style="list-style-type: none"> - Netzbrücken - Mobile Netze - Services und Protokolle der Transportschicht - TCP, UDP
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Diese Vorlesung betrachtet Grundfunktionalitäten, Services, Protokolle, Algorithmen und Standards von Kommunikationssystemen. Vermittelt Kompetenzen sind grundlegendes Wissen über die vier unteren Schichten des ISO-OSI-Modells: Bitübertragungsschicht, Sicherungsschicht, Vermittlungsschicht und Transportschicht. Desweiteren wird Grundwissen über Kommunikationssysteme vermittelt. Besucher der Vorlesung werden Funktionen heutiger Netzwerktechnologien und des Internets erlernen.</p>
4	<p>Voraussetzung für die Teilnahme</p>
5	<p>Prüfungsform</p> <p>Fachprüfung schriftlich/mündlich 60-120/30 min.</p>
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Modulabschlussprüfung (100%)</p>
7	<p>Benotung</p> <p>Standard</p> <p>In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.</p>
8	<p>Verwendbarkeit des Moduls</p> <p>Wi-CS, Wi-ETiT, BSc CS, BSc ETiT, BSc iST</p>
9	<p>Literatur</p> <p>Ausgewählte Kapitel aus folgenden Büchern:</p> <ul style="list-style-type: none"> - Andrew S. Tanenbaum: Computer Networks, 5th Edition, Prentice Hall, 2010 - Andrew S. Tanenbaum: Computernetzwerke, 3. Auflage, Prentice Hall, 1998 - Larry L. Peterson, Bruce S. Davie: Computer Networks: A System Approach, 2nd Edition, Morgan Kaufmann Publishers, 1999 - Larry L. Peterson, Bruce S. Davie: Computernetze, Ein modernes Lehrbuch, 2. Auflage, Dpunkt Verlag, 2000 - James F. Kurose, Keith W. Ross: Computer Networking: A Top-Down Approach Featuring the Internet, 2nd Edition, Addison Wesley-Longman, 2002 - Jean Walrand: Communication Networks: A First Course, 2nd Edition, McGraw-Hill, 1998
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Kommunikationsnetze II					
Modul Nr. 18-sm-2010	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Wintersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-2010-vl	Kommunikationsnetze II		Vorlesung	3
	18-sm-2010-ue	Kommunikationsnetze II		Übung	1
2	Lerninhalt				
	<p>Die Vorlesung Kommunikationsnetze II umfasst die Konzepte der Computervernetzung und -telekommunikation mit dem Fokus auf dem Internet. Beginnend mit der Geschichte werden in der Vorlesung vergangene, aktuelle und zukünftige Aspekte von Kommunikationsnetzen behandelt. Zusätzlich zu bekannten Protokollen und Technologien wird eine Einführung in Neuentwicklungen im Bereich von Multimedia Kommunikation (u.a. Video Streaming, P2P, IP-Telefonie, Cloud Computing und Service-orientierte Architekturen) gegeben. Die Vorlesung ist als Anschlussvorlesung zu Kommunikationsnetze I geeignet.</p> <p>Themen sind:</p> <ul style="list-style-type: none"> - Grundlagen und Geschichte von Kommunikationsnetzen (Telegrafie vs. Telefonie, Referenzmodelle, ...) - Transportschicht (Adressierung, Flusskontrolle, Verbindungsmanagement, Fehlererkennung, Überlastkontrolle, ...) - Transportprotokolle (TCP, SCTP) - Interaktive Protokolle (Telnet, SSH, FTP, ...) - Elektronische Mail (SMTP, POP3, IMAP, MIME, ...) - World Wide Web (HTML, URL, HTTP, DNS, ...) - Verteilte Programmierung (RPC, Web Services, ereignisbasierte Kommunikation) - SOA (WSDL, SOAP, REST, UDDI, ...) - Cloud Computing (SaaS, PaaS, IaaS, Virtualisierung, ...) - Overlay-Netzwerke (unstrukturierte P2P-Systeme, DHT-Systeme, Application Layer Multicast, ...) - Video Streaming (HTTP Streaming, Flash Streaming, RTP/RTSP, P2P Streaming, ...) - VoIP und Instant Messaging (SIP, H.323) 				

3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Die Vorlesung Kommunikationsnetze II umfasst die Konzepte der Computervernetzung und -telekommunikation mit dem Fokus auf dem Internet. Beginnend mit der Geschichte werden in der Vorlesung vergangene, aktuelle und zukünftige Aspekte von Kommunikationsnetzen behandelt. Zusätzlich zu bekannten Protokollen und Technologien wird eine Einführung in Neuentwicklungen im Bereich von Multimedia Kommunikation (u.a. Video Streaming, P2P, IP-Telefonie, Cloud Computing und Service-orientierte Architekturen) gegeben. Die Vorlesung ist als Anschlussvorlesung zu Kommunikationsnetze I geeignet.</p>
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Grundlegende Kurse der ersten 4 Semester werden benötigt. Die Vorlesung Kommunikationsnetze I wird empfohlen. Das Theoriewissen aus der Vorlesung Kommunikationsnetze II wird in praktischen Programmierübungen vertieft. Grundlegende Programmierkenntnisse sind daher hilfreich.</p>
5	<p>Prüfungsform</p> <p>Fachprüfung</p>
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Modulabschlussprüfung (100%)</p>
7	<p>Benotung</p> <p>Standard</p>
8	<p>Verwendbarkeit des Moduls</p> <p>MSc ETiT, MSc iST, Wi-ETiT, CS, Wi-CS</p>
9	<p>Literatur</p> <p>Ausgewählte Kapitel aus folgenden Büchern:</p> <ul style="list-style-type: none"> - Andrew S. Tanenbaum: Computer Networks, 5th Edition, Prentice Hall, 2010 - James F. Kurose, Keith Ross: Computer Networking: A Top-Down Approach, 6th Edition, Addison-Wesley, 2009 - Larry Peterson, Bruce Davie: Computer Networks, 5th Edition, Elsevier Science, 2011
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Software Defined Networking					
Modul Nr. 18-sm-2280	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Prof. Dr.-Ing. Ralf Steinmetz		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-2280-ue	Software Defined Networking	0	Übung	2
	18-sm-2280-vl	Software Defined Networking	0	Vorlesung	2
2	Lerninhalt Der Kurs behandelt Themen aus dem Bereich Software Defined Networking: <ul style="list-style-type: none"> • SDN Data Plane • SDN Control Plane • SDN Application Plane • Network Function Virtualization • Network Virtualization and Slicing • QoS and QoE in Software Defined Networks 				
3	Qualifikationsziele / Lernergebnisse Studierende erhalten einen vertieften Einblick in Software Defined Networking, sowie grundlegende Technologien und Anwendungen.				
4	Voraussetzung für die Teilnahme Grundlegende Kurse der ersten 4 Semester werden benötigt. Die Vorlesungen in Kommunikationsnetze I und II werden empfohlen.				
5	Prüfungsform Modulabschlussprüfung: <ul style="list-style-type: none"> • Modulprüfung (Fachprüfung, fakultativ, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten				
7	Benotung Modulabschlussprüfung: <ul style="list-style-type: none"> • Modulprüfung (Fachprüfung, fakultativ, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls MSc ETiT, BSc/MSc iST, MSc Wi-ETiT, CS, Wi-CS				
9	Literatur Lehrbücher gemäß Ankündigung. Folienskript der Vorlesung und Artikelkopien nach Bedarf.				
10	Kommentar				

Modulhandbuch
M. Sc. Distributed Software Systems

Wahlbereich
Formal Methods, Programming Languages
and Software Engineering

Modulbeschreibung

Modulname Konzepte der Programmiersprachen					
Modul Nr. 20-00-0072	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0072-iv	Konzepte der Programmiersprachen	6	integrierte Lehrveranstaltung	4
2	Lerninhalt Die wesentlichen Konzepte von Programmiesprachen. Insbesondere werden dazu Programmiersprachen in ihre Basiskonzepte aufgespalten und diese detailliert betrachtet: <ul style="list-style-type: none"> • Die Rolle von Syntax • Funktionen • Meta-Interpreter • Rekursion • Verzögerte Auswertung • Zustand und Seiteneffekte • Continuations • Statische Typsysteme • Domain-spezifische Sprachen und Makros • Objektorientierte Programmierung 				
3	Qualifikationsziele / Lernergebnisse Nach dem erfolgreichen Abschluss der Veranstaltung verfügen die Studierenden über die folgenden Fähigkeiten: <ul style="list-style-type: none"> • Sie können die entscheidenden Merkmale von Programmiersprachen benennen und im konkreten Fall identifizieren; • die Studierenden sind mit den wesentlichen theoretischen Konzepten von Programmiersprachen vertraut; • sie können verschiedene Vorgehensweisen bei der Implementierung von Programmiersprachen benennen und einfache Programmiersprachen umsetzen; • die Studierenden verstehen, wie Programmiersprachen den Lösungsraum von Problemen beeinflussen; sie können die Auswirkung der Wahl einer Programmiersprache auf die Softwareentwicklung abschätzen; • die Studierenden sind in der Lage stereotypische Kategorisierungen von Programmiersprachen zu überwinden. 				
4	Voraussetzung für die Teilnahme Empfohlen: Funktionale und Objektorientierte Programmierkonzepte				

5	Prüfungsform Fachprüfung schriftlich/mündlich 60-120/30 min.
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Distributed Software Systems B.Sc. Computational Engineering M.Sc. Computational Engineering M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur <ul style="list-style-type: none"> ● S. Krishnamurthi: Programming Languages - Application and Interpretation ● M. Scott: Programming Language Pragmatics, Morgan Kaufmann ● D. Friedman et al.: Programming Language Essentials, MIT Press
10	Kommentar

Modulbeschreibung

Modulname IT-Lösungen durch praxiserprobtes Software Engineering					
Modul Nr. 20-00-0635	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0635-iv	IT-Lösungen durch praxiserprobtes Software Engineering	0	Integrierte Veranstaltung	2
2	Lerninhalt - Modellierung mit UML bzw. DSL und Code-Generierung				
3	Qualifikationsziele / Lernergebnisse Die Teilnehmer lernen theoretisch und praktisch - anhand von Fallbeispielen aus der Praxis - wie Software-Engineering zur Erarbeitung von IT-Lösungen eingesetzt wird. Dabei werden moderne, praxiserprobte Konzepte zur Erstellung von IT-Lösungen vorgestellt, zum Beispiel Modellierung (Geschäftsprozesse, UML, DSL), Generierung und Testautomatisierung. Die Teilnehmer können die Wirtschaftlichkeit von IT-Projekten bewerten, praxiserprobte Projektmanagement-Pattern einsetzen und lernen die umgebenden Rahmenbedingungen einer IT-Organisation sowie die Rolle des CIO in einem Unternehmen als Berater der Fachbereiche kennen. Sie beherrschen das Anforderungsmanagement und den Lösungsentwurf, insbesondere für mobile Anwendungen und SAP-Lösungen. Die Veranstaltung wird durch eingeladene Vorträge von Experten aus der Praxis ergänzt.				
4	Voraussetzung für die Teilnahme Empfohlen: Funktionale und objektorientierte Programmierkonzepte Algorithmen und Datenstrukturen Einführung in Software Engineering				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0635-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0635-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Software Engineering - Design and Construction					
Modul Nr. 20-00-0341	Kreditpunkte 8 CP	Arbeitsaufwand 240 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Wintersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0341-iv	Software Engineering - Design and Construction	8	integrierte Lehrveranstaltung	4
2	Lerninhalt Der primäre Inhalt der Veranstaltung ist der Entwurf modularer Software, um wartbare, wiederverwendbare und erweiterbare Softwaresysteme zu erhalten. Integraler Bestandteil der Veranstaltung ist die Diskussion der Beziehung zwischen den Eigenschaften fortschrittlicher Programmiersprachen und dadurch möglicher Entwurfsalternativen. Weiterhin wird die Auswirkung der Programmiersprache auf den Entwurf eines Softwaresystems als Ganzes besprochen. Die Vorlesung behandelt insbesondere: <ul style="list-style-type: none"> • Prinzipien des Klassenentwurfs unter Verwendung fortgeschrittener Entwurfsmuster und fortschrittlicher Programmiersprachen; • Prinzipien des Entwurfs auf Paketebene; • Architekturelle Stile; • Dokumentation des Entwurfs; • Refactorings existierender Software; • Metriken zur Evaluierung von Entwürfen. 				
3	Qualifikationsziele / Lernergebnisse Nach dem erfolgreichen Abschluss der Lehrveranstaltung sind Studierende in der Lage die folgenden Aufgaben durchzuführen: <ul style="list-style-type: none"> • Sie können den Entwurf existierender Systeme in Hinblick auf ihre Modularität analysieren und ggf. Refactorings vorschlagen, die der Verbesserung bzw. Wiederherstellung selbiger dienen. • Sie verstehen die mittel- und langfristigen Auswirkung nicht-modularer Softwaresysteme. • Sie kennen fortgeschrittene Entwurfsmuster und können diese in existierendem Code identifizieren und auch einsetzen, um neue Probleme zu lösen. • Sie kennen etablierte architekturelle Stile und können diese einsetzen. • Sie verstehen, dass die Lösung eines Entwurfsproblems von der gewählten Programmiersprache abhängt und sind in der Lage entsprechende Entscheidungen kritisch zu hinterfragen. 				
4	Voraussetzung für die Teilnahme Empfohlen: Successful completion of the lecture Software Engineering				

5	Prüfungsform Fachprüfung schriftlich/mündlich 60-120/30 min.
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard In dieser Vorlesung findet eine Anrechnung von vorlesungsbegleitenden Leistungen statt, die lt. §25(2) der 4. Novelle der APB und den vom FB 20 am 02.10.2012 beschlossenen Anrechnungsregeln zu einer Notenverbesserung um bis zu 1.0 führen kann.
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Autonome Systeme M.Sc. Distributed Software Systems B.Sc. Computational Engineering M.Sc. Computational Engineering M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik M.Sc. Informationssystemtechnik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur <ul style="list-style-type: none"> • Bass, L.; Clements, P.; Kazman, R. ; Software Architecture in Practice, Addison-Wesley • Booch, G. Object-Oriented Analysis and Design with Applications. Addison-Wesley • Budd, T. Introduction to Object-Oriented Programming. 2nd. ed., Addison-Wesley • Buschmann, F. et al. Pattern-Oriented Software Architecture: A System of Patterns. John Wiley & Sons. • Czarnecki, K. and Eisenecker, U. Generative Programming. Addison-Wesley. • Garland, D. and Shaw, M. Software Architecture: Perspectives on an Emerging Discipline. Prentice Hall. • Gamma, E. et al. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley. • Martin, Robert. Agile Software Development. Principles, Patterns, and Practices. Pearson US Imports & PHIPes. • Riel, A. Object-Oriented Design Heuristics. Addison-Wesley.
10	Kommentar

Modulbeschreibung

Modulname Formale Methoden der Informationssicherheit					
Modul Nr. 20-00-0362	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0362-iv	Formale Methoden der Informationssicherheit	9	integrierte Lehrveranstaltung	6
2	Lerninhalt <ul style="list-style-type: none"> • formale Modellierung sicherheitskritischer Systeme in Prädikatenlogik • Theoretische Grundlagen von Zugriffskontrollen und Informationsflusskontrollen • formale Modellierung von Sicherheitseigenschaften in Prädikatenlogik • Unterscheidung von qualitativen und quantitativen Sicherheitseigenschaften • Entscheidbarkeits- und Komplexitätsresultate für Sicherheitseigenschaften • Verifikation von Sicherheitsgarantien in verteilten Systemen • Auswirkung von Komposition und Verfeinerung auf Sicherheitsgarantien • formale Sprachen zur Beschreibung von Sicherheitspolitiken und deren Semantik • Zertifizierung sicherheitskritischer Systeme 				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende relevante formale Sicherheitsmodelle und Analysetechniken. Sie verstehen fundamentale Unterschiede zwischen verschiedenen Klassen von Sicherheitseigenschaften und das Zusammenspiel zwischen schrittweiser Softwareentwicklung und Sicherheitseigenschaften. Sie können Systeme und Sicherheitsanforderungen formal modellieren und sicherheitsrelevante Aspekte basierend auf formalen Spezifikationen formal analysieren.				
4	Voraussetzung für die Teilnahme Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere grundlegende Logikkenntnisse und Fähigkeit mit formalen Sprachen und Kalkülen umzugehen				
5	Prüfungsform Fachprüfung schriftlich/mündlich 60-120/30 min.				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				
7	Benotung Standard				

8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p> <ul style="list-style-type: none"> • M. Bishop: Computer Security, Addison-Wesley • J. Biskup: Security in Computing Systems, Springer-Verlag • C. P. Pfleeger, S. L. Pfleeger: Security in Computing, Prentice Hall • D. Denning: Cryptography and Data Security, Addison Wesley <p>Die Literaturempfehlungen werden kontinuierlich aktualisiert.</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Programmierung Massiv-Paralleler Prozessoren					
Modul Nr. 20-00-0419	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0419-iv	Programmierung Massiv-Paralleler Prozessoren	0	Integrierte Veranstaltung	4
2	Lerninhalt - Grundlagen massiv-paralleler Hardware mit einem Schwerpunkt auf modernen Beschleunigern - parallele Algorithmen - effiziente Programmierung massiv-paralleler Systeme - praktische Programmierprojekte mit Co-Betreuung durch einen Wissenschaftler aus seiner Anwendungsdomain				
3	Qualifikationsziele / Lernergebnisse Nach dem erfolgreichen Besuch der Veranstaltung sind Studierende dazu in der Lage, Problemstellungen im Kontext massiv-paralleler Systeme zu analysieren. Sie können selbständig neue Anwendungen entwickeln und ihre Performanz systematisch verbessern. Sie verstehen grundlegende parallele Algorithmen und Programmierparadigmen und können sich selbständig aktuelle Literatur erarbeiten.				
4	Voraussetzung für die Teilnahme Solide Programmierkenntnisse in C/C++ Empfohlen: Systemnahe und Parallele Programmierung				
5	Prüfungsform Bausteinbegleitende Prüfung: • [20-00-0419-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: • [20-00-0419-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik B.Sc. Computational Engineering M.Sc. Computational Engineering M.Sc. Wirtschaftsinformatik				

	<p>B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur Wird in der Veranstaltung bekanntgegeben.</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Statische und dynamische Programmanalyse					
Modul Nr. 20-00-0580	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0580-iv	Statische und dynamische Programmanalyse	0	Integrierte Veranstaltung	4
2	Lerninhalt <ul style="list-style-type: none"> - operationelle Semantiken für sequentielle und parallele Programme - Übersicht über Techniken zur statischen und dynamischen Programmanalyse - Abstrakte Interpretation - Datenflussanalysen - Slicing-Techniken - typbasierte Programmanalysen - Konzepte der Laufzeitüberwachung - Implementierungstechniken zur Laufzeitüberwachung - Sprachbasierte Sicherheit - Korrektheit und Präzision von Programmanalysen 				
3	Qualifikationsziele / Lernergebnisse <p>Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende ein Spektrum von unterschiedlichen Programmanalysen. Sie verstehen die Funktionsweise der einzelnen Analysetechniken und verstehen die Unterschiede zwischen diesen. Sie können beurteilen, welche Analysetechnik für welche Problemstellung in Frage kommt und haben die Fähigkeit, die ausgewählte Analysetechnik einzusetzen. Sie können Programmanalysen bezüglich ihrer Präzision und Korrektheit beurteilen. Sie können Programmanalysen auch implementieren und Varianten von bekannten Programmanalysen definieren.</p>				
4	Voraussetzung für die Teilnahme <p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere grundlegende Logikkenntnisse und Fähigkeit, mit formalen Sprachen und Kalkülen umzugehen</p>				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0580-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung				

	Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0580-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname					
Formale Spezifikation und Verifikation von Software					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0794	6 CP	180 h	120 h	1 Semester	Jedes 2. Semester
Sprache			Modulverantwortliche Person		
Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0794-iv	Formale Spezifikation und Verifikation von Software	0	Integrierte Veranstaltung	4
2	<p>Lerninhalt</p> <p>In dieser Vorlesung behandeln wir fortgeschrittene Themen aus dem Gebiet der formalen Spezifikation und deduktiven Verifikation objekt-orientierter Software.</p> <p>Der Kurs deckt insbesondere folgende Themen ab:</p> <ul style="list-style-type: none"> * Spezifikation von Interfaces und Klassen mit Hilfe von Queries, Ghost- und Modellfeldern; * Das "Framing" Problem: Statische und dynamische Frames * Programmlogik und -kalkül als Grundlage der deduktiven Verifikation * Spezifikation und Verifikation rekursiver Methoden und Schleifen * Modulare Verifikation: Sichtbarkeiten, Beweis und Anwendung von Framing-Eigenschaften * Automatische Erzeugung von Schleifeninvarianten und Methodenverträgen <p>Der Kurs behandelt vorwiegend sequentielle Programme. Es werden aber auch aktuelle Ansätze zur Spezifikation und Verifikation nebenläufiger bzw. verteilter Software diskutiert.</p> <p>Für fast alle Themen wird deren praktische Anwendung mit Hilfe geeigneter Tools demonstriert und in den Übungen vertieft.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <ul style="list-style-type: none"> * Erwerbung der Fähigkeit zur Spezifikation komplexer objekt-orientierter Software * Studierende sollen in der Lage sein einen für das vorliegende Problem passenden Spezifikationsansatz auszuwählen und anzuwenden * Studierende sollen in der Lage sein rekursive Methoden und Schleifen zu spezifizieren * Studierende sollen in der Lage sein mit Hilfe von deduktiver Verifikation ihre Programme als korrekt zu beweisen 				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen:</p> <p>Grundlagenwissen über Logik erster Ordnung</p> <p>Inhalt der Vorlesungen</p> <p>Formale Grundlagen der Informatik 2 und 3 (oder vergleichbarer)</p>				
5	Prüfungsform				

	Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0794-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0794-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Applied Static Analysis					
Modul Nr. 20-00-0949	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0949-iv	Applied Static Analysis	0	Integrierte Veranstaltung	2
2	Lerninhalt Foundations of (scalable) static analyses for large(r) software systems; in particular <ul style="list-style-type: none"> - Basic Terminology: - AST, SSA, - Object-/ Field-/ Context-/ Flow-/ Path Sensitivity - (I)CFG - Inter-procedural analyses - ... - stack based intermediate representations (JVM Bytecode) - register based intermediate representations (LLVM IR) - program transformations and native code analyses using LLVM Concrete static analyses and algorithms: <ul style="list-style-type: none"> - Call graph algorithms for libraries and applications - Inter procedural data- and control-flow analyses - IDE/IFDS - Points-to analyses - Escape analyses Applications <ul style="list-style-type: none"> - General software quality analyses - Capability Analysis - Security Vulnerabilities Detection - Dead Paths/Computations - Next generation software development tools 				
3	Qualifikationsziele / Lernergebnisse Students can effectively use the basic static analyses related terminology. Students are familiar with modern static analyses working on intermediate representations. They are able to apply and adapt available static analysis algorithms to new scenarios.				
4	Voraussetzung für die Teilnahme Empfohlen: The lecture is targeted towards Master students with a very high degree of interest in reading, analyzing and also writing code. Basic knowledge in compiler construction is helpful. Deep knowledge of object-oriented programming concepts and in particular of object-oriented				

	programming in Java is required. Interest in learning new programming languages (in particular Scala) is required.
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0949-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%) Studierende, die die Lehrveranstaltung 20-00-0732 oder 20-00-0771 besucht haben, dürfen diese Veranstaltung nicht hören, da die Inhalte sehr vergleichbar sind.
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0949-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Einführung in Enterprise-Architektur-Management					
Modul Nr. 20-00-0967	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0967-v1	Einführung in Enterprise-Architektur-Management	0	Vorlesung	2
2	Lerninhalt <p>Die Vorlesung gibt eine Einführung in die Kernkonzepte und Vorgehensmodelle der Unternehmensarchitektur. Anhand eines in der Praxis weitverbreiteten Frameworks (TOGAF) werden diese vorgestellt. Es werden theoretische Grundlagen zu Architekturmodellen und Metamodellen (ISO/IEC/IEEE 42010) präsentiert. Der Bedarf und Zweck von Unternehmensarchitekturen in komplexen und großen Organisationen wird mit praxisnahen Beispielen illustriert. Zudem werden die wesentlichen Managementprozesse der Unternehmensarchitektur vorgestellt und deren Zusammenspiel mit anderen IT-Managementprozessen, wie Demand Management und Projektportfolio Management, diskutiert. Abschließend wird eine Einführung in die Standardisierung der Lösungsentwicklung für Unternehmensarchitekturen gegeben und in diesem Kontext der Einsatz von Referenzarchitekturen vorgestellt.</p> <p>Das Ziel der Vorlesung ist es, einerseits eine theoretische Einführung in die Grundlagen des Enterprise Architektur Managements zu geben und andererseits praktische Herausforderungen der Unternehmensarchitekturen mit Beispielen aus der Praxis zu illustrieren.</p>				
3	Qualifikationsziele / Lernergebnisse <p>Einführung in Unternehmensarchitekturen und deren Managementprozesse. Die Rolle und Zweck von Unternehmensarchitekturen für das Business IT Alignment nachvollziehen. Kernkonzepte der Unternehmensarchitektur und das Management von Unternehmensarchitekturen verstehen. Einblicke in Perspektiven, Aspekte und Beziehungen in Enterprise Architecture Frameworks geben (TOGAF). Einblicke in Architekturbeschreibungsmodelle und Metamodelle (ISO/IEC/IEEE 42010). Einführung in die Standardisierung von Lösungen und Referenzarchitekturen. Modellierung von Unternehmensarchitekturen mit ArchiMate.</p>				
4	Voraussetzung für die Teilnahme Empfohlen: Grundwissen in Informatik und Software Engineering				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0967-v1] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten				

	Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0967-vl] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname					
Grundlagen des Symmetrischen Kryptographischen Designs					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-1062	3 CP	90 h	60 h	1 Semester	Jedes 2. Semester
Sprache			Modulverantwortliche Person		
Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1062-vl	Grundlagen des Symmetrischen Kryptographischen Designs	0	Vorlesung	2
2	Lerninhalt				
	<p>In diesem Kurs werden die Studenten in die Grundlagen der symmetrischen Kryptographie eingeführt. Der Fokus wird auf dem Design verschiedener Typen von Verschlüsselungsverfahren, kollisionsresistenten Hashfunktionen und Nachrichtenauthentifizierungs-codes (MACs) aus grundlegenden Primitiven wie Blockchiffren und universellen Hashfunktionen liegen.</p> <p>Wir werden insbesondere die neuesten kryptographischen Verfahren wie GCM, HMAC, OCB, SHA3 und SIV untersuchen, die heute verwendet werden. Der Kurs wird unter Verwendung der Methode der beweisbaren Sicherheit mit einem Ausblick auf die kryptografische Praxis durchgeführt, wobei auch praktische Angriffe auf solche kryptografische Verfahren behandelt werden. Dieser Kurs beinhaltet jedoch nicht das Design von Blockchiffren oder deren Kryptanalyse.</p>				
3	Qualifikationsziele / Lernergebnisse				
	Die Studenten lernen die notwendigen Werkzeuge und Abstraktionen, um moderne kryptografische Designs und die Hintergründe für ihr Design zu verstehen. Außerdem werden die Studierenden mit der Methode der beweisbaren Sicherheit vertraut gemacht und erfahren, wie Kryptosysteme in der Praxis scheitern können. Dieser Kurs befähigt NICHT dazu, neue kryptografische Designs zu entwerfen.				
4	Voraussetzung für die Teilnahme				
	Empfohlen wird mindestens eine der folgenden Vorlesungen: Einführung in die Kryptographie, Kryptographie in der Praxis und Kryptoplexität.				
5	Prüfungsform				
	Bausteinbegleitende Prüfung:				
	<ul style="list-style-type: none"> [20-00-1062-vl] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten				
	Bestehen der Prüfung (100%)				
7	Benotung				
	Bausteinbegleitende Prüfung:				
	<ul style="list-style-type: none"> [20-00-1062-vl] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Automatische Softwareverifikation					
Modul Nr. 20-00-1069	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1069-iv	Automatische Softwareverifikation	0	Integrierte Veranstaltung	4
2	Lerninhalt Die Veranstaltung befasst sich mit dem Techniken zur automatischen Softwareverifikation und behandelt dabei folgende Themebereiche: <ul style="list-style-type: none"> - operationelle Semantik von sequentiellen Programmen - konfigurierbare Programmanalyse inklusive Konfiguration für Datenflussanalysen und Model Checking - counter-example guided abstraction refinement (CEGAR) - Bounded Model Checking - k-Induktion - kooperative Verifikation, insbesondere Conditional Model Checking - inkrementelle Verifikation - Nachprüfung von Verifikationsergebnissen (a la Proof-Carrying Code, Witness Validation) - Generierung von Testeingaben mittels Verifizierern 				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung können die Studierenden eine Vielzahl von Verfahren zur automatischen Verifikation benennen. Sie können die den Verfahren zugrunde liegenden Formalismen wiedergeben, die Funktionsweise der Verfahren beschreiben und die Verfahren klassifizieren. Außerdem können die Studierenden die Verfahren auf Beispielen anwenden und neue konfigurierbare Programmanalysen entwickeln.				
4	Voraussetzung für die Teilnahme Empfohlen: Informatikkenntnisse entsprechend der ersten vier Semester des Bachelorstudiengangs Informatik, insbesondere Kenntnisse aus der Vorlesung Aussagen und Prädikatenlogik oder Vergleichbares.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1069-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none">• [20-00-1069-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Typsysteme					
Modul Nr. 20-00-1076	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1076-iv	Typsysteme	0	Integrierte Veranstaltung	4
2	<p>Lerninhalt</p> <p>Typsysteme bieten einen effizienten Weg, um die korrekte Funktionsweise von Programmen zu garantieren, bevor diese überhaupt gestartet werden. Es gibt sie in den verschiedensten Ausprägungen: als Standard-Konstrukt und Teil einer Programmiersprache oder speziell für bestimmte Anwendungen entworfen.</p> <p>Wir werden uns u.A. mit den folgenden Themen beschäftigen:</p> <ul style="list-style-type: none"> - Einfach getypter lambda-Kalkül - Statische vs. dynamische Analyse von Typen - Operationale Semantik - Soundness von Typsystemen - Typ Inferenz - Curry-Howard-Korrespondenz - Polymorphism - Subtyping - Safety und Liveness Garantien durch Typsysteme - Abhängige Typen 				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende ein Spektrum von unterschiedlichen Typsystemen und ihre Einsatzgebiete. Sie verstehen die Grundlagen und Funktionsweise statische Programmanalyse und die Unterschiede verschiedener Typsysteme. Sie können verschiedenartige Typsysteme anwenden. Darüber hinaus können sie beurteilen und formal analysieren, welche Eigenschaften ein Typsystem garantieren kann. Sie kennen die Grenzen statischer Analysen und können Varianten bekannter Typsysteme für neue Anwendungen definieren.</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere grundlegende Logikkenntnisse und Fähigkeit, mit formalen Sprachen und Kalkülen umzugehen.</p>				

5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1076-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1076-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B,Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Verifikation paralleler Programme					
Modul Nr. 20-00-1079	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1079-iv	Verifikation paralleler Programme	0	Integrierte Veranstaltung	4
2	Lerninhalt Die Veranstaltung befasst sich mit überwiegend automatischen Techniken zur Verifikation von parallelen Programmen, insbesondere multi-threaded Programmen mit gemeinsamen Speicher. Die Veranstaltung behandelt dabei folgende Themenbereiche: - Semantik von parallelen Programmen (z.B. Interleaving-Semantik, Semantik von ausgewählten schwachen Speichermodellen) - Statische und dynamische Techniken zur Erkennung von Data Races - Techniken der Deadlockanalyse - Analyse von Programmeigenschaften (z.B. mittels Sequentialisierung, Bounded Model Checking, etc.) - Partial Order Reduction - Thread-modulare Verifikation - Verifikation unter schwachen Speichermodellen				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung können die Studierenden Verfahren zur Verifikation von parallelen Programmen, insbesondere Verfahren zur Analyse von Data Races, Deadlocks und Sicherheitseigenschaften (Safety) benennen. Sie können die den Verfahren zugrunde liegenden Formalismen wiedergeben, die Funktionsweise der Verfahren beschreiben und die Verfahren auf Beispielen anwenden. Außerdem können die Studierenden die Stärken und Schwächen der Verfahren beurteilen.				
4	Voraussetzung für die Teilnahme Empfohlen: Informatikkenntnisse entsprechend der ersten vier Semester des Bachelorstudiengangs Informatik. Vorteilhaft, aber nicht erforderlich ist der Besuch der Veranstaltung Automatische Software Verifikation.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1079-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none">• [20-00-1079-iv] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Model Checking					
Modul Nr. 20-00-1115	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1115-vl	Model Checking	0	Vorlesung	2
2	Lerninhalt * Temporallogiken: - Lineare temporal Logik (LTL), Computation Tree Logic (CTL) und CTL*: Syntax, Semantik, Komplexität * Modelprüfungsverfahren für LTL, CTL, CTL*, insbesondere Büchautomaten * Partial Order Reduction * Timed Automata				
3	Qualifikationsziele / Lernergebnisse Im Rahmen des Kurses sollen die Studierenden folgende Fähigkeiten erwerben: * Verständnis der theoretischen Grundlagen der Temporallogiken LTL, CTL und CTL* * Fähigkeit zur Auswahl der geeigneten Logik zur Spezifikation und Modellprüfung in Abhängigkeit von dem zu modellierenden System und der zu prüfenden Eigenschaft * Verfahren und Techniken zur Modellprüfung (Model Checking) z.B. Modellprüfung mit Büchautomaten, Partial Order Reduction u.ä. * Wissen über die Charakteristika und Grenzen der Modellprüfung * Kenntnisse in der Modellprüfung von Timed Automate * Fähigkeit zur Anwendung von Tools zur Modellprüfung				
4	Voraussetzung für die Teilnahme Empfohlen werden Kenntnisse in * Aussagenlogik * Deduktionssystemen * Automatentheorie				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1115-vl] (Fachprüfung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%).				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1115-vl] (Fachprüfung, mündliche / schriftliche Prüfung, Gewichtung: 100%, Standard)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulhandbuch
M. Sc. Distributed Software Systems

Studienbegleitende Leistungen

**Praktika, Projektpraktika und ähnliche
Veranstaltungen**

Modulbeschreibung

Modulname Internet - Praktikum Telekooperation					
Modul Nr. 20-00-0131	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0131 - pr	Internet - Praktikum Telekooperation	6	Praktikum	4
2	Lerninhalt Das Praktikum selbst ist in drei Teile unterteilt. In jedem Teil wird es eine Vorlesung geben, um das Thema einzuführen und neue Arbeitswerkzeuge vorzustellen. Wichtige Themen sind: <ul style="list-style-type: none"> • Einführung in Java Netzwerk Programmierung und HTTP • Peer-to-peer technologies • Web caching • Internet Standards 				
3	Qualifikationsziele / Lernergebnisse Studierende haben nach Besuch dieser Veranstaltung Wissen über zur Zeit aktuell aufkommende Technologien erworben. Ebenso haben Studierende diese Technologien (Bausteine der zukünftigen Generation von Internetdiensten) praktisch eingesetzt und Erfahrungen bei der Nutzung, Entwicklung und Integration dieser Technologien gesammelt.				
4	Voraussetzung für die Teilnahme Empfohlen: „Computer-Netzwerke und verteilte Systeme“				
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				
7	Benotung Standard				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Internet- und Web-basierte Systeme				

	<p>M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur Handbook of Research: Ubiquitous Computing Technology for Real Time Enterprises edited by Prof. Dr. Max Mühlhäuser, Dr. Iryna Gurevych, 2008, Information Science Reference, ISBN-10: 1599048329</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname					
Implementierung von Programmiersprachen					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0306	6 CP	180 h	120 h	1 Semester	Jedes 2. Semester
Sprache			Modulverantwortliche Person		
Deutsch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0306-pr	Implementierung von Programmiersprachen	0	Praktikum	4
2	Lerninhalt				
	Es werden Konzepte der Implementierung von Programmiersprachen vermittelt. Ferner werden diese Konzepte angewendet, um Erweiterungen für Programmiersprachen zu implementieren.				
3	Qualifikationsziele / Lernergebnisse				
	Die Fähigkeit, eine professionelle Aufgabe aus der Informatik selbstständig und erfolgreich nach den anerkannten Grundsätzen der Profession zu bearbeiten.				
4	Voraussetzung für die Teilnahme				
	Es wird kein Vorwissen vorausgesetzt. Jedoch sind gute Programmiererfahrungen sowie Kenntnisse über Compilerbau und virtuelle Maschinen von Vorteil.				
5	Prüfungsform				
	Bausteinbegleitende Prüfung:				
	<ul style="list-style-type: none"> [20-00-0306-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten				
	Bestehen der Prüfung (100%)				
7	Benotung				
	Bausteinbegleitende Prüfung:				
	<ul style="list-style-type: none"> [20-00-0306-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls				
	B.Sc. Informatik M.Sc. Informatik M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik				

	Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Visualisierung und Animation von Algorithmen und Datenstrukturen					
Modul Nr. 20-00-0344	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0344-pr	Visualisierung und Animation von Algorithmen und Datenstrukturen	0	Praktikum	4
2	Lerninhalt Im Rahmen des Praktikums beschäftigen wir uns mit der Frage, wie die Dynamik von Algorithmen und Datenstrukturen sinnvoll dargestellt werden kann. Dazu wird die Erstellung solcher Animation praktisch an einem System erprobt.				
3	Qualifikationsziele / Lernergebnisse Nach dem Besuch der Veranstaltung sind Studierende in der Lage, <ul style="list-style-type: none"> - die zur Verfügung gestellte API zur Animation von Algorithmen anzuwenden. - einen gegebenen Algorithmus auf seine zentralen Elemente zu untersuchen. - jeweils eine Visualisierung für die zentralen Elemente von zwei ausgewählten Algorithmen zu konstruieren. - die erstellten Visualisierungen durch die geeignete Wahl von Parametern zu generalisieren. - kritisch zu beurteilen, ob die gewählte Visualisierung den Lernprozess beim Betrachter unterstützt. 				
4	Voraussetzung für die Teilnahme Empfohlen: Die Teilnehmer benötigen neben guten Java-Kenntnissen Verständnis für die Algorithmen und Datenstruktur, die meist in GdI 2 gelehrt werden.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0344-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0344-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls				

	B.Sc. Informatik M.Sc. Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Forschungsprojekt Telekooperation					
Modul Nr. 20-00-0485	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0485-pr	Forschungsprojekt Telekooperation	9	Projektpraktikum	6
2	Lerninhalt Forschungsrelevante Projektarbeit. An einem individuellen Projekt soll das eigenständige Forschen unter Anleitung erlernt werden. Dabei werden die Themen jeweils in Zusammenarbeit mit dem Betreuer definiert. Mögliche Themenfelder: * Multimodale Interaction * Multitouch * Assistenzsysteme * Sensor Fusion				
3	Qualifikationsziele / Lernergebnisse Studierende kennen nach erfolgreichem Besuch der Veranstaltung die grundlegenden Methoden der Forschungsarbeit von der Idee bis zur fertigen Publikation. Sie verstehen wie sie komplexe Forschungsfragen in Teilprobleme zerlegen und umfassend beantworten können. Sie können die Qualität der Ergebnisse durch umfassende Evaluation bewerten und angemessen darüber berichten.				
4	Voraussetzung für die Teilnahme				
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				
7	Benotung Standard				

8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur variierend</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Praktikum Sichere Mobile Netze					
Modul Nr. 20-00-0552	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0552-pr	Praktikum Sichere Mobile Netze	6	Praktikum	4
2	<p>Lerninhalt</p> <p>Das Praktikum Sichere Mobile Netze behandelt die angewandte Softwareentwicklung und Hardware-Software Entwicklung in den Themenbereichen Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation bzw. der Kombination dieser Bereiche. Ziel ist das Lösen einer Problemstellung im Team aus den genannten Bereichen durch Implementierung in Software bzw. Hardware/Software.</p> <p>Lerninhalte:</p> <ul style="list-style-type: none"> - Lösen einer Fragestellung im Bereich Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation - Recherche von Lösungsalternativen und Abwägung von Vor-/Nachteilen der Alternativen - Konzipieren einer Softwarearchitektur bzw. kombinierten Hardware-Software Architektur - Entwerfen eines auf die Zielplattform angepassten Hardware-/Softwaredesigns - Prototypische Umsetzung auf der ausgewählten Zielplattform - Evaluation des Gesamtsystems in Bezug auf verschiedene Gütemaße - Dokumentation der erstellten Lösung 				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an der Veranstaltung besitzen die Studierenden die Fähigkeit Problemstellungen im Bereich Sichere Mobile Netze softwaretechnisch zu lösen. Die Studierenden haben hierzu Kenntnisse im Entwurf/der Umsetzung komplexer Protokolle bzw. Anwendungen in einem/mehreren der Bereiche Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation erlangt. Die Studierenden sind in der Lage die gewählten Protokolle und Anwendungen zu implementieren, zu testen und deren Funktionsfähigkeit und Leistungsfähigkeit zu evaluieren. Sie sind in der Lage die erstellten Softwareartefakte verständlich zu dokumentieren und die erzielten Projektfortschritten und -ergebnissen verständlich zu präsentieren.</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Erfolgreiche Teilnahme an einer Integrierten Veranstaltung des Fachgebiets SEEMOO</p>				

5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Themenspezifisch ausgewählte, aktuelle wissenschaftliche Veröffentlichungen
10	Kommentar

Modulbeschreibung

Modulname Projektpraktikum Sichere Mobile Netze					
Modul Nr. 20-00-0553	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0553-PP	Projektpraktikum Sichere Mobile Netze	9	Projektpraktikum	6
2	<p>Lerninhalt</p> <p>Das Projektpraktikum Sichere Mobile Netze behandelt die angewandte Softwareentwicklung und Hardware-Software Entwicklung in den Themenbereichen Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation bzw. der Kombination dieser Bereiche. Ziel ist das eigenständige Bearbeiten eines Entwicklungsprojektes im Team.</p> <p>Lerninhalte:</p> <ul style="list-style-type: none"> - Eigenständiges Bearbeiten eines Entwicklungsprojektes im Bereich Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation - Projektplanung und Projektmanagement - Recherche von Lösungsalternativen und Abwägung von Vor-/Nachteilen der Alternativen - Konzipieren einer Softwarearchitektur bzw. kombinierten Hardware-Software Architektur - Entwerfen eines auf die Zielplattform angepassten Hardware-/Softwaredesigns - Prototypische Umsetzung auf der ausgewählten Zielplattform - Evaluation des Gesamtsystems in Bezug auf verschiedene Gütemaße - Dokumentation der erstellten Lösung sowie ausführliche Dokumentation des Projektmanagements 				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an der Veranstaltung besitzen die Studierenden die Fähigkeit komplexe Problemstellungen im Bereich Sichere Mobile Netze softwaretechnisch zu lösen. Die Studierenden können hierzu eigenständig ein Projekt definieren, verwalten und durchführen. Die Studierenden haben Kenntnisse im Entwurf/der Umsetzung komplexer Protokolle bzw. Anwendungen in einem/mehreren der Bereiche Kommunikationsnetze, Sicherheit, Mobile Netze und Drahtloser Kommunikation erlangt. Die Studierenden sind in der Lage die gewählten Protokolle und Anwendungen zu implementieren, zu testen und deren Funktionsfähigkeit und Leistungsfähigkeit zu evaluieren. Sie sind in der Lage die Projektplanung und -verwaltung sowie die erstellten Softwareartefakte verständlich zu dokumentieren und die erzielten Projektfortschritten und -ergebnissen verständlich zu präsentieren.</p>				

4	Voraussetzung für die Teilnahme Empfohlen: Erfolgreiche Teilnahme an einer Integrierten Veranstaltung des Fachgebiets SEEMOO
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Themenspezifisch ausgewählte, aktuelle wissenschaftliche Veröffentlichungen
10	Kommentar

Modulbeschreibung

Modulname Praktikum: Zuverlässige Softwaresicherheit für mobile Endgeräte					
Modul Nr. 20-00-0640	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus i.d.R. jedes Sommersemester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0640-pr	Praktikum: Zuverlässige Softwaresicherheit für mobile Endgeräte	6	Praktikum	4
2	Lerninhalt <ul style="list-style-type: none"> • Einführung in Android und in die Programmierung von Apps • mögliche Bedrohungen der Privatheit durch die Ausführung von Apps • Entdecken möglicher Informationslecks durch Informationsflussanalysen • statische und dynamische Sicherheitsanalysen • Proof-Carrying-Code • eigenständige Entwicklung von Apps und Sicherheitsanalyse dieser Apps • eigenständige Erweiterung einer bestehenden Infrastruktur zur formal fundierten Sicherheitsanalyse von Android Apps 				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende grundlegende Konzepte von Android wie das Berechtigungssystem. Sie verstehen Sicherheitsprobleme, die durch die Ausführung von Apps entstehen können und verstehen wie diese durch Informationsflussanalysen verhindert werden. Sie verstehen die Vorteile der Verwendung von Proof-Carrying Code. Sie können Apps eigenständig entwickeln und die durch ihre Ausführung entstehenden Informationsflüsse bezüglich Privatheitsanforderungen evaluieren. Sie können Erweiterungen für eine existierende Sicherheitsinfrastruktur entwickeln und funktionsfähig integrieren.				
4	Voraussetzung für die Teilnahme Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere Programmierkenntnisse in Java und die Fähigkeit, mit formalen Sprachen und Kalkülen umzugehen.				
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				

7	Benotung Standard
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur ausgewählte Konferenz- und Zeitschriftenartikel
10	Kommentar

Modulbeschreibung

Modulname Software Development Tools					
Modul Nr. 20-00-0673	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0673-pr	Software Development Tools	0	Praktikum	4
2	Lerninhalt Die Entwicklung von Werkzeugen zur Unterstützung der Entwicklung von Software.				
3	Qualifikationsziele / Lernergebnisse Gewinnen von praktischer Erfahrung in der Entwicklung von Softwareentwicklungswerkzeugen. Verstehen der Grenzen von Softwareentwicklungswerkzeugen.				
4	Voraussetzung für die Teilnahme Empfohlen: Software Engineering				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0673-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0673-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.				
9	Literatur				
10	Kommentar				

Modulbeschreibung

Modulname					
Praktikum: Formale Spezifikation und Verifikation in Isabelle/HOL					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0778	6 CP	180 h	120 h	1 Semester	i.d.R. jedes Wintersemester
Sprache			Modulverantwortliche Person		
Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0778-pr	Praktikum: Formale Spezifikation und Verifikation in Isabelle/HOL	6	Praktikum	4
2	Lerninhalt				
	<ul style="list-style-type: none"> • Logik höherer Stufe (HOL) • Einführung in das Werkzeug Isabelle/HOL • Definition von Typen, Funktionen, Mengen und anderen grundlegenden Konzepten in der Spezifikationsprache von Isabelle/HOL • Führen von Beweisen für einfache Aussagen in Isabelle/HOL • Modellierung von Systemen und Eigenschaften sowie Beweis von Aussagen von schrittweise wachsender konzeptioneller Komplexität • Diskussion und Bewertung von formalen Modellen und Beweisen 				
3	Qualifikationsziele / Lernergebnisse				
	Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende die Formalismen auf denen Isabelle/HOL basiert, und sie können dieses moderne Verifikationswerkzeug verwenden. Sie können in Isabelle/HOL sowohl eigenständig als auch im Team formale Modelle von Systemen und Eigenschaften konstruieren und Aussagen beweisen. Sie können erstellte formale Modelle und Beweise beurteilen, anderen präsentieren und im Team fundiert diskutieren.				
4	Voraussetzung für die Teilnahme				
	Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere grundlegende Logikkenntnisse und Fähigkeit mit formalen Sprachen und Kalkülen umzugehen				
5	Prüfungsform				
	Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)				
6	Voraussetzung für die Vergabe von Kreditpunkten				
	Bestehen der Modulabschlussprüfung (100%)				
7	Benotung				
	Standard				

8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. Distributed Software Systems B.Sc. Computational Engineering M.Sc. Computational Engineering M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p> <ul style="list-style-type: none"> • T. Nipkow, L. C. Paulson, M. Wenzel: Isabelle/HOL: A Proof Assistant for Higher-Order Logic; Springer • online documentation material on Isabelle and Higher-Order Logic (HOL) <p>Die Literaturempfehlungen werden kontinuierlich aktualisiert.</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Dynamische Kontrolle von Systemanforderungen					
Modul Nr. 20-00-0797	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0797-pp	Dynamische Kontrolle von Systemanforderungen	0	Praktikum	6
2	Lerninhalt <ul style="list-style-type: none"> - grundlegende Konzepte der dynamischen Kontrolle in verteilten Systemen - Einführung in Werkzeuge zur Laufzeitkontrolle wie CliSeAu, JavaMOP und Polymer - Spezifikation von Systemanforderungen in unterschiedlichen Formalismen - Kombination von dynamischen Kontrollmechanismen mit Zielprogrammen - zentrale vs dezentrale Kontrolle in verteilten Systemen - Protokolle zur Koordination zwischen dezentralen Kontrollmechanismen in verteilten Systemen - eigenständige Adaption von dynamischen Kontrollmechanismen für Zielprogramme - eigenständige Erweiterung einer bestehenden Infrastruktur zur dynamischen Kontrolle von Anforderungen in verteilten Systemen und Evaluation von Erweiterungen 				
3	Qualifikationsziele / Lernergebnisse <p>Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende grundlegende Konzepte der dynamischen Kontrolle in verteilten Systemen. Sie verstehen wie Schwachstellen in verteilten Softwaresystemen, wie z.B. Sicherheitslücken, mit Hilfe von dynamischen Kontrollen beseitigt werden können. Sie verstehen, wie Anforderungen als Politiken formalisiert werden können und können solche Formalisierungen von Anforderungen in verschiedenen Sprachen durchführen. Sie können Mechanismen zur dynamischen Kontrolle für konkrete Systeme und Anforderungen einsetzen und adaptieren. Sie können Mechanismen zur dynamischen Kontrolle entwickeln, evaluieren und mit anderen Mechanismen integrieren.</p>				
4	Voraussetzung für die Teilnahme <p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere Programmierkenntnisse in Java und die Fähigkeit, mit formalen Sprachen umzugehen</p>				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0797-pp] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten				

	<p>Bestehen der Prüfung (100%)</p> <p>Module Exclusions: 20-00-0719 - Dynamic Enforcement of Software Security</p>
7	<p>Benotung</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0797-pp] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Zuverlässige Softwaresicherheit für mobile Endgeräte					
Modul Nr. 20-00-0799	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0799-pr	Zuverlässige Softwaresicherheit für mobile Endgeräte	0	Praktikum	4
2	Lerninhalt <ul style="list-style-type: none"> - Einführung in Android und in die Programmierung von Apps - mögliche Bedrohungen der Privatheit durch die Ausführung von Apps - Entdecken möglicher Informationslecks durch Informationsflussanalysen - statische und dynamische Sicherheitsanalysen - Proof-Carrying-Code - eigenständige Entwicklung von Apps und Sicherheitsanalyse dieser Apps - eigenständige Erweiterung einer bestehenden Infrastruktur zur formal fundierten Sicherheitsanalyse von Android Apps 				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung kennen Studierende grundlegende Konzepte von Android wie das Berechtigungssystem. Sie verstehen Sicherheitsprobleme, die durch die Ausführung von apps entstehen können und verstehen wie diese durch Informationsflussanalysen verhindert werden. Sie verstehen die Vorteile der Verwendung von Proof-Carrying-Code. Sie können apps eigenständig entwickeln und die durch ihre Ausführung entstehenden Informationsflüsse bezüglich Privatheitsanforderungen evaluieren. Sie können Erweiterungen für eine existierende Sicherheitsinfrastruktur entwickeln und funktionsfähig integrieren.				
4	Voraussetzung für die Teilnahme Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere Programmierkenntnisse in Java und die Fähigkeit mit formalen Sprachen und Kalkülen umzugehen.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0799-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				

6	<p>Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)</p> <p>Module Exclusions: 20-00-0640 - Software Security for Mobile Devices</p>
7	<p>Benotung Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0799-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik</p> <p>Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Distributed Systems Programming: Projektpraktikum					
Modul Nr. 20-00-0984	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0984-pp	Distributed Systems Programming: Projektpraktikum	0	Praktikum	6
2	Lerninhalt Das "DSP-Projektpraktikum" adressiert Forschungsthemen im Bereich von distributed systems (DS, deutsch verteilten Anwendungen) und Programmiersprachen für DS. Die angebotenen Themen hängen von der aktuellen Forschung der DSP Gruppe ab und umfassen unter anderem: <ul style="list-style-type: none"> • Software-defined networking (SDN) • Network function virtualization (NFV) and in-network processing (INP) • Traffic engineering (TE) • Network monitoring • Resource management in datacenters (RMF) • Big data analytics (Spark, YARN, OpenStack, ..) • Event-based systems • Security in SDN, INP, and big data • Geo-distributed data processing • Compiler infrastructures for DS • Language abstractions for DS • Session types / calculi for DS • Network Protocols <p>Die teilnehmenden Studierenden realisieren ein Forschungsprojekt welches zusammen mit dem Betreuer definiert wird. Das "DSP: Projektpraktikum" hat im Vergleich zum "DSP: Praktikum" einen größeren Umfang.</p>				
3	Qualifikationsziele / Lernergebnisse Nach der Teilnahme am "DSP-Projektpraktikum" können Studierende technische und wissenschaftliche Probleme im Bereich DS lösen.				

	<p>Je nach ausgewähltem Thema erlernen Studierende folgende Kompetenzen:</p> <p>Entwurf komplexer DS</p> <p>Methodische Analyse und Auswertung von:</p> <ul style="list-style-type: none"> • Modellen • Experimenten • Software • Entwurf von Programmiersprachen • Schreiben von technischen Dokumenten oder Projektberichten • Erstellen und vortragen eines Abschlussvortrages
4	<p>Voraussetzung für die Teilnahme</p> <p>Interesse am Erarbeiten von Lösungsvorschlägen für herausfordernde Probleme im Bereich DS, eigenverantwortliches arbeiten und ein großes Interesse an aktuellen Forschungsthemen.</p> <p>Da die angebotenen Themen ein großes Themengebiet abdecken, sind die Anforderungen sehr verschieden und projektabhängig. Eine detaillierte Beschreibung der Themen als auch der Anforderungen wird in der ersten Vorlesung präsentiert.</p>
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0984-pp] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Prüfung (100%)</p>
7	<p>Benotung</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0984-pp] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Distributed Systems Programming: Praktikum					
Modul Nr. 20-00-0985	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0985-pr	Distributed Systems Programming: Praktikum	0	Praktikum	4
2	Lerninhalt Das "DSP-Praktikum" adressiert Forschungsthemen im Bereich von distributed systems (DS, deutsch verteilten Anwendungen) und Programmiersprachen für DS. Die angebotenen Themen hängen von der aktuellen Forschung der DSP Gruppe ab und umfassen unter anderem: <ul style="list-style-type: none"> • Software-defined networking (SDN) • Network function virtualization (NFV) and in-network processing (INP) • Traffic engineering (TE) • Network monitoring • Resource management in datacenters (RMF) • Big data analytics (Spark, YARN, OpenStack, ..) • Event-based systems • Security in SDN, INP, and big data • Geo-distributed data processing • Compiler infrastructures for DS • Language abstractions for DS • Session types / calculi for DS • Network Protocols <p>Die teilnehmenden Studierenden realisieren ein Forschungsprojekt welches zusammen mit dem Betreuer definiert wird. Das "DSP: Projektpraktikum" hat im Vergleich zum "DSP: Praktikum" einen größeren Umfang.</p>				
3	Qualifikationsziele / Lernergebnisse Nach der Teilnahme am "DSP-Praktikum" können Studierende technische und wissenschaftliche Probleme im Bereich DS lösen. Je nach ausgewähltem Thema erlernen Studierende folgende Kompetenzen:				

	<p>Entwurf komplexer DS</p> <p>Methodische Analyse und Auswertung von:</p> <ul style="list-style-type: none"> • Modellen • Experimenten • Software • Entwurf von Programmiersprachen • Schreiben von technischen Dokumenten oder Projektberichten • Erstellen und vortragen eines Abschlussvortrages
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Interesse am Erarbeiten von Lösungsvorschlägen für herausfordernde Probleme im Bereich DS, eigenverantwortliches arbeiten und ein großes Interesse an aktuellen Forschungsthemen.</p> <p>Da die angebotenen Themen ein großes Themengebiet abdecken, sind die Anforderungen sehr verschieden und projektabhängig. Eine detaillierte Beschreibung der Themen als auch der Anforderungen wird in der ersten Vorlesung präsentiert.</p>
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0985-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Prüfung (100%)</p>
7	<p>Benotung</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0985-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.</p>
9	<p>Literatur</p>
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Parallele Programmierertechnologie					
Modul Nr. 20-00-1008	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1008-pr	Parallele Programmierertechnologie	0	Praktikum	4
2	Lerninhalt Die Praktikumsteilnehmer entwickeln Technologien zur parallelen Programmierung aus den folgenden Themenbereichen und/oder wenden diese an: <ul style="list-style-type: none"> • Erschließung möglicher Parallelität • Leistungsanalyse und -modellierung • Korrektheitsanalyse • Profiling • Skalierbare Algorithmen • Ressourcenmanagement und Scheduling • Anwendungen (z.B. Deep Learning) 				
3	Qualifikationsziele / Lernergebnisse <ul style="list-style-type: none"> • Kennenlernen und praktische Entwicklung und/oder Anwendung paralleler Programmierertechnologien • Einüben softwaretechnischer Methoden • Teamarbeit in Softwareprojekten • Präsentation von Projektergebnissen in Berichten und Vorträgen 				
4	Voraussetzung für die Teilnahme Empfohlen: <ul style="list-style-type: none"> • Kenntnisse paralleler Programmierung und Systeme 				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1008-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1008-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Projektpraktikum Friedens- und Kriseninformatik					
Modul Nr. 20-00-1027	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1027-pp	Projektpraktikum Friedens- und Kriseninformatik	0	Projekt	6
2	Lerninhalt Das Projektpraktikum beinhaltet Entwicklungsthemen aus der aktuellen Forschung des Fachgebiets „Wissenschaft und Technik für Frieden und Sicherheit“ (PEASEC). Neben einem generellen Überblick über aktuelle Themen wird ein tiefgehender Einblick in ein spezielles Entwicklungsgebiet vermittelt. Die Themen bestimmen sich aus den spezifischen Arbeitsgebieten der Mitarbeiter und vermitteln technische und einleitende wissenschaftliche Kompetenzen. Die Bearbeitung erfolgt in kleinen Gruppen. Projektmanagement und die Selbstorganisation im Team ist explizit Teil der Aufgabenstellung. Themen für das aktuelle Semester finden Sie unter www.peasec.de/lehre				
3	Qualifikationsziele / Lernergebnisse Die Fähigkeit eine praktische Aufgabe ggf. im Team erfolgreich nach Vorgabe zu bearbeiten und deren Ergebnisse angemessen zu präsentieren. Beispiele sind: <ul style="list-style-type: none"> • Anforderungserhebung und (empirische) Vorstudien • Konzeption und Implementierung innovativer Anwendungen • Evaluation und Weiterentwicklung bestehender Anwendungen 				
4	Voraussetzung für die Teilnahme				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1027-pp] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1027-pp] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur Reuter, C. (2018) Sicherheitskritische Mensch-Computer-Interaktion: Interaktive Technologien und Soziale Medien im Krisen- und Sicherheitsmanagement, 660 S., Wiesbaden: Springer Vieweg Altmann, J., Bernhardt, U., Nixdorff, K., Ruhmann, I., & Wöhrle, D. (2016). Naturwissenschaft - Rüstung - Frieden - Basiswissen für die Friedensforschung (Vol. 49), Wiesbaden: Springer Vieweg. Weitere Literatur wird in der Veranstaltung je nach gewähltem Thema genannt.
10	Kommentar

Modulbeschreibung

Modulname Projektpraktikum Algorithmik					
Modul Nr. 20-00-1029	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1029-pp	Projektpraktikum Algorithmik	0	Projekt	6
2	Lerninhalt Das Projektpraktikum behandelt die angewandte Softwareentwicklung in den Themenbereichen der Arbeitsgruppe Algorithmik. Ziel ist das eigenständige Bearbeiten eines Entwicklungsprojektes im Team. Lerninhalte: - Eigenständiges Bearbeiten eines Entwicklungsprojektes, - Projektplanung und Projektmanagement, - Recherche von Lösungsalternativen und Abwägung von Vor-/Nachteilen der Alternativen, - Konzipieren einer Softwarearchitektur, - prototypische Umsetzung auf der ausgewählten Zielplattform, - Evaluation des Gesamtsystems in Bezug auf verschiedene Gütemaße, - Dokumentation der erstellten Lösung.				
3	Qualifikationsziele / Lernergebnisse Nach erfolgreicher Teilnahme an der Veranstaltung besitzen die Studierenden die Fähigkeit, komplexe Problemstellungen im Themenbereich softwaretechnisch zu lösen. Die Studierenden können hierzu eigenständig ein Projekt definieren, verwalten und durchführen. Die Studierenden haben Kenntnisse im Entwurf/der Umsetzung von Algorithmen und Anwendungen erlangt. Sie sind in der Lage, die gewählten Algorithmen und ihre Anwendung zu implementieren, zu testen und deren Funktionsfähigkeit und Leistungsfähigkeit zu evaluieren, die Projektplanung und -verwaltung sowie die erstellten Softwareartefakte verständlich zu dokumentieren.				
4	Voraussetzung für die Teilnahme Empfohlen: FOP und AuD				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1029-pp] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung				

	Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1029-pp] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Praktikum zur Vorlesung Cryptocurrencies					
Modul Nr. 20-00-1031	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1031-pr	Praktikum zur Vorlesung Cryptocurrencies	0	Praktikum	4
2	Lerninhalt Diese Veranstaltung richtet sich an Studierende, die die Vorlesung Cryptocurrencies besucht haben und einige Aspekte dieses Themenkomplexes eingehender verstehen und untersuchen wollen. Sie bietet eine Plattform, um neuartige Anwendungen basierend auf Blockchain Technologie auf ihre Umsetzbarkeit und Sinnhaftigkeit zu überprüfen. Komplexe kryptografische Systeme und Bausteine aus der Vorlesung Cryptocurrencies sollen dabei in Teamarbeit verstanden und in einem dezentralen System implementiert werden. Dabei wird die eigenständige Konzeption eines Projektes gefordert, was im Verlauf der Veranstaltung von den Studierenden geplant und umgesetzt werden soll. Die Studierenden erhalten dabei erste Erfahrungen mit der Umsetzung eines komplexeren Entwicklungsprojektes.				
3	Qualifikationsziele / Lernergebnisse Studierende verstehen nach erfolgreichem Besuch der Veranstaltung die technischen und praktischen Implikationen von verteilten kryptographischen Systemen. Dazu gehören zum Beispiel erste Erfahrungen in den folgenden Bereichen: <ul style="list-style-type: none"> • Entwicklung von Smart Contracts und verteilten Applikationen • Kommunikation von Systemen durch dezentrale Peer-to-Peer Netze • Entwicklung von Software unter Benutzung kryptographischer Bausteine • Sicherheit und Anonymität von Nutzern von kryptographischen Währungen • Mögliche Angriffe auf Smart Contracts und Cryptocurrencies 				
4	Voraussetzung für die Teilnahme Empfohlen: Dieser Kurs richtet sich an Studenten, die die Vorlesung Cryptocurrencies mit guten Noten abgeschlossen haben. Weiterhin sollten Programmierkenntnisse und ein Interesse an den Themen der Vorlesung vorhanden sein.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1031-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung				

	Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1031-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B. Sc. Informatik M. Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Data Management - Praktikum					
Modul Nr. 20-00-1041	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 120 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1041-pr	Data Management - Praktikum	0	Praktikum	4
2	Lerninhalt Die Teilnehmer lösen in kleinen Projektgruppen ein gegebenes Problem. Bei den Problemen handelt es sich um Programmierprojekte, die sich auf Fragestellungen aus aktuellen Forschungsthemen des Data Management Lab beziehen. Mögliche Themenbereiche sind: - Skalierbare Datenbanksysteme und moderne Hardware - Cloud Datenbanken und Blockchains - Interaktive Daten- und Textexploration - Natural Language Interfaces für Datenbanken - Skalierbare Systeme für Maschinelles Lernen In dieser Veranstaltung setzen Studenten ein ausgewähltes Projekt um. Im Vergleich zum Praktikum haben die Probleme des Projektpraktikum einen erweiterten Umfang.				
3	Qualifikationsziele / Lernergebnisse Nach Beendigung der Veranstaltung haben Studierende folgende Lernziele erreicht: - Vertieftes Verständnis von aktuellen Techniken für moderne Datenmanagement-Systeme - Anwendung und Implementierung der Techniken in individuellen Projekten - Evaluierung von möglichen Designalternativen mit Hilfe von Benchmarks bzw. realen Workloads				
4	Voraussetzung für die Teilnahme Abhängig vom ausgewählten Thema.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1041-pr] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1041-pr] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Praktikum Multimedia Kommunikation II					
Modul Nr. 18-sm-2070	Kreditpunkte 6 CP	Arbeitsaufwand 180 h	Selbststudium 135 h	Moduldauer 1 Semester	Angebotsturnus Jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Prof. Dr.-Ing. Ralf Steinmetz		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-2070-pr	Praktikum Multimedia Kommunikation II	0	Praktikum	3
2	Lerninhalt Der Kurs bearbeitet aktuelle Entwicklungsthemen aus dem Bereich der Multimedia Kommunikationssysteme. Neben einem generellen Überblick wird ein tiefgehender Einblick in ein spezielles Entwicklungsgebiet vermittelt. Die Themen bestimmen sich aus den spezifischen Arbeitsgebieten der Mitarbeiter und vermitteln technische und einleitende wissenschaftliche Kompetenzen in einem oder mehreren der folgenden Gebiete: <ul style="list-style-type: none"> • Netzwerk und Verkehrsplanung und Analyse • Leistungsbewertung von Netzwerk-Anwendungen • Diskrete Event-basierte Simulation von Netzdiensten • Protokolle für mobile Ad hoc Netze / Sensor Netze • Infrastruktur Netze zur Mobilkommunikation / Mesh- Netze • Kontext-abhängige/bezogene Kommunikation und Dienste • Peer-to-Peer Systeme und Architekturen • Verteil-/ und Managementsysteme für Multimedia-/e-Learning-Inhalte • Multimedia Authoring- und Re-Authoring Werkzeuge • Web Service Technologien und Service-orientierte Architekturen • Anwendungen für Verteilte Geschäftsprozesse 				
3	Qualifikationsziele / Lernergebnisse Die Fähigkeit selbständig Probleme im Bereich des Design und der Entwicklung von Kommunikationsnetzen und -anwendungen für Multimediasysteme zu lösen und zu evaluieren soll erworben werden. Erworbenene Kompetenzen sind unter anderem: <ul style="list-style-type: none"> • Design komplexer Kommunikationsanwendungen und Protokolle • Implementierung und Testen von Software Komponenten für Verteilte Systeme • Anwendung von Objekt-Orientierten Analyse- und Design-Techniken • Erlernen von Projekt-Management Techniken für Entwicklung in kleinen Teams • Schreiben von Software-Dokumentation und Projekt-Berichten • Präsentation von Projektfortschritten und -ergebnissen 				

4	<p>Voraussetzung für die Teilnahme</p> <p>Das Interesse sich mit herausfordernden Themen der aktuellen Technologien und der Forschung auseinanderzusetzen. Außerdem erwarten wir:</p> <ul style="list-style-type: none"> • Solide Erfahrungen in der Programmierung mit Java und/oder C# (C/C++) • Solide Kenntnisse von Objekt-Orientierter Analyse und Design Techniken • Solide Kenntnisse in Computer Kommunikationsnetzen werden empfohlen • Die Vorlesungen in Kommunikationsnetze I (II, III, oder IV) sind von Vorteil
5	<p>Prüfungsform</p> <p>Modulabschlussprüfung:</p> <ul style="list-style-type: none"> • Modulprüfung (Studienleistung, fakultativ, Standard)
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p>
7	<p>Benotung</p> <p>Modulabschlussprüfung:</p> <ul style="list-style-type: none"> • Modulprüfung (Studienleistung, fakultativ, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>MSc ETiT, MSc iCE, BSc/MSc iST, Wi-ETiT, BSc/MSc CS, Wi-CS,</p>
9	<p>Literatur</p> <p>Die Literatur besteht aus einer Auswahl an Fachartikeln zu den einzelnen Themen. Als Ergänzung wird die Lektüre ausgewählter Kapitel aus folgenden Büchern empfohlen:</p> <ul style="list-style-type: none"> • Andrew Tanenbaum: "Computer Networks". Prentice Hall PTR (ISBN 0130384887) • Christian Ullenboom: "Java ist auch eine Insel: Programmieren mit der Java Standard Edition Version 5 / 6" (ISBN-13: 978-3898428385) • Joshua Bloch: "Effective Java Programming Language Guide" (ISBN-13: 978-0201310054) • Erich Gamma, Richard Helm, Ralph E. Johnson: "Design Patterns: Objects of Reusable Object Oriented Software" (ISBN 0-201-63361-2) • Kent Beck: "Extreme Programming Explained - Embrace Changes" (ISBN-13: 978-0321278654)
10	<p>Kommentar</p>

Modulbeschreibung

Modulname Projektpraktikum Multimedia Kommunikation II					
Modul Nr. 18-sm-2130	Kreditpunkte 9 CP	Arbeitsaufwand 270 h	Selbststudium 180 h	Moduldauer 1 Semester	Angebotsturnus Jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Prof. Dr.-Ing. Ralf Steinmetz		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-2130-pr	Projektpraktikum Multimedia Kommunikation II	0	Praktikum	6
2	Lerninhalt Der Kurs bearbeitet aktuelle Entwicklungsthemen aus dem Bereich der Multimedia Kommunikationssysteme. Neben einem generellen Überblick wird ein tiefgehender Einblick in ein spezielles Entwicklungsgebiet vermittelt. Die Themen bestimmen sich aus den spezifischen Arbeitsgebieten der Mitarbeiter und vermitteln technische und einleitende wissenschaftliche Kompetenzen in einem oder mehreren der folgenden Gebiete: <ul style="list-style-type: none"> • Netzwerk und Verkehrsplanung und Analyse • Leistungsbewertung von Netzwerk-Anwendungen • Diskrete Event-basierte Simulation von Netzdiensten • Protokolle für mobile Ad hoc Netze / Sensor Netze • Infrastruktur Netze zur Mobilkommunikation / Mesh-Netze • Kontext-abhängige/bezogene Kommunikation und Dienste • Peer-to-Peer Systeme und Architekturen • Verteil-/ und Managementsysteme für Multimedia-/e-Learning-Inhalte • Multimedia Authoring- und Re-Authoring Werkzeuge • Web Service Technologien und Service-orientierte Architekturen • Anwendungen für Verteilte Geschäftsprozesse • Ressourcen-basiertes Lernen 				
3	Qualifikationsziele / Lernergebnisse Die Fähigkeit selbständig technische und wissenschaftliche Probleme im Bereich des Design und der Entwicklung von Kommunikationsnetzen und -anwendungen für Multimediasysteme mit wissenschaftlichen Methoden zu lösen und zu evaluieren soll erworben werden. Erworbene Kompetenzen sind unter anderem: <ul style="list-style-type: none"> • Suchen und Lesen von Projekt relevanter Literatur • Design komplexer Kommunikationsanwendungen und Protokolle • Implementierung und Testen von Software Komponenten für Verteilte Systeme • Anwendung von Objekt-Orientierten Analyse- und Design-Techniken • Erlernen von Projekt-Management Techniken für Entwicklung in kleinen Teams • Systematische Evaluation und Analyse von wissenschaftlichen/technischen Experimenten 				

	<ul style="list-style-type: none"> • Schreiben von Software-Dokumentation und Projekt-Berichten • Präsentation von Projektfortschritten und -ergebnissen
4	<p>Voraussetzung für die Teilnahme</p> <p>Das Interesse herausfordernde Lösungen und Anwendungen in aktuellen Multimedia Kommunikationssystemen zu entwickeln und unter Verwendung wissenschaftlicher Methoden zu erforschen. Außerdem erwarten wir:</p> <ul style="list-style-type: none"> • Solide Erfahrungen in der Programmierung mit Java und/oder C# (C/C++). • Solide Kenntnisse von Objekt-Orientierten Analyse- und Design-Techniken. • Grundkenntnisse in Design Patterns, Refactorings, und Projekt Management. • Solide Kenntnisse in Computer Kommunikationsnetzen werden empfohlen. • Die Vorlesungen „Kommunikationsnetze I“ und „Kommunikationsnetze II“ werden empfohlen.
5	<p>Prüfungsform</p> <p>Modulabschlussprüfung:</p> <ul style="list-style-type: none"> • Modulprüfung (Studienleistung, fakultativ, Standard)
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p>
7	<p>Benotung</p> <p>Modulabschlussprüfung:</p> <ul style="list-style-type: none"> • Modulprüfung (Studienleistung, fakultativ, Gewichtung: 100%)
8	<p>Verwendbarkeit des Moduls</p> <p>MSc Wi-ETiT, BSc/MSc CS, MSc Wi-CS, MSc ETiT, MSc iST</p>
9	<p>Literatur</p> <p>Die Literatur besteht aus einer Auswahl an Fachartikeln zu den einzelnen Themen. Als Ergänzung wird die Lektüre ausgewählter Kapitel aus folgenden Büchern empfohlen:</p> <ul style="list-style-type: none"> • Andrew Tanenbaum: "Computer Networks". Prentice Hall PTR (ISBN 0130384887) • Raj Jain: "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling" (ISBN 0-471-50336-3) • Joshua Bloch: "Effective Java Programming Language Guide" (ISBN-13: 978-0201310054) • Erich Gamma, Richard Helm, Ralph E. Johnson: "Design Patterns: Objects of Reusable Object Oriented Software" (ISBN 0-201-63361-2) • Martin Fowler: "Refactorings - Improving the Design of Existing Code" (ISBN-13: 978-0201485677) • Kent Beck: "Extreme Programming Explained - Embrace Changes" (ISBN-13: 978-0321278654)
10	<p>Kommentar</p>

Modulhandbuch
M. Sc. Distributed Software Systems

Studienbegleitende Leistungen

Seminare

Modulbeschreibung

Modulname Seminar Telekooperation					
Modul Nr. 20-00-0130	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus unregelmäßig
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0130-se	Seminar Telekooperation	3	Seminar	2
2	Lerninhalt Das Seminar Telekooperation setzt sich mit der strukturierten Arbeit an wissenschaftlichen Veröffentlichungen auseinander.				
3	Qualifikationsziele / Lernergebnisse Nach dem Besuch des Seminars Telekooperation <ul style="list-style-type: none"> • sind Studierende mit dem Forschungsgebiet ihres Seminarthemas vertraut • können sich Studierende kritische mit wissenschaftlicher Literatur auseinandersetzen • eine solchen Auseinandersetzung und zugehöriger Schlussfolgerung in schriftlicher und mündlicher Form dokumentieren und vortragen 				
4	Voraussetzung für die Teilnahme Empfohlen: Allgemeine Informatik --Kenntnisse aus dem Grundstudium				
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				
7	Benotung Standard				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik				

	M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur W. Strunk, E. B. White. The Elements of Style, Pearson, ISBN 0-321-24861-9
10	Kommentar

Modulbeschreibung

Modulname Design und Implementierung moderner Programmiersprachen					
Modul Nr. 20-00-0182	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0182-se	Design und Implementierung moderner Programmiersprachen	0	Seminar	2
2	Lerninhalt				
3	Qualifikationsziele / Lernergebnisse Fähigkeit zur selbständigen Erarbeitung wissenschaftlicher Fragestellungen im Bereich „Design und Implementierung moderner Programmiersprachen“ Erwerb von Kenntnissen über ausgewählte aktuelle Themen Aneignung von Präsentationstechniken				
4	Voraussetzung für die Teilnahme Empfohlen: Vordiplom oder gleichwertige Qualifikation (d.h. fachlicher Kenntnisstand nach den ersten vier Semestern des Bachelor-Studiengangs Informatik). Das Seminar kann auch zur Einarbeitung z.B. für Studien-, Semester-, Bachelor-, Master- oder Diplomarbeiten dienen.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0182-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0182-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.				
9	Literatur				
10	Kommentar				

Modulbeschreibung

Modulname Software Engineering - Projektseminar					
Modul Nr. 20-00-0359	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0359-se	Software Engineering - Projekt Seminar	0	Seminar	2
2	Lerninhalt				
	<ul style="list-style-type: none"> • Angebotsmesse der Auftraggeber • Projektauswahl • Anforderungsanalyse beim externen Auftraggeber • Präsentation des Pflichtenheftes insbesondere der Projektorganisation und des iterativen Entwicklungsplans • Analyse der Werkzeuge und der Designkonzepte • Präsentation der Architektur und des Designs risikobehafteter Funktionen • Design und Implementierung der Iterationen • Präsentation der Implementierung und der Qualitätssicherung • Präsentation des abgeschlossenen Projekts der nächsten Studentengeneration 				
3	Qualifikationsziele / Lernergebnisse				
	<ul style="list-style-type: none"> • Erfahrung mit selbständiger Durchführung von Softwareprojekten mittleren Umfangs • Fähigkeit die verschiedenen Rollen innerhalb eines Softwareprojekts wahrzunehmen • Fähigkeit die Methoden und Werkzeuge zu bewerten und einzusetzen • Einschätzung der eigenen Kompetenz und Leistungsfähigkeit in realitätsnahen Situationen • Training der Soft Skills, insbesondere Teamfähigkeit • Kommunikation mit Kunden • Präsentationsfähigkeit 				

4	Voraussetzung für die Teilnahme Empfohlen: <ul style="list-style-type: none"> • Software Engineering - Requirements (parallel) • Software Engineering - Design (parallel) • Software Engineering - Softwarequalitätssicherung (parallel, empfehlenswert) • Empfehlenswert ist Praxiserfahrung • Teamtraining und Präsentationstechnik durch die HDA
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0359-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0359-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname					
Forschungsseminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0549	4 CP	120 h	75 h	1 Semester	i.d.R. jedes Semester
Sprache			Modulverantwortliche Person		
Deutsch und Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0549-se	Forschungsseminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation	4	Seminar	3
2	Lerninhalt				
	<p>Das Forschungsseminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation erarbeitet aktuelle Fragstellungen, die als hoch-relevant für die zukünftige Entwicklung der genannten Themenfelder eingeschätzt werden. Es umfasst das Studium, die kritische Analyse und Diskussion, das Zusammenfassen und die Präsentation ausgewählter erstklassiger Forschungsbeiträge. Ein Einblick in wissenschaftliche Arbeitsweise wird vermittelt. Ein Kurzreferat und ein abschließendes Referat sowie eine schriftliche Ausarbeitung werden erstellt.</p> <p>Die Themen des Forschungsseminars speisen sich aus den aktuellen Forschungsthemen der Arbeitsgruppe SEEMOO.</p> <p>Lernziele:</p> <ul style="list-style-type: none"> - Eigenständiges Einarbeiten in ein Thema auf dem Gebiet Kommunikationsnetze, Sicherheit, Mobilität und Drahtloser Kommunikation (i.d.R. englischsprachig) - Eigene darüber hinausgehende Literaturrecherchen - Interpretation und Einordnen der Ergebnisse der Literaturarbeit - Erstellen eines einführenden und eines vertiefenden Vortrags über die Thematik einschließlich Folienpräsentationen - Halten der beiden Vorträge vor einem Publikum mit heterogenem Vorwissen - Fachdiskussion nach jedem Vortrag - Feedback an die Vortragenden zu den Vorträgen (u.a. betreffend Rhetorik, Präsentationstechniken) und zur Fachdiskussion - Kennen des wissenschaftlichen Arbeitsprozesses und Publikationsprozesses 				
3	Qualifikationsziele / Lernergebnisse				
	<p>Nach erfolgreicher Teilnahme an der Veranstaltung besitzen die Studierenden die Fähigkeit selbstständig wissenschaftlich neue Themen zu erschließen. Sie haben ein tiefgreifendes Verständnis ausgewählter Basismechanismen, Methoden und Anwendungen in dem bearbeiteten Themenfeld erworben. Arbeitstechniken wie ausführliche Literaturrecherche, kritische Diskussion und Analyse wissenschaftlicher Artikel und die Presentation der</p>				

	erzielten Arbeitsergebnisse werden von den Studierenden beherrscht. Die Studierenden können ihre Arbeit vor einem kritischen Fachpublikum verteidigen.
4	Voraussetzung für die Teilnahme Empfohlen: Erfolgreiche Teilnahme an einer Integrierten Veranstaltung des Fachgebiets SEEMOO
5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Internet- und Web-basierte Systeme M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Themenspezifisch ausgewählte, aktuelle wissenschaftliche Veröffentlichungen
10	Kommentar

Modulbeschreibung

Modulname					
Seminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0582	3 CP	90 h	60 h	1 Semester	i.d.R. jedes Semester
Sprache			Modulverantwortliche Person		
Deutsch und Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0582-se	Seminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation	3	Seminar	2
2	Lerninhalt				
	<p>Das Seminar zu Netzen, Sicherheit, Mobilität und Drahtloser Kommunikation erarbeitet aktuelle Fragestellungen auf den genannten Gebieten. Unter Anleitung der Dozenten umfasst es das Studium, die kritische Analyse und Diskussion, das Zusammenfassen und die Präsentation ausgewählter Forschungsbeiträge. Ein Kurzreferat und ein abschließendes Referat sowie eine schriftliche Ausarbeitung werden erstellt.</p> <p>Die Themen des Seminars speisen sich aus den aktuellen Forschungsthemen der Arbeitsgruppe SEEMOO.</p> <p>Lernziele:</p> <ul style="list-style-type: none"> - Eigenständiges Einarbeiten in ein Thema auf dem Gebiet Kommunikationsnetze, Sicherheit, Mobilität und Drahtloser Kommunikation (i.d.R. englischsprachig) - Darüber hinausgehende Literaturrecherchen, angeleitet von Betreuer - Interpretation und Einordnen der Ergebnisse der Literaturarbeit, angeleitet von Betreuer - Erstellen eines einführenden und eines vertiefenden Vortrags über die Thematik einschließlich Folienpräsentationen, angeleitet von Betreuer - Halten der beiden Vorträge vor einem Publikum mit heterogenem Vorwissen - Fachdiskussion nach jedem Vortrag - Feedback an die Vortragenden zu den Vorträgen (u.a. betreffend Rhetorik, Präsentationstechniken) und zur Fachdiskussion 				
3	Qualifikationsziele / Lernergebnisse				
	<p>Nach erfolgreicher Teilnahme an der Veranstaltung besitzen die Studierenden die Fähigkeit unter Anleitung wissenschaftlich zu arbeiten. Sie kennen die grundlegenden Techniken der wissenschaftlichen Literaturarbeit und können diese für ein definiertes Thema anwenden. Sie haben ein mitteltiefes Verständnis ausgewählter Basismechanismen, Methoden und Anwendungen in dem bearbeiteten Themenfeld. Die Studierenden können dieses erworbene Wissen einem heterogenen Publikum verständlich präsentieren und die technischen Details des bearbeiteten Themas erläutern.</p>				
4	Voraussetzung für die Teilnahme				
	Empfohlen: Erfolgreiche Teilnahme an einer Integrierten Veranstaltung des Fachgebiets SEEMOO				

5	Prüfungsform Studienleistung schriftlich/mündlich (Präsentation, Dokumentation, technische Umsetzung oder vergleichbare Leistungen)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)
7	Benotung Standard
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. IT Sicherheit M.Sc. Distributed Software Systems M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Themenspezifisch ausgewählte, aktuelle wissenschaftliche Veröffentlichungen
10	Kommentar

Modulbeschreibung

Modulname Decision Procedures					
Modul Nr. 20-00-0774	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0774-se	Decision Procedures	0	Seminar	2
2	<p>Lerninhalt</p> <p>Automatisches Beweisen, insbesondere SMT, sind grundlegende Techniken für die statische Analyse von Software. Sie werden in einer Vielzahl von statischen Analysewerkzeugen, wie z.B. in SDV (Microsoft), VCC oder Krakatoa, eingesetzt.</p> <p>Damit sie jedoch sinnvoll und erfolgreich in der Praxis angewendet werden können, müssen sie in der Lage sein Probleme zu behandeln für deren Lösung lineare und nicht-lineare Arithmetik, Bitvektoren, Arrays, quantifizierte Formeln und weitere Theorien notwendig sind.</p> <p>In diesem Seminar werden wir uns mit aktuellen (state-of-the-art) Entscheidungsprozeduren für diese Theorien beschäftigen, sowie der Frage nachgehen wie diese kombiniert werden können und deren Realisierung in Beweisen wie Z3.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Erwerb von Wissen über Entscheidungsprozeduren</p> <p>Erwerb der Fähigkeit sich in ein komplexes Thema einzuarbeiten</p> <p>Erwerb der Fähigkeit Forschungsarbeiten (Konferenzbeiträge, Journalartikel) zu lesen und zu verstehen</p> <p>Erwerb der Fähigkeit komplexe Themen anderen Studierenden verständlich zu erklären</p> <p>Vortragskompetenz</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen:</p> <p>Grundlegendes Wissen in Logik erster Stufe und zugehöriger Kalküle</p> <p>Interesse und Neugier am Thema</p>				
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> [20-00-0774-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehend der Prüfung (100%)</p>				
7	<p>Benotung</p> <p>Bausteinbegleitende Prüfung:</p>				

	<ul style="list-style-type: none">• [20-00-0774-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Typsysteme von Programmiersprachen					
Modul Nr. 20-00-0796	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0796-se	Typsysteme von Programmiersprachen	0	Seminar	2
2	Lerninhalt Typsysteme sind formale Methoden, die sicherstellen, dass sich Programme gemäß einer Spezifikation korrekt verhalten. Ihr Anwendungsgebiet ist breit gefächert und umfasst bspw. Softwareentwicklung, Programmiersprachendesign und IT Sicherheit. Dieses Seminar beschäftigt sich mit grundlegenden und aktuellen Forschungsthemen zu Typsystemen, zum Beispiel Dependent Types, Typinferenz, Verfahren zur Typprüfung, constraint-basierte Ansätze usw.				
3	Qualifikationsziele / Lernergebnisse Neben praktischen Erfahrungen im wissenschaftlichen Arbeiten erlangen die Studierenden ein tieferes Verständnis für aktuelle Entwicklungen und Herausforderungen in der Forschung an Typsystemen.				
4	Voraussetzung für die Teilnahme Empfohlen: Grundkenntnisse in Mathematik und formalen Methoden.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0796-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0796-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik				

	M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Seitenkanalangriffe gegen Software					
Modul Nr. 20-00-0798	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0798-se	Seitenkanalangriffe gegen Software	0	Seminar	2
2	<p>Lerninhalt</p> <p>In diesem Seminar sollen Forschungsartikel bezüglich verschiedener Aspekte von Seitenkanalangriffen gegen Software sowie entsprechender Gegenmaßnahmen diskutiert werden; so beispielsweise:</p> <ul style="list-style-type: none"> - Seitenkanalangriffe gegen kryptographische Software, - Seitenkanalangriffe gegen Webanwendungen, - Seitenkanalangriffe gegen Betriebssysteme, - Seitenkanalangriffe auf mobile Endgeräte, - Seitenkanalangriffe in der Cloud. <p>Seitenkanäle sind indirekte, unbeabsichtigte Informationsflüsse, die durch die physikalische Ausführung eines Computerprogramms aufgedeckt werden. Beispiele hierfür sind Programmlaufzeit, Cache-Verhalten, Stromverbrauch, elektromagnetische Ausstrahlung usw. Da solche unbeabsichtigte Informationsflüsse mit geheimen Dateien wie z. B. privaten kryptographischen Schlüsseln korrelieren können, stellen Seitenkanäle ernste Sicherheitsschwachstellen dar. Während eines Seitenkanalangriffs ist der Hacker in der Lage, durch den Seitenkanal aufgedeckte Informationen zu sammeln, sie zu analysieren und anhand dieser Analyse die geheimen Dateien zu rekonstruieren. Da es dank neuer Sicherheitsmechanismen fortwährend schwieriger wird, herkömmliche Sicherheitsschwachstellen wie z. B. Programmfehler auszunutzen, werden Seitenkanäle für Hacker immer interessanter.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an diesem Seminar kennen die Studierenden das Konzept von Seitenkanalangriffen gegen Software sowie dazugehörige Beispiele. Sie verstehen die Ernsthaftigkeit der Problematik von Seitenkanälen sowie deren Verbreitung. Die Studierenden verbessern ihre Fähigkeit zum Lesen und Verstehen wissenschaftlicher Artikel, dem Präsentieren wissenschaftlicher Ergebnisse sowie zur Diskussion und Vergleich der Ansätze.</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik</p>				

5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0798-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0798-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Privatheit & Anonymität in einer vernetzten Welt					
Modul Nr. 20-00-0807	Kreditpunkte 4 CP	Arbeitsaufwand 120 h	Selbststudium 75 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0807-se	Privatheit & Anonymität in einer vernetzten Welt	0	Seminar	3
2	Lerninhalt Im Rahmen dieses Seminars werden Privatsphäre und Sicherheit sowie Auswirkungen entstehender Technologien wie das „Internet der Dinge“ diskutiert. Insbesondere werden neue Bedrohungen sowie verschiedene Angriffstechniken und entsprechende Gegenmaßnahmen betrachtet. Beispiele von Themen sind: wearable privacy, smart cars privacy, device fingerprinting, in-store tracking, HTTP(s) Traffic analysis, privacy leaks in Android-Geräte, data anonymization und differential privacy, transparency-enhancing technologies. Die Seminarteilnehmer bekommen ein Thema zugewiesen, sollen aktuelle Forschungsarbeiten lesen, den weiteren Teilnehmern vorstellen und in einer Seminararbeit zusammenfassen. Das primäre Ziel des Seminars ist es, die Fähigkeit der Studenten zu verbessern, ein wissenschaftliches Thema zu bearbeiten, eine Präsentation ähnlich wie bei einer wissenschaftlichen Konferenz zu halten und eine wissenschaftliche Diskussion zu ausgewählten Privacy-Forschungsthemen (mit-) zu gestalten. Die Studierenden simulieren die verschiedenen Phasen einer wissenschaftlichen Konferenz: Einreichung der Arbeiten, Begutachtung der Arbeiten, Feedback, Einreichung der finalen Version, Präsentation des Papiers und ggf. Sitzungsleitung.				
3	Qualifikationsziele / Lernergebnisse Das Seminar richtet sich an Bachelor- und Masterstudenten die sich für das Thema Privatheit in der digitalen Welt interessieren. Sie sollten die Bereitschaft mitbringen, neue veröffentlichte Forschungsarbeiten zum Thema "Privacy" zu begutachten bzw. zu diskutieren.				
4	Voraussetzung für die Teilnahme Grundlegendes Verständnis der Computer-Sicherheit und Netzwerkprotokolle könnte hilfreich sein.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0807-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0807-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT M.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname					
Seminar: Formale Spezifikation und Verifikation					
Modul Nr.	Kreditpunkte	Arbeitsaufwand	Selbststudium	Moduldauer	Angebotsturnus
20-00-0914	3 CP	90 h	60 h	1 Semester	Jedes 2. Semester
Sprache			Modulverantwortliche Person		
Englisch			Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0914-se	Seminar: Formale Spezifikation und Verifikation	0	Seminar	2
2	Lerninhalt				
	<ul style="list-style-type: none"> - Eigenständiges Einarbeiten in ein Thema aus dem Bereich Formale Spezifikation und Verifikation anhand von bereitgestellten wissenschaftlichen Arbeiten (englischsprachig) - Eigene darüber hinausgehende Literaturrecherchen, angeleitet durch Betreuer - Reflektion und Einordnen der Ergebnisse der Literaturarbeit, angeleitet von Betreuer - Erstellen eines Vortrags über die Thematik einschließlich Folienpräsentationen, angeleitet durch Betreuer - Halten des Vortrags vor einem Publikum mit heterogenem Vorwissen - Fachdiskussion basierend auf dem Vortrag - Feedback an die Vortragenden zu den Vorträgen (betreffend u.a. Rhetorik, Präsentationstechnik) und zur Fachdiskussion 				
3	Qualifikationsziele / Lernergebnisse				
	<p>Nach erfolgreicher Teilnahme an der Veranstaltung können die Studierenden sich eigenständig in ein Thema anhand von wissenschaftlichen Veröffentlichungen einarbeiten. Sie sind mit den verschiedenen Techniken der Literaturrecherche vertraut. Sie können über mehrere wissenschaftliche Arbeiten hinweg Techniken vergleichen und Forschungsergebnisse übergreifend evaluieren. Sie können die wesentlichen Aspekte der untersuchten Arbeiten erkennen und diese kompakt einem Publikum mit heterogenem Vorwissenstand vortragen, wobei sie dabei effektiv verschiedene Präsentationstechniken anwenden. Nach dem Vortrag können die Vortragenden aktiv eine Fachdiskussion zu dem von ihnen präsentierten Thema bestreiten.</p>				
4	Voraussetzung für die Teilnahme				
	<p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten 4 Semestern des Bachelorstudiengangs Informatik, insbesondere grundlegende Logikkenntnisse und Fähigkeit mit formalen Sprachen und Kalkülen umzugehen</p>				
5	Prüfungsform				
	<p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0914-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				

6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0914-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik B.Sc. Computational Engineering M.Sc. Computational Engineering M.Sc. Wirtschaftsinformatik B.Sc. Psychologie in IT Joint B.A. Informatik B.Sc. Sportwissenschaft und Informatik M.Sc. Sportwissenschaft und Informatik Kann im Rahmen fachübergreifender Angebote auch in anderen Studiengängen verwendet werden.
9	Literatur Wird jeweils passend zu den aktuellen Themen bekanntgegeben
10	Kommentar

Modulbeschreibung

Modulname Grundlagen der Computersicherheit					
Modul Nr. 20-00-0925	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0925-se	Grundlagen der Computersicherheit	0	Seminar	2
2	<p>Lerninhalt</p> <p>In diesem Seminar sollen Forschungsartikel bezüglich verschiedener Aspekte von Computersicherheit und deren Grundlagen diskutiert werden; die Forschungsartikel behandeln beispielsweise:</p> <ul style="list-style-type: none"> - Sicherheitsmodelle und Sicherheitseigenschaften, - Angriffe und Angreifermodelle, - Komposition, Abstraktion und Verfeinerung im Kontext von Computersicherheit - Verifizierbare Sicherheit, - Quantifizierte Sicherheit, - Zugriffskontrolle und Verwendungskontrolle, - Sicherheitsmodelle und Sicherheitseigenschaften - Informationsflusskontrolle, und - Sprach-basierte Sicherheit. <p>Die Grundlagen der Computersicherheit umfassen Theorien von Computersicherheit, formale Modelle für diese Theorien und Techniken zur Verifikation von Computersicherheit. Dabei erleichtern Theorien das konzeptuelle Verständnis für Computersicherheit und für Bedrohungen der Computersicherheit. Basierend auf diesem Verständnis bieten formale Modelle ein Gerüst für die Spezifikation der gewünschten Sicherheitseigenschaften, für die Definition des betrachteten Systems und für die eindeutige Definition der Annahmen an die Systemumgebung. Schließlich kann die Erfüllung der spezifizierten Sicherheitseigenschaften durch eine Implementierung des Systems mit Hilfe von Techniken zur Verifikation sicher gestellt werden.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an diesem Seminar werden die Studierenden fähig sein aktuelle Entwicklungen in den Grundlagen der Computersicherheit mit Bezug zu formalen Methoden zu diskutieren. Des Weiteren, werden die Studierenden ihre Fähigkeit im Lesen und Verstehen wissenschaftlicher Artikel, im Präsentieren wissenschaftlicher Ergebnisse und im Diskutieren und Vergleichen formaler Ansätze der Computersicherheit und derer Implementierung verbessern.</p>				

4	Voraussetzung für die Teilnahme Empfohlen: Informatikkenntnisse entsprechend der ersten vier Semester des Bachelorstudiengangs Informatik.
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0925-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0925-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Privatsphäre-schützende Technologien					
Modul Nr. 20-00-0935	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0935-se	Privatsphäre-schützende Technologien	0	Seminar	2
2	<p>Lerninhalt</p> <p>Daten sind das Öl des 21. Jahrhunderts und Benutzer hinterlassen immer mehr digitale Spuren, die von Firmen wie Facebook oder Google, sowie von Geheimdiensten zusammengetragen und ausgewertet werden.</p> <p>In diesem Seminar wollen wir Techniken zum Schutz der Privatsphäre betrachten, die es erlauben sensitive Daten unter Verschlüsselung zu verarbeiten, ohne die Daten selbst Preis zu geben.</p> <p>Es werden sowohl die theoretischen Hintergründe als auch die praktischen Aspekte solcher Lösungen betrachtet.</p> <p>Die Studierenden wählen ein Thema und erhalten dazu ein oder zwei Publikationen, die sie in einer Ausarbeitung schriftlich zusammenfassen und in einem Vortrag vorstellen.</p> <p>Mögliche Themen sind beispielsweise:</p> <ul style="list-style-type: none"> - Privatsphäre-schützende biometrische Identifikation - Privatsphäre-schützende mobile Anwendungen, z.B. für Standort-abhängige Dienste - Privatsphäre-schützendes Herunterladen von Dateien, z.B. für Medizinische- oder Patent-Datenbanken (Private Information Retrieval) - Privatsphäre-schützendes Finden gemeinsamer Kontakte oder Kunden (Private Set Intersection) - Privatsphäre-schützendes Prüfen der Kreditwürdigkeit (Private Function Evaluation) - Privatsphäre-schützendes Datenbanksystem (Semi-Private Function Evaluation) - Representation von Funktionen als Daten (Universal Circuits) - Oblivious RAM in Privatsphären-schützenden Technologien (ORAM + Secure Computation) - Werkzeuge für Privatsphäre-schützende Anwendungen 				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Die Studierenden lernen aktuelle und praktikable Techniken zum Schutz der Privatsphäre.</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Teilnahme an der Veranstaltung "Einführung in die Kryptographie" ist von Vorteil, aber nicht unbedingt notwendig.</p>				
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-0935-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				

6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0935-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Modellierung und Analyse von Aktoren-basierten Softwaresystemen					
Modul Nr. 20-00-0992	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0992-se	Modellierung und Analyse von Aktoren-basierten Softwaresystemen	0	Seminar	2
2	Lerninhalt Ansätze zur aktor-basierten Modellierung und formalen Analyse von Softwaresystemen: - Theorie - Modellierungssprachen - Analysen - Werkzeuge				
3	Qualifikationsziele / Lernergebnisse - Wissen über das aktor-basierte Programmierparadigma - Wissen über die Modellierung und formale Analyse von Softwaresystemen - Durchdringung und Aufarbeitung von wissenschaftlichen Konferenzbeiträgen und Journalartikeln zum Seminarthema - Verständliche und organisierte Präsentation wissenschaftlicher Artikel				
4	Voraussetzung für die Teilnahme Empfohlen: Interesse in formaler Grundkenntnisse in formalen Methoden und statischer Programmanalyse				
5	Prüfungsform Bausteinbegleitende Prüfung: • [20-00-0992-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard)				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: • [20-00-0992-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)				

8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Parallel Computing					
Modul Nr. 20-00-0994	Kreditpunkte 4 CP	Arbeitsaufwand 120 h	Selbststudium 75 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0994-se	Parallel Computing	0	Seminar	3
2	Lerninhalt Aktuelle Trends in der Parallelverarbeitung, z.B. <ul style="list-style-type: none"> • Neue Anwendungsfelder (z.B. Deep-Learning) • Neue parallele Programmiermodelle • Entwicklung paralleler Software für Smartphones • GPUs, Manycore-Architekturen • FPGAs • Architekturen für die Post-Moore-Ära • Parallele Dateisysteme • Neue parallele Algorithmen • Exascale-Computing • Cloud-Computing 				
3	Qualifikationsziele / Lernergebnisse <ul style="list-style-type: none"> • Kennenlernen aktueller Themen im Bereich Parallelverarbeitung • Literatur auswählen und analysieren • Verständliche Berichte formulieren • Übersichtliche Folien erstellen • Mündlich präsentieren • Anderen Teilnehmern Feedback geben 				
4	Voraussetzung für die Teilnahme Empfohlen: <ul style="list-style-type: none"> • Kenntnisse in Rechnerarchitektur, Programmierung, Softwaretechnik • Grundlagen paralleler Systeme 				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0994-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-0994-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Aktuelle Themen zu Programmsemantiken					
Modul Nr. 20-00-1009	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1009-se	Aktuelle Themen zu Programmsemantiken	0	Seminar	2
2	<p>Lerninhalt</p> <p>In diesem Seminar werden Forschungsartikel zu verschiedenen Aspekten von Programmsemantiken diskutiert. Beispielthemen beinhalten:</p> <ul style="list-style-type: none"> - sequentielle Programmsemantiken, - nebenläufige Programmsemantiken, - instrumentierte Programmsemantiken, - Testen von Programmsemantiken, und - Verifikation basierend auf Programmsemantiken. <p>Formale Programmsemantiken werden genutzt um ein klares Verständnis von Eigenschaften von Programm zu erreichen. Neben anderen Vorteilen erlauben solche Semantiken das Design und die Implementierung von Programmanalysen, die genutzt werden können um Eigenschaften von Programmen zu verifizieren. Während die höhere Komplexität von Programmiersprachen (z.B. Unterstützung von nebenläufigen und verteilten Systemen) formale Programmsemantiken noch wünschenswerter machen, führt diese Komplexität zu noch größeren Herausforderungen in der Formalisierung von Programmsemantiken.</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach erfolgreicher Teilnahme an diesem Seminar werden die Studierenden fähig sein, aktuelle Entwicklungen im Bereich von Programmsemantiken zu diskutieren. Des Weiteren werden die Studierenden ihre Fähigkeiten im Lesen und Verstehen wissenschaftlicher Artikel und im Präsentieren, Diskutieren und Vergleichen wissenschaftlicher Ergebnisse verbessern.</p>				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Informatik- und Mathematikkenntnisse entsprechend den ersten vier Semestern des Bachelorstudiums Informatik, insbesondere die Fähigkeit, mit formalen Sprachen und Kalkülen umzugehen</p>				
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-1009-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	<p>Voraussetzung für die Vergabe von Kreditpunkten</p> <p>Bestehen der Prüfung (100%)</p>				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1009-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Grundlagen statischer Analysen					
Modul Nr. 20-00-1028	Kreditpunkte 3 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1028-se	Grundlagen statischer Analysen	0	Seminar	2
2	Lerninhalt Die Grundlagen statischer Analysen, die zur Implementierung von fortgeschrittenen Qualitäts- und Sicherheitsanalysen gebraucht werden. Exemplarische Auswahl der Themen: - Berechnung von Kontrol- und Datenabhängigkeiten in der Gegenwart von unendlichen Schleifen und nicht reduzierbarer Kontrollflussgraphen. - Slicing von Code - Identifikation von Schleifen in Machinencode - Konstruktion von Aufrufgraphen - Statische Analyse Frameworks (z.B., IDE, IFDS, Reactive Async) - "Self-Adaptation" und statische Analysen - Sound(iness) - Specification Mining				
3	Qualifikationsziele / Lernergebnisse Die Studierenden werden vertraut sein mit den Grundlagen von fortgeschrittenen Analysen und werden in der Lage sein, die Angemessenheit bestimmter Techniken und Algorithmen für konkrete Anwendungsfälle zu beurteilen. Die Studierenden werden weiterhin in der Lage sein fortgeschrittene, technische Themen im Bereich statische Analyse effektiv zu präsentieren.				
4	Voraussetzung für die Teilnahme Das Seminar richtet sich an fortgeschrittene Bachelor- und Masterstudierende. Vertrautheit mit den Grundlagen des Compilerbaus (z.B. SSA Form) ist sehr empfehlenswert.				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-1028-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				

7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1028-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Erweitertes Seminar - Systems and Machine Learning					
Modul Nr. 20-00-1057	Kreditpunkte 4 CP	Arbeitsaufwand 120 h	Selbststudium 75 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1057-se	Erweitertes Seminar - Systems and Machine Learning	0	Seminar	3
2	<p>Lerninhalt</p> <p>Dieses Seminar dient der Diskussion neuer Forschungspapiere im Zusammenhang von Hardware-/Softwaresystemen und maschinellem Lernen (ML). Das Seminar zielt auf die Verbindungen zwischen diesen Themenbereichen ab und diskutiert Fragestellungen, die auf praktisch anwendbares maschinelles Lernen zugeschnitten sind wie z.B. Hardware-Beschleuniger für ML, verteilte skalierbare ML-Systeme, neuer Programmierparadigmen für ML, Automatisiertes ML, sowie Anwendungen von ML für Systeme.</p> <p>Jeder Teilnehmer präsentiert ein Forschungspapier, das anschließend von allen Teilnehmern diskutiert wird. Darüber hinaus werden zusammenfassende Arbeiten in Gruppen verfasst und einem Peer-Review Prozess unterzogen. Die vorzustellenden Arbeiten stellen in der Regel aktuelle Publikationen in relevanten Konferenzen und Zeitschriften dar.</p> <p>Das Seminar wird als Blockveranstaltung angeboten. Weitere Informationen unter: http://binnig.name</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Nach diesem Seminar sollten Studierende in der Lage sein:</p> <ul style="list-style-type: none"> - einen unbekanntem Text aus den Bereichen des Seminars selbständig aufzuarbeiten - eine Präsentation und eine schriftliche Zusammenfassung für ein Fachpublikum in diesem Gebiet zu entwickeln - an einer Fachdiskussion über ein Thema aus den Bereichen des Seminars sinnvoll teilzunehmen - die Meinung über eine wissenschaftliche Arbeit in der Form eines schriftlichen Peer-Reviews zu artikulieren 				
4	<p>Voraussetzung für die Teilnahme</p> <p>Empfohlen: Grundkenntnisse in maschinellem Lernen, skalierbares Datenmanagement und Hardware-/Softwaresystemen.</p>				
5	<p>Prüfungsform</p> <p>Bausteinbegleitende Prüfung:</p> <ul style="list-style-type: none"> • [20-00-1057-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				

6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1057-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Software-Engineering für Künstliche Intelligenz					
Modul Nr. 20-00-1097	Kreditpunkte 4 CP	Arbeitsaufwand 90 h	Selbststudium 60 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Englisch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-1097-se	Software-Engineering für Künstliche Intelligenz	0	Seminar	3
2	<p>Lerninhalt</p> <p>Künstliche Intelligenz (KI) ist mittlerweile Bestandteil vieler datengetriebenen Anwendungen; zum Beispiel in der Finanzindustrie, Medizin, Kognitionswissenschaft oder Biologie. Derartige Ansätze des maschinellen Lernens (ML) erfordern eine genaue Domänen- und Anforderungsanalyse, angemessenes Softwaredesign und -Entwicklung, besonderes Testen und Debugging sowie spezielle Techniken, um Skalierbarkeit und Wartbarkeit sicherzustellen. Während KI-Systeme zunehmend größeren Einfluss in vielen Bereichen besitzen, verwenden Entwickler und Data-Scientists weiterhin Methoden (Scripting, informelle/nicht-verschriftlichte Spezifikationen, trial-and-error Testing), die nicht dem aktuellen Stand der Technik in den Ingenieursdisziplinen entsprechen. Vor diesem Hintergrund ist es von entscheidender Bedeutung die Jahrzehnte lange Entwicklung im Software-Engineering (SE) zur Systematisierung von Entwicklungsprozessen für diesen Bereich zu nutzen.</p> <p>In diesem Kurs wird Studierenden ein Thema im Bereich SE für KI zugewiesen. Ausgehend von vorgegebenen Quellen und persönlicher erweiternder Literaturrecherche bereiten Studierende eine Präsentation mit anschließender Diskussion vor. Diese werden an regelmäßigen Terminen gehalten. Alle Studierenden, die an einem Termin nicht präsentieren, bereiten sich auf die jeweilige Diskussion mit einführendem Lesematerial vor. Die Benotung basiert auf der Vorbereitung und der Präsentation der zugewiesenen Themenschwerpunkte sowie auf der Teilnahme an allen Diskussionen.</p> <p>Beachten Sie bitte die Kursseite für mehr Informationen und Ankündigungen: https://allprojects.github.io/SE4AI/</p>				
3	<p>Qualifikationsziele / Lernergebnisse</p> <p>Die Studenten entwickeln ein tieferes Verständnis zu SE für KI. Dies umfasst die Schwerpunkte Requirements Engineering, Qualitätssicherung, Entwicklungsprozesse sowie Softwarearchitektur und -Design für Modularität, Wiederverwendbarkeit, Effizienz, Skalierbarkeit, Fairness und Privatsphäre.</p> <p>Die Studierenden lernen die Vorbereitung und Präsentation von wissenschaftlichen Inhalten für ein Publikum mit unterschiedlichem Hintergrundwissen. Außerdem üben die Studierenden die</p>				

	effiziente Vorbereitung von und aktive Teilnahme an wissenschaftlichen Diskussionen sowie deren Moderation.
4	Voraussetzung für die Teilnahme Empfohlen: Basiswissen zu Software-Engineering. Interesse an Künstlicher Intelligenz.
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1097-se] (Studienleistung, mündliche / schriftliche Prüfung, Standard)
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%).
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> • [20-00-1097-se] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%)
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.
9	Literatur
10	Kommentar

Modulbeschreibung

Modulname Seminar Multimedia Kommunikation II					
Modul Nr. 18-sm-2090	Kreditpunkte 4 CP	Arbeitsaufwand 120 h	Selbststudium 90 h	Moduldauer 1 Semester	Angebotsturnus Jedes Semester
Sprache Deutsch und Englisch			Modulverantwortliche Person Prof. Dr.-Ing. Ralf Steinmetz		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	18-sm-2090-se	Seminar Multimedia Kommunikation II	0	Seminar	2
2	Lerninhalt Das Seminar befasst sich mit aktuellen und aufkommenden Trends, die als relevant für die zukünftige Entwicklung von Multimedia Kommunikationssystemen eingeschätzt werden. Lernziel ist es, Kenntnisse über zukünftige Forschungstrends im verschiedenen Bereichen zu erarbeiten. Hierzu erfolgt eine ausführliche Literaturlerarbeit, die Zusammenfassung sowie die Präsentation von ausgewählten, hochwertigen Forschungsarbeiten aus aktuellen Top-Zeitschriften, -Magazinen und -Konferenzen im Themenfeld Multimedia Kommunikation. Mögliche Themen sind: <ul style="list-style-type: none"> • Knowledge & Educational Technologies • Self organizing Systems & Overlay Communication • Mobile Systems & Sensor Networking • Service-oriented Computing • Multimedia Technologies & Serious Games 				
3	Qualifikationsziele / Lernergebnisse Die Studierenden erarbeiten sich an Hand von aktuellen wissenschaftlichen Artikeln, Standards und Fachbüchern tiefe Kenntnisse über Multimedia Kommunikationssysteme und Anwendungen, welche die Zukunft des Internet bestimmen. Dabei werden Kompetenzen in folgenden Gebieten erworben: <ul style="list-style-type: none"> • Suchen und Bewerten von relevanter wissenschaftlicher Literatur • Analysieren und Einschätzen von komplexen technischen und wissenschaftlichen Informationen • Schreiben von technischen und wissenschaftlichen Zusammenfassungen • Präsentation von technischer und wissenschaftlicher Information 				
4	Voraussetzung für die Teilnahme Solide Kenntnisse in Computer Kommunikationsnetzen. Die Vorlesungen Kommunikationsnetze I und II werden empfohlen.				
5	Prüfungsform Modulabschlussprüfung: <ul style="list-style-type: none"> • Modulprüfung (Studienleistung, fakultativ, Standard) 				

6	Voraussetzung für die Vergabe von Kreditpunkten
7	Benotung Modulabschlussprüfung: <ul style="list-style-type: none">• Modulprüfung (Studienleistung, fakultativ, Gewichtung: 100%)
8	Verwendbarkeit des Moduls CS, Wi-CS, ETiT, Wi-ETiT, MSc CS, MSc ETiT, MSc iST
9	Literatur Entsprechend des gewählten Themenbereichs (ausgewählte Artikel aus Journalen, Magazine und Konferenzen).
10	Kommentar

Modulhandbuch
M. Sc. Distributed Software Systems

Studienbegleitende Leistungen

Praktikum in der Lehre

Modulbeschreibung

Modulname Praktikum in der Lehre - Ubiquitous / Mobile Computing					
Modul Nr. 20-00-0996	Kreditpunkte 5 CP	Arbeitsaufwand 150 h	Selbststudium 105 h	Moduldauer 1 Semester	Angebotsturnus Jedes 2. Semester
Sprache Deutsch			Modulverantwortliche Person Koordinatoren/Koordinatorinnen Distributed Software Systems		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
	20-00-0996-pl	Praktikum in der Lehre - Ubiquitous / Mobile Computing	0	Praktikum in der Lehre	3
2	Lerninhalt Es sollen Fähigkeiten erlernt werden, geeignete Lernmaterialien selbst zu erstellen, ihren Einsatz kritisch zu begleiten und dabei auch die Lernenden zu betreuen und anzuleiten.				
3	Qualifikationsziele / Lernergebnisse Aufbereitung und Vermittlung des Vorlesungsinhaltes.				
4	Voraussetzung für die Teilnahme				
5	Prüfungsform Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0996-pl] (Studienleistung, mündliche / schriftliche Prüfung, Standard) 				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Prüfung (100%)				
7	Benotung Bausteinbegleitende Prüfung: <ul style="list-style-type: none"> [20-00-0996-pl] (Studienleistung, mündliche / schriftliche Prüfung, Gewichtung: 100%) 				
8	Verwendbarkeit des Moduls B.Sc. Informatik M.Sc. Informatik Kann in anderen Studiengängen verwendet werden.				
9	Literatur				
10	Kommentar				

Modulhandbuch

M. Sc. Distributed Software Systems

Masterarbeit

Modulbeschreibung

Modulname Masterarbeit Distributed Software Systems					
Modul Nr. 20-AM-5200	Kreditpunkte 30 CP	Arbeitsaufwand 900 h	Selbststudium 900 h	Moduldauer	Angebotsturnus Jedes Semester
Sprache Deutsch/Englisch			Modulverantwortliche Person Studiendekan/Studiendekanin		
1	Kurse des Moduls				
	Kurs Nr.	Kursname	Arbeitsaufwand (CP)	Lehrform	SWS
2	Lerninhalt Selbständige Bearbeitung einer wissenschaftlichen Fragestellung mit inhaltlichen Bezug zum Einsatz, der Entwicklung oder dem Management von verteilten Softwaresystemen. Die Bearbeitung erfolgt nach wissenschaftlichen Grundsätzen in begrenzter Zeit. Die Problemstellung, Vorgehensweise sowie die Ergebnisse werden schriftlich dokumentiert und mündlich in einem Kolloquium präsentiert.				
3	Qualifikationsziele / Lernergebnisse / Kompetenzen Die Studierenden sind nach der Masterarbeit in der Lage, <ul style="list-style-type: none"> • eine komplexere wissenschaftliche Fragestellung mit Forschungsbezug nach wissenschaftlichen Grundsätzen selbstständig zu bearbeiten, • die im Studium erworbenen Kenntnisse, Methoden und Kompetenzen zu verknüpfen und anzuwenden, • die relevante Literatur zu recherchieren, einzugrenzen und auszuwerten, • das Thema sinnvoll zu systematisieren und einen Argumentationsstrang aufzubauen, • die Validität von Pro- und Kontraargumenten nachvollziehbar abzuwägen, • die Ergebnisse in die aktuelle Forschung einzuordnen und zu bewerten, • die Ergebnisse schriftlich nach wissenschaftlichen Grundsätzen niederzulegen, • die Ergebnisse zu präsentieren und argumentativ zu vertreten. 				
4	Voraussetzung für die Teilnahme				
5	Prüfungsform schriftliche Arbeit und ein mündliches Kolloquium				
6	Voraussetzung für die Vergabe von Kreditpunkten Bestehen der Modulabschlussprüfung (100%)				
7	Benotung Die schriftliche Arbeit geht mit 85% und das Kolloquium mit 15% in die Note für die Masterarbeit ein.				
8	Verwendbarkeit des Moduls M. Sc. Distributed Software Systems				

9	<p>Literatur</p> <ul style="list-style-type: none"> - Balzert, Helmut; Schäfer, Christian; Schröder, Marion: Wissenschaftliches Arbeiten - Wissenschaft, Quellen, Artefakte, Organisation, Präsentation. W3L-Verlag; Auflage: 2, 2011 - Sandberg, Berit: Wissenschaftlich Arbeiten von Abbildung bis Zitat: Lehr- und Übungsbuch für Bachelor, Master und Promotion. De Gruyter Oldenbourg; Auflage: 2, 2013
10	<p>Kommentar</p> <p>Die Abschlussarbeit muss innerhalb von 26 Wochen angefertigt und eingereicht werden. Sie hat einen Arbeitsaufwand von 900 Stunden. Ein Studium in Regelstudienzeit setzt voraus, dass bei Beginn der Masterarbeit im 4. Semester bei voller Ausschöpfung der Bearbeitungszeit von 26 Wochen nicht später als Anfang Februar bei Studienbeginn zum Wintersemester bzw. Anfang August bei Studienbeginn zum Sommersemester begonnen werden muss.</p>