



Ostfalia
Hochschule für angewandte
Wissenschaften

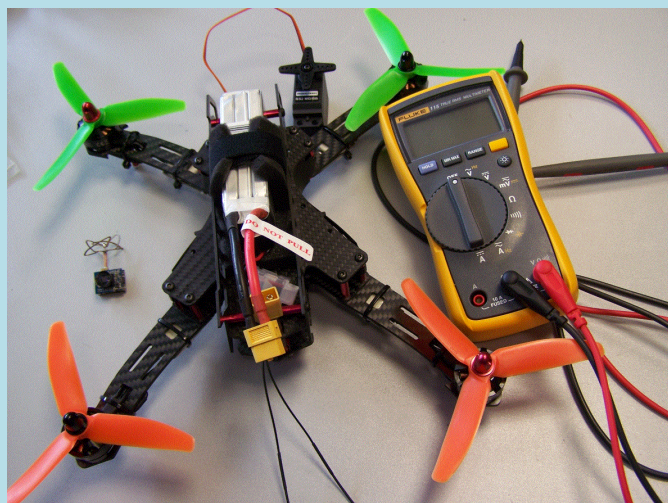
Dekanat

Katalog der Wahlpflichtfächer - Präsenz-Bachelorstudiengänge Fakultät Informatik

(PO2018)

Version 1.1 - Änderungen vorbehalten, 25.09.2020

Dies ist eine veraltete Version.
Die aktuellen Modulhandbücher finden Sie unter:
www.ostfalia.de/i/mhb



Inhaltsverzeichnis

1	Allgemeine Hinweise	3
1.1	Hinweise zu Formularfeldern und Modulprüfungen	3
1.2	Weitere Informationen im Leitfaden	3
1.3	Abkürzungsverzeichnis	3
2	Modulbeschreibungen	4
2.1	Überfachlich	4
2.1.1	Gesprächs- und Verhandlungsführung	4
2.1.2	Konfliktmanagement	6
2.1.3	Leiten von Arbeitsgruppen	7
2.1.4	Rhetorik und Argumentation	8
2.2	Fachlich	10
2.2.1	Autonomes Fahren	10
2.2.2	Autosar	12
2.2.3	Concurrent Computing	13
2.2.4	Einführung in die Elektromobilität	15
2.2.5	Embedded Linux	17
2.2.6	Fahrerassistenzsysteme	18
2.2.7	Hard- und Software intelligenter Systeme (HuSiS)	19
2.2.8	Interdisziplinäres Projekt	20
2.2.9	Java Enterprise Edition (Java EE)	21
2.2.10	Malware-Labor	22
2.2.11	Medizinische Informatik	23
2.2.12	Pervasive Systeme	25
2.2.13	Programmierparadigmen C++	26
2.2.14	RFID - Radio Frequency Identification	27
2.2.15	Software für sichere Systeme	28
2.2.16	Virtualisierung	29
2.2.17	Wissenschaftstheorie	30
3	Dokumenthistorie	31

Suchen (bzw. Finden) innerhalb eines PDFs / einer Word-Datei mit <Strg> <F>

1 Allgemeine Hinweise

1.1 Hinweise zu Formularfeldern und Modulprüfungen

ECTS = „European Credit Transfer and Accumulation System“. Das ECTS ermöglicht Studierenden die einfache Anerkennung von im In- und Ausland erbrachten Studienleistungen. Dabei werden jedem Modul eine bestimmte Anzahl an Leistungspunkten zugeordnet, die dann bei erfolgreichem Abschluss einer Veranstaltung angerechnet werden.

Die studentische Arbeitsbelastung wird als Mittelwert aufgeführt. Der erforderliche Aufwand setzt sich aus der Kontaktzeit (= Veranstaltung) und dem Eigenanteil zusammen. Pro Lehrveranstaltung müssen ca. sechs Stunden für Anwesenheit sowie Vor- und Nachbereitung gerechnet werden.

Die Lehrenden geben die angewendete Prüfungsform und die Lehrformen zu Anfang jedes Semesters in der Lehrveranstaltung bekannt. Mündliche Prüfungen dauern 15-30 Minuten. Eine besondere Prüfungsform stellen die Modulprüfungen dar. Wenn in den Lehrveranstaltungen desselben Moduls die „Modulprüfung“ angewendet wird, dann werden die Inhalte aller Lehrveranstaltungen dieses Moduls gleichzeitig in einer gemeinsamen Prüfung abgefragt.

SWS = Semesterwochenstunden; 2 SWS entsprechen 90 Minuten.

1.2 Weitere Informationen im Leitfaden

Weitere Informationen zu den Wahlpflichtfächern stehen im Dokument „**Leitfaden für die Informatik-Präsenzstudiengänge**“.

Im Zweifelsfall erkundigen Sie sich bitte bezüglich Wahlpflichtfächern beim Studiendekan.

1.3 Abkürzungsverzeichnis

EA	Experimentelle Arbeit	PA	Projektarbeit
EP	Elektronische Prüfung	PB	Praxisbericht
H	Hausarbeit	PF	Portfolioprüfung
K	Klausur	PO	Prüfungsordnung
M	Mündliche Prüfung	R	Referat

2 Modulbeschreibungen

2.1 Überfachlich

2.1.1 Gesprächs- und Verhandlungsführung

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Gesprächs- und Verhandlungsführung				
LV alte PO (2013):	Gesprächs- und Verhandlungsführung				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	regelm.	WPF	5.0	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Ragna Winter Norbert Köhler (CS)
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Hausarbeit	Vortrag, Gruppenarbeit und Verhandlungsübungen	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • erwerben Gesprächsführungskompetenz im beruflichen Kontext • kennen unterschiedliche Arten von Mitarbeitergesprächen • kennen die Gesprächsphasen und –techniken • reflektieren eigenes Verhalten und Körpersprache im Kontext professioneller Kommunikation • erwerben Kenntnisse für Vorbereitung, Planung, Strukturierung und Durchführung von Verhandlungen • kennen Grundregeln des klassischen Verhandeln • kennen und verstehen das Harvard-Konzept als Alternative zum klassischen Verhandeln • verfügen über Grundkenntnisse rhetorischer Instrumente
Lehrinhalte
<ul style="list-style-type: none"> • Planung, Durchführung und Auswertung von beruflichen Gesprächen • Was ist in den einzelnen Phasen der Gespräche besonders zu beachten? • Verschiedene Arten von Mitarbeitergesprächen • Auffrischung von Gesprächstechniken • Authentisch und professionell kommunizieren • Bedeutung der Körpersprache und der bewusste Einsatz zur Gesprächsförderung • Gesprächshaltungen nach A. Harris • Selbstwahrnehmung und Fremdwahrnehmung (Johari Fenster) • Grundbegriffe des Verhandeln • Grundregeln des klassischen Verhandeln

- Vor- und Nachteile des klassischen Verhandeln
- Das Harvard-Konzept des sachgerechten Verhandeln:
 - o Regel 1: Menschen und Probleme getrennt voneinander behandeln
 - o Rhetorische Instrumente zur Steuerung von Gesprächen und Verhandlungen
 - o Regel 2: Auf Interessen konzentrieren, nicht auf Positionen
 - o Regel 3: Entscheidungsmöglichkeiten zum beiderseitigen Vorteil entwickeln
 - o Analytische und kreative Methoden in der Verhandlungsvorbereitung
 - o Regel 4: Ergebnisse auf objektiven Standards aufbauen
 - o Rhetorische Methoden zur Herstellung einer produktiven Verhandlungssituation

Inhalt / Aufgabe des Kontaktstudiums:

Training, Feedback und Reflektion der Methoden

Inhalt / Aufgabe des Eigenstudiums:

Reflektion der Methoden und Vertiefung des Gelernten, Transfer auf andere Situationen

Literatur

- Preuß-Scheuerle: Praxishandbuch Kommunikation. 2.Aufl., Wiesbaden 2016
- Scharlau, Christine; Rossié, Michael: Gesprächstechniken, 1. Aufl., Freiburg 2012
- Fisher/Ury/Patton: Das Harvard – Konzept: Klassiker der Verhandlungstechnik. 25. Aufl., Frankfurt/Main 2015.
- Mühlen, Alexander: Internationales Verhandeln: Konfrontation, Wettbewerb, Zusammenarbeit. - 4. Aufl., Berlin 2010

Versionsnummer: Eintrag erstellt am:

2.1.2 Konfliktmanagement

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Konfliktmanagement				
LV alte PO (2013):	Konfliktmanagement				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	regelm.	WPF	3.0	75h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Jutta Schwiebert (CS)
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Hausarbeit	Input und Moderation, Einzel- Paar- und Gruppenarbeit, Übungen und Fallbeispiele, Szenisches Arbeiten, Präsentation	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • sind in der Lage, frühzeitig einen Konflikt zu erkennen • kennen wesentliche Ursachen für die Entstehung von Konflikten • kennen geeignete Interventions- und Präventionsmöglichkeiten • haben ihren eigenen Konfliktstil reflektiert • haben Phasen und Werkzeuge für ein Konfliktgespräch kennen gelernt • kennen die Bedeutung subjektiver Wahrheiten in Konflikten
Lehrinhalte
<ul style="list-style-type: none"> • Das Wesen von Konflikten • Symptome und Eskalationsdynamik • Analyse von Konflikten • Lösungsorientiertes Vorgehen in Konflikten • Das Konfliktgespräch • Methoden zur Klärung von Teamkonflikten <p>Inhalt / Aufgabe des Kontaktstudiums: Training, Feedback und Reflektion der Methoden</p> <p>Inhalt / Aufgabe des Eigenstudiums: Reflektion der Methoden und Vertiefung des Gelernten, Transfer</p>
Literatur
<ul style="list-style-type: none"> • Allhoff, D.-W./Allhoff W.: Rhetorik & Kommunikation. München 2014. • Glasl, F.: Selbsthilfe in Konflikten. Konzepte, Übungen, praktische Methoden. Stuttgart 2000. • Fehlau, E.-G.: Konflikte im Beruf: Erkennen, lösen, vorbeugen. Freiburg 2013.

Versionsnummer: Eintrag erstellt am:

2.1.3 Leiten von Arbeitsgruppen

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Leiten von Arbeitsgruppen				
LV alte PO (2013):	Leiten von Arbeitsgruppen				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	regelm.	WPF	3.0	75h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Jutta Schwiebert (CS)
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Hausarbeit	Input und Moderation, Einzel-, Paar- und Gruppenarbeit, Übungen und Fallbeispiele, Szenisches Arbeiten, Präsentation	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen die Definition von „Team“ • kennen die Kriterien „echter“ Teamarbeit und können diese von teamähnlichem Arbeiten unterscheiden • kennen den Teamentwicklungsprozess • haben ihren eigenen Teamtypen und Arbeitsstil erkannt und erlebt • kennen Instrumente zur Optimierung von Teamarbeit und Konfliktlösung • kennen Strategien zur Leitung von Teams
Lehrinhalte
<ul style="list-style-type: none"> • Team und Teamarbeit – Definition und Kriterien • Der Teamentwicklungsprozess – Phasen und Ebenen • Teamtypen erkennen und gezielt einbinden • Teamrollen • Methoden zur Problemlösung im Team • Lösungsorientierte Interventionswerkzeuge <p>Inhalt / Aufgabe des Kontaktstudiums: Training, Feedback und Reflektion der Methoden</p> <p>Inhalt / Aufgabe des Eigenstudiums: Teamaufgabe, Reflektion der Methoden und Vertiefung des Gelernten, Transfer</p>
Literatur
<ul style="list-style-type: none"> • Gellert, M. & Nowak, Claus (2010): Teamarbeit, Teamentwicklung, Teamberatung. Ein Praxisbuch für das Arbeiten in und mit Teams. • Wellhöfer, P.R. (2007): Gruppendynamik und soziales Lernen

Versionsnummer: Eintrag erstellt am:

2.1.4 Rhetorik und Argumentation

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Rhetorik und Argumentation				
LV alte PO (2013):	Rhetorik und Argumentation				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	regelm.	WPF	3.0	75h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Jutta Schwiebert (CS)
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Ausarbeitung + Präsentation	Methodentraining, Übungen mit Videofeedback	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • erwerben grundlegende Argumentationskompetenzen für ein überzeugendes Auftreten im Rahmen von Präsentationen und Vorträgen • kennen den Aufbau informativer und argumentativer Gliederungen • kennen grundlegende Redestrategien zur Erreichung einer Meinungsbildung oder eines Meinungswechsels • können schlüssige Argumentationen erarbeiten und in Präsentationsübungen und hinsichtlich ihrer Überzeugungskraft unter Beweis stellen • reflektieren und optimieren ihr eigenes Vortrags- und Sprechverhalten. • Können ihre Entscheidung für ein bestimmtes Argumentationskonzept begründen • erwerben Strategien im Umgang mit kritischen Fragen und Einwänden
Lehrinhalte
<ul style="list-style-type: none"> • Überzeugungskraft – welche Faktoren sind bedeutsam? • Zielorientierte Stoffsammlung und -auswahl • Aufbau informativer und argumentativer Gliederungen • Strategien zur Erreichung einer Meinungsbildung oder eines Meinungswechsels • Argumentationsfiguren • Umgang mit Einwänden, Fragen und Störungen <p>Inhalt / Aufgabe des Kontaktstudiums: Training, (Video-)Feedback und Reflektion der Methoden</p> <p>Inhalt / Aufgabe des Eigenstudiums: Reflektion der Methoden und Vertiefung des Gelernten, Transfer</p>

Literatur

- | |
|--|
| <ul style="list-style-type: none">• Allhoff, D.-W./Allhoff W.: Rhetorik & Kommunikation. München 2014.• Schilling, G.: Angewandte Rhetorik und Präsentationstechnik. Berlin 2012. |
|--|

Versionsnummer: Eintrag erstellt am:

2.2 Fachlich

2.2.1 Autonomes Fahren

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Autonomes Fahren				
LV alte PO (2013):	Autonomes Fahren				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	2x pro Jahr	WPF	5.0	150h, davon ca. 40h Kontaktstudium, ca. 110h Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Wissen über C-Programmierung, Mikrocontroller, Echtzeitbetriebssysteme	WPF (auch Master PO2013)	Prof. Dr.-Ing. G. Bikker
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
EA 70% + 30% K oder M	Vorlesung, Übung, Labor	

Kompetenzziele (nach Bloom)
<p>Ziel ist es, den Studierenden fachübergreifende Kenntnisse im Bereich des autonomen Fahrens zu vermitteln und sie schrittweise in die nötigen Grundlagen und Begrifflichkeiten einzuführen. Das dazu erforderliche Wissen wird in Vorlesungen vermittelt und durch integrierte Übungen sowie Labore gefestigt.</p> <p>Die Studierenden werden für ein gesamtheitliches Verständnis der Fahrzeugtechnik, Fahrdynamik und der Fahrzeugführung sensibilisiert. Durch die praktische Umsetzung wird das Wissen gefestigt und das analytische Denkvermögen der Teilnehmer gestärkt. Es sollen insbesondere die fahrdynamischen Grundlagen und die Steuerung/Regelung auf Basis der Sensorfunktion Berücksichtigung finden.</p> <p>Nach Abschluss des Moduls haben die Studierenden ein fundiertes Verständnis für die grundlegenden Begriffe und Gesetzmäßigkeiten der Fahrdynamik entwickelt. Sie erhalten ein Basiswissen über die Themenfelder des autonomen Fahrens und sind in der Lage, die erlernten Modelle anzuwenden sowie die Ergebnisse mit angemessenen Verfahren zu analysieren.</p>
Lehrinhalte
<ul style="list-style-type: none"> • Systemarchitektur (Hardware- und Software-Architektur, Vernetzung) • Sensorik und Aktorik (Video-, Ultraschall-, Lidar-basierte Sensoren, el. Antriebe) • Umgebungserfassung und Navigation • Spezielle Fahrsituationen • Sicherheit und Test

Literatur

- Hermann Winner, Stephan Hakuli, Felix Lotz, Christina Singer (Hrsg.) 2015: Handbuch Fahrerassistenzsysteme, ATZ/MTZ-Fachbuch, Springer Vieweg, Berlin/Heidelberg ISBN 978-3-658-05733-6

Weitere Literatur wird in der Veranstaltung bekanntgegeben.

Versionsnummer: Eintrag erstellt am:

2.2.2 Autosar

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Autosar				
LV alte PO (2013):	Autosar				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr	WPF	5.0	150h, davon ca. 30% Kontaktstudium, ca. 70% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	G. Kircher
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
50% Klausur / mündliche Prüfung 50% Referat / experimentelle Arbeit	Unterricht, Labor mit Projektvorträgen, Projektarbeit (4 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen die Komponenten und Schnittstellen der AUTomotive Open System ARchitecture • entwerfen, implementieren und betreiben verteilte, eingebettete Systeme mit der im Standard vorgeschlagenen Methodik • kennen die Methoden der verteilten Entwicklung, Implementierung und Tests • können mit exemplarisch vorgestellten Tools Systeme realisieren
Lehrinhalte
<ul style="list-style-type: none"> • Entwicklungsprozess für eingebettete Systeme im automotiven Umfeld • Grundlagen und technische Funktionsweise der Autosar Hard- und Softwarekomponenten • Systembeschreibung mit Modellierungswerkzeugen • Kommunikationsbeschreibung mit Signalmatrix • Parametrisierung und Generierung von Architekturkomponenten • Umgang mit der vorgestellten Werkzeugkette bis zur Implementierung
Literatur
<ul style="list-style-type: none"> • O. Kindel, M. Friedrich; Softwareentwicklung mit AUTOSAR; Dpunkt Verlag 2009; ISBN 978-3898645638 • O. Scheid; AUTOSAR Compendium; ISBN: 978-1502751522

Versionsnummer: Eintrag erstellt am:

2.2.3 Concurrent Computing

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Concurrent Computing				
LV alte PO (2013):	Concurrent Computing				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr wird angestrebt	WPF	5.0	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Gute Programmierkenntnisse in Java	WPF	Prof. I. Schiering
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
50% Klausur / mündliche Prüfung 50% Erstellung und Dokumentation von Rechnerprogrammen	Vorlesung (2 SWS), Labor (2 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • verstehen Begriffe zur Skalierung paralleler Implementierungen und können diese anwenden, • kennen Konzepte des Concurrent Computing für Shared Memory und Distributed Memory Architekturen, • realisieren Programme unter Verwendung der Konzepte
Lehrinhalte
<p>Die Studierenden lernen neben Grundlagen in einem allgemeinen Teil Ansätze für Shared Memory und Distributed Memor Architekturen kennen</p> <p>I. Allgemeiner Teil</p> <ul style="list-style-type: none"> • Gründe für die Entwicklung von Multicore-CPU's • Flynn'sche Taxonomie sowie Klassifikation von MIMD-Systemen nach Johnson • Skalierung paralleler Implementierungen (Speedup) sowie parallele Effizienz • Grenzen der Skalierung unter Beachtung der Gesetze von Amdahl und Gustafson <p>II. Beispiel für einen Shared Memory Ansatz ist die Java Concurrency API.</p> <ul style="list-style-type: none"> • Lambda Expressions • Atomare Datenstrukturen • Concurrent Collections • Konzepte zur Synchronisierung (Semaphore, Monitore, Locks, etc.) und klassische Probleme (Wechselseitiger Ausschluss, Producer-Consumer, Reader Writer, Dining Philosophers) • Phasen-Modelle (Barrier, Phaser) • Verwaltung von Threads (ThreadPool, Fork-Join Pool)

Alternativ wird eine funktionale Sprache (Clojure) verwendet:

- Einführung in die funktionale Programmierung
- Datenstrukturen und Nutzung von Libraries
- Concurrency in Clojure: Atoms, Refs, Vars
- Library core.async

III. Beispiel für einen Distributed Memory Ansatz ist MPI:

- Ideen des Message Passing auf Basis der MPJ-Implementierung der mpiJava-Spezifikation
- Grundlegender Aufbau von MPJ-Anwendungen (Prozessbegriff, Kommunikatoren sowie das Versenden und Empfangen von Nachrichten)
- Fallstudie zum parallelen Sortieren sowie Bestimmung des Speedups auf Basis eines Modells
- Kollektive Kommunikationsfunktionen zur Synchronisation von Prozessen, Verteilen und Einsammeln von Ergebnissen mittels Broadcast, Scatter(v) und Gather(v), Zusammenführen von (Teil-)Ergebnissen mittels Reduce und Allreduce
- Verwendung von Wildcards beim Empfangen von Daten
- Implementierung eigener Reduce-Operationen sowie Erkennen von einzuhaltenden Rahmenbedingungen
- Erstellung eigener Kommunikatoren sowie Betrachtung möglicher Anwendungsgebiete
- Nicht-blockierendes Senden und Empfangen von Nachrichten

Alternativ wird eine funktionale Sprache (Erlang) verwendet:

- Einführung in die funktionale Programmierung
- Datenstrukturen und Nutzung von Libraries
- Concurrency in Erlang: Message Passing
- Fehlerbehandlung
- Concurrency Patterns

Literatur

- Fernandez Javier, Java 7 Concurrency Cookbook (Quick Answers to Common Problems), Packt Publishing, 2012.
- Daniel Higginbotham, Clojure for the Brave and True: Learn the Ultimate Language and Become a Better Programmer, No Starch Press, 2015. (<http://www.braveclojure.com/>)
- Fred Hébert, Learn You Some Erlang for great good!, No Starch Press, 2013. (<http://learnyousomeerlang.com/>)
- Marc Snir et al., MPI: The Complete Reference (Vol. 1: The MPI Core) – 2nd Edition, MIT Press, 1998.
- William Gropp, Ewing Lusk, Anthony Skjellum, Using MPI - 2nd Edition: Portable Parallel Programming with the Message Passing Interface (Scientific and Engineering Computation) – 2nd Edition, 2000.
- Ananth Grama et al., Introduction to Parallel Computing – 2nd Edition, Addison Wesley, 2003.

Versionsnummer: Eintrag erstellt am:

2.2.4 Einführung in die Elektromobilität

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Einführung in die Elektromobilität				
LV alte PO (2013):	Einführung in die Elektromobilität				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1	WPF	5.0	150h, davon 40 Stunden Anwesenheitszeit und 110 Stunden für Vor- und Nachbereitung des Lehrstoffes

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Technisches Grundverständnis	WPF	Prof. G. Bikker
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
K90	Vorlesung, Übung	

Kompetenzziele (nach Bloom)
<p>Ziel ist es, Studierenden fachübergreifende Kenntnisse im Bereich der Elektromobilität zu vermitteln und sie schrittweise in die nötigen Grundlagen und Begrifflichkeiten einzuführen. Das dazu erforderliche Wissen wird in Vorlesungen vermittelt und durch integrierte Übungen gefestigt.</p> <p>Die Studierenden werden für ein gesamtheitliches Verständnis der Elektromobilität sensibilisiert. Durch die interdisziplinäre Gliederung der Veranstaltung wird das analytische und abstrakte Denkvermögen der Teilnehmer schrittweise gestärkt. Es sollen insbesondere die Zusammenhänge zwischen Energiespeicherung, unterschiedlichen Antriebskonzepten (Hybrid, Elektrofahrzeug), Mobilität, Umwelt und Smart Home Berücksichtigung finden.</p> <p>Nach Abschluss des Moduls haben die Studierenden ein fundiertes Verständnis für die grundlegenden Begriffe und Gesetzmäßigkeiten der Elektromobilität entwickelt. Sie erhalten ein Basiswissen über die Themenfelder der Elektromobilität und sind in der Lage, die erlernten Modelle anzuwenden sowie die Ergebnisse mit angemessenen Verfahren zu analysieren.</p>
Lehrinhalte
<ul style="list-style-type: none"> • Mobilität (Mobilität, Verkehr) und Umwelt (Mobilität versus Umwelt, Treiber der Elektromobilität, Verkehrslärm) • Smart Home (Lastprofile, dezentrale regenerative Energieerzeugung/Energiespeicherung, Energieverbrauch/Autarkie, Demand Side Management, private Elektromobilität, Lademanagement) • Elektrische Antriebe (Anforderungen, Aufbau und Eigenschaften elektrischer Fahrzeugantriebe) und (Hybride) Fahrzeugkonzepte (Strukturen hybrider Fahrzeuge, Micro-, Mild-, Full-, Power- und Plug-in-Hybrid, Paralleler und Serieller Hybrid, Powersplit Hybrid, Verbrauchspotenzial, Funktionsprinzipien Verbrennungsmotor, Betriebsstrategien hybrider Fahrzeuge) • Batterietechnik (Grundlagen Elektrochemie, Aufbau, Funktionen von Li-Ionen-Batterien, Eigenschaften)

von Li-Ionen-Batterien: Lebensdauer, Kosten, Sicherheit, Lagerung, Transport, Entsorgung)
Literatur
<ul style="list-style-type: none">• Kampker, A./ Vallée, D./ Schnettler, A. (Hrsg.) 2013: Elektromobilität. Grundlagen einer Zukunftstechnologie, Springer Vieweg, Berlin/Heidelberg• Öko-Institut/Optum 2011 (Ergebnisbroschüre): Umweltentlastungspotenziale von Elektrofahrzeugen – Integrierte Betrachtung von Fahrzeugnutzung und Energiewirtschaft, Berlin, 09/2011• Umweltbundesamt (UBA) 2014: Umweltverträglicher Verkehr 2050: Argumente für eine Mobilitätsstrategie für Deutschland, Berlin, 02/2014• BEE/InnoZ 2015: Die neue Verkehrswelt - Mobilität im Zeichen des Überflusses: schlau organisiert, effizient, bequem und nachhaltig unterwegs, Berlin, 01/2015• Fischer, R. 2013: Elektrische Maschinen, 16. Auflage, Carl Hanser Verlag, München• Wallentowitz, H./Freialdenhoven, A. 2011: Strategien zur Elektrifizierung des Antriebsstranges, Vieweg+Teubner, Wiesbaden• Hofmann, P. 2014: Hybridfahrzeuge. Ein alternatives Antriebssystem für die Zukunft, Springer-Verlag, Wiesbaden• Stan, C. 2012: Alternative Antriebe für Automobile. Hybridsysteme, Brennstoffzellen, alternative Energieträger, Springer-Verlag, Berlin Heidelberg• Reif, K. 2010: Konventioneller Antriebsstrang und Hybridantriebe. Mit Brennstoffzellen und alternativen Kraftstoffen, Vieweg+Teubner Verlag, Wiesbaden• Jossen, A. / Weydanz W. / 2006: Moderne Akkumulatoren, Verlag Ubooks, Untermeitlingen, ISBN 3-939359-11-4• Bieger, T./zu Knyphausen-Aufseß, D./Krys, C. 2011: Innovative Geschäftsmodelle, Springer-Verlag, Berlin/Heidelberg <p>Weitere Literatur wird in der Veranstaltung bekanntgegeben.</p>

Versionsnummer: Eintrag erstellt am:

2.2.5 Embedded Linux

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Embedded Linux				
LV alte PO (2013):	Embedded Linux				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	unregelmäßig	WPF	5.0	150h, davon ca. 30% Kontaktstudium, ca. 70% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Tiefere Kenntnisse in der Programmiersprache C	WPF	Prof. D. Justen
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
100% Mündliche Prüfung	Vorlesung (4 SWS) mit praktischen Aufgaben zur Bearbeitung im Eigenstudium	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen die wesentlichen Linux Strukturen (Kernel, Library, Init-System) • haben ein Verständnis für die Aufteilung zwischen Kernaufgaben und Useraufgaben • kennen die Systemintegration des Treibers (Proc-, Sys und Device-Filesystem) • kennen die wesentlichen Zugriffsverfahren auf einen DeviceTreiber • kennen den Aufbau eines Device Treibers
Lehrinhalte
<p>Auf Basis des Lego Mindstorm EV3 Systems (Angstorm Linux) lernen die Studierende</p> <ul style="list-style-type: none"> • den Zugriff auf ein embedded Linux System • die Erstellung eigener Programme (POSIX Standard) • das Debuggen auf embedded Systemen unter Linux • das Erstellen eines eigenen Treibers zur Ansteuerung spezieller Hardware • das Erstellen einer eigenen Shared Library zum komfortablen Zugriff auf den Treiber
Literatur
<ul style="list-style-type: none"> • Jonathan Corbet; Alessandro Rubini; Greg Kroah-Hartman; "Linux Device Drivers", 3rd Edition by Jonathan Corbet (2005-02-17) • Jürgen Quade; "Linux-Treiber entwickeln: Eine systematische Einführung in die Gerätetreiber- und Kernelprogrammierung"; dpunkt • Michael Beck, Harald Böhme; „Linux-Kernel-programmierung“; Addison-Wesley • Karim Yaghmour; „Building Embedded Linux Systems“;

Versionsnummer: Eintrag erstellt am:

2.2.6 Fahrerassistenzsysteme

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Fahrerassistenzsysteme				
LV alte PO (2013):	Fahrerassistenzsysteme				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr	WPF	5.0	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Vernetzte Systeme	WPF	F. Pramme
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
60% Klausur / mündliche Prüfung 40% Referat / experimentelle Arbeit	Unterricht, Labor mit Projektvorträgen, Projektarbeit (4 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • lernen die wesentlichen elektronischen Fahrzeugsysteme und deren Anwendungsbereiche kennen • lernen die Grundlagen der Fahrerassistenzsysteme kennen • entwerfen unter der Berücksichtigung von Funktionaler Sicherheit insb. Sicherheitsanforderungen • simulieren selbst entwickelte Fahrerassistenzsysteme und beschäftigen sich mit virtueller Integration, Umfeld-Sensorik und Umwelt • entwerfen Konzepte zur Datenfusion, Umfeld-Präsentation und Car 2 X Kommunikation
Lehrinhalte
<ul style="list-style-type: none"> • Grundlagen der Fahrerassistenzsysteme • Rahmenbedingungen der Entwicklung • Funktionale Sicherheit (Rückverfolgbarkeit, Verifikation und Validierung) • Virtuelle Integration und Test von FAS • Sensorik und Aktuatorik für FAS • Maschinelles Sehen, Datenfusion und Umfeld-Präsentation • Car 2 X Kommunikation und Infrastruktur • Autonomes Fahren
Literatur
<ul style="list-style-type: none"> • Konrad Reif; "Fahrstabilisierungssysteme und Fahrerassistenzsysteme (Bosch Fachinformation Automobil)"; Vieweg+Teubner Verlag; Auflage: 2010 (25. Juni 2010); ISBN-13: 978-3834813145

Versionsnummer: Eintrag erstellt am:

2.2.7 Hard- und Software intelligenter Systeme (HuSiS)

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):					
Lehrveranstaltung:	Hard- und Software intelligenter Systeme (HuSiS)				
LV alte PO (2013):					
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4.	1	2	Wahlpflicht	5	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
gemäß PO	Teamarbeit und selbstständige Problemlösung	Dipl.-Ing. K. Dammann
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
EA und PA	Eigenständige Projektarbeiten im Team.	

Kompetenzziele (nach Bloom)
Echtzeit Erkennung.
Lehrinhalte
Softwareentwicklung und Hardwareanwendung von embedded Systemen am Beispiel des Arduino, benchmarktest systemeigener Bibliotheken im Vergleich zu „reiner“ C Programmierung, diskreter Aufbau von Prozessorsystemen, Ausarbeitung von verschiedenen Projekten, Präsentation der Projekte.
Literatur

Versionsnummer: Eintrag erstellt am:

2.2.8 Interdisziplinäres Projekt

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Interdisziplinäres Projekt				
LV alte PO (2013):	Interdisziplinäres Projekt				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	2	WPF	5.0	150h

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Prof. R. Gerndt
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
PF	Vorlesung, Seminar, Labor	

Kompetenzziele (nach Bloom)
<ul style="list-style-type: none"> • Wissen über das Gebiet der Projektplanung, -durchführung und des -abschluss im transdisziplinären und/oder inter-kulturellen Kontext, • Verstehen von nicht-funktionalen Zusammenhängen in der Realisierung technischer Lösungen, • Anwendung und praktische Erprobung des Wissens in transdisziplinären und/oder inter-kulturellen Arbeitsgruppen
Lehrinhalte
<ul style="list-style-type: none"> • Elementare Grundlagen einer an die Informatik angrenzende Fachdisziplin, bzw. eines für Informatik (Forschung, Industrie, Markt) relevanten Kulturkreises • Planung eines (Informatik-) Projektes unter besonderen Randbedingungen • Projektdurchführung in einem transdisziplinären und/oder inter-kulturellen Umfeld • Berichtswesen und Präsentation von Projektzielen • ggf. Vertiefen einer Fremdsprache
Literatur
<p>www.erasmusplus.de</p>

Versionsnummer: Eintrag erstellt am:

2.2.9 Java Enterprise Edition (Java EE)

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Java Enterprise Edition (Java EE)				
LV alte PO (2013):	Java Enterprise Edition (Java EE)				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	unregelmäßig	WPF	5.0	150h, davon ca. 20% Kontaktstudium, ca. 80% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Java-Kenntnisse	WPF	Prof. B. Müller
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Projekt	Vorlesung und Projektarbeit	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen wesentlichen Anwendungsbereiche betrieblicher Informationssysteme • können die von Java EE angebotenen Systeme den Anwendungsbereichen zuordnen • können Anforderungen in lauffähige Software umsetzen • kennen die Vor- und Nachteile verschiedener Implementierungsalternativen
Lehrinhalte
<p>Java Enterprise Edition (Bereiche und Lösungsansätze)</p> <ul style="list-style-type: none"> • JavaServer Faces • Enterprise JavaBeans • Context and Dependency Injection • Java Persistence API • Web-Services • Java Messaging Service • Bean Validation
Literatur
<ul style="list-style-type: none"> • Oracle. Java EE Tutorial (online). • Dirk Weil. Java EE 7. entwickler.press, 2013. • Arun Gupta. Java EE 7. O'Reilly, 2013. • Marcus Schießler, Martin Schmollinger. Workshop Java EE 7. dpunkt.verlag, 2015.

Versionsnummer: Eintrag erstellt am:

2.2.10 Malware-Labor

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Malware-Labor				
LV alte PO (2013):	Malware-Labor				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	semesterweise	WPF	5.0	150h, davon 60 Stunden Anwesenheitszeit und 90 Stunden für Vor- und Nachbereitung des Lehrstoffes, Prüfungsabnahme

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Vorlesung IT Sicherheit bzw. eine ähnliche, sicherheitsrelevante LV	WPF	Prof. Sh. Gharaei
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
EA	Einleitende Vorlesung (1 SWS) plus Labor (3 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende können</p> <ul style="list-style-type: none"> • systematische Verbreitung einer Untersuchung der Windows-Executives beim Verdacht einer Malware • eine Auswahl und Einsatz von Tools zwecks Malware-Analyse selbstständig durchführen • Malware-Samples statisch und dynamisch analysieren • die Fähigkeit entwickeln, die Nebeneffekte während einer dynamischen Analyse zu erkennen, zu dokumentieren und zu behandeln
Lehrinhalte
<p>Das Experimentieren mit Malware-Samples ist der Fokus dieser Labor-LV. Dieses ist eine strukturierte, aufwändige Tätigkeit, die folgende wesentliche Merkmale aufweist:</p> <ul style="list-style-type: none"> • Jedes Experiment besteht aus mehreren Phasen • Jede Phase umfasst mehrere Einzelschritte • Im Laufe der Untersuchung werden diverse statische und dynamische Analysen durchgeführt • Für jede Analyse ist i.d.R. der Einsatz von dedizierten Tools notwendig • ggf., jedoch nicht immer, müssen eigene spezifische Skripte ausgearbeitet werden • Die Vor- und Nachbereitung der Experimente ist ein wesentlicher Bestandteil eines Experiments
Literatur
<p>Eine Liste der aktuellen Literatur und weiteren Quellen wird vor der LV bekanntgegeben.</p>

Versionsnummer: Eintrag erstellt am:

2.2.11 Medizinische Informatik

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):					
Lehrveranstaltung:	Medizinische Informatik				
LV alte PO (2013):					
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
5.	1	1	Wahlpflicht	5.0	150h (ca. 40% Kontakt-, 60% Eigenstudium)

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Prof. Dr. Wolfram Ludwig
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
K1,5h/PA/EA/M	Vorlesung, Übungen, Fallstudien	

Kompetenzziele (nach Bloom)

Die Studierenden kennen

- Grundbegriffe der Medizinischen Informatik und können diese sicher anwenden.
- zentrale Akteure des deutschen Gesundheitssystems und verstehen ihre jeweiligen Aufgaben.
- die gesundheitsökonomischen Rahmenbedingungen des deutschen Gesundheitssystems und verstehen die Implikationen, die sich hieraus für die Einrichtungen des Gesundheitswesens und damit für die Medizinische Informatik ergeben.
- die Komponenten von Krankenhausinformationssystemen, elektronischen Patientenakten und Arztpraxisinformationssystemen und können diese beschreiben.
- ausgewählte Anwendungssysteme der Telemedizin und der Assistierenden Gesundheitstechnologien und können diese beschreiben.
- ausgewählte IT-Standards, Taxonomien und Klassifikationen der Medizin und der Medizinischen Informatik und können diese zur Codierung von Daten anwenden.
- die Herausforderungen der Interoperabilität und einrichtungsübergreifenden Kommunikation im Gesundheitswesen und können diese beschreiben. Q9

Lehrinhalte

Die Veranstaltung vermittelt einen Überblick über folgende Themengebiete:

- Gesundheitsökonomie
- Krankenhausinformationssysteme
- elektronische Patientenakten
- Arztpraxisinformationssysteme
- Telemedizin
- Assistierende Gesundheitstechnologien
- Interoperabilität und IT-Standards

Literatur

- Dickhaus H, Knaup-Gregori P. Biomedizinische Technik - Medizinische Informatik. Bd. 6. De Gruyter: 2015
- Dugas M. Medizininformatik. Ein Kompendium für Studium und Praxis. Springer Vieweg: 2017

Versionsnummer: Eintrag erstellt am:

2.2.12 Pervasive Systeme

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Pervasive Systeme				
LV alte PO (2013):	Pervasive Systeme				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr	WPF	5.0	150h, davon ca. 40h Kontaktstudium, ca. 110h Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Vernetzte Systeme	WPF	F. Pramme
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
60% Klausur / mündliche Prüfung 40% Referat / experimentelle Arbeit	Unterricht, Labor mit Projektvorträgen, Projektarbeit (4 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • lernen die wesentlichen Begriffe, Methoden sowie Anwendungsbereiche kennen • entwerfen unter der Berücksichtigung von Funktionaler- und Datensicherheit pervasive Systeme • simulieren selbst entwickelte Systeme und beschäftigen sich mit der Integration von Umfeld-Sensorik und Umwelt • entwerfen Konzepte zur Datenfusion
Lehrinhalte
<ul style="list-style-type: none"> • Grundlagen der Pervasiven Systeme • Anwendungsfelder des Pervasiven Computing • Technologien des Pervasiven Computing (Sensorik, Kommunikation, Sicherheit) • Sozio-ökonomische Voraussetzungen und Auswirkungen des Pervasiven Computing • Mensch-Maschine-Interaktion • Digitalisierung von unterschiedlichen Anwendungsfällen
Literatur
<ul style="list-style-type: none"> • Ciprian Dobre; "Pervasive Computing – Next Generation Platforms for Intelligent Data Collection"; Elsevier Verlag; Auflage: 2016; ISBN-13: 978-0128037027 • Natalia Silvis-Cividjian; „Pervasive Computing – Engineering Smart Systems“; Springer; Auflage: 1 2017

Versionsnummer: Eintrag erstellt am:

2.2.13 Programmierparadigmen C++

Studiengang: Informatik (M. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Programmierparadigmen C++				
LV alte PO (2013):	Programmierparadigmen C++				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
3-5	1	jährlich Winter	Wahlpflicht	5.0	150h, davon ca. 40% Kontaktstudium (Präsens bzw. virtuell durch BBB), ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
gemäß PO	alle BA und Master	Hon. Prof. H. Helmke
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
abhängig von Teilnehmerzahl	begleitende Übungsaufgaben zur eigenständigen Bearbeitung und Vertiefung der Vorlesungsinhalte, die teilweise aufeinander aufbauen,	

Kompetenzziele (nach Bloom)
<ul style="list-style-type: none"> • Die verschiedenen Programmierparadigmen und C++ verstehen und selbstständig anwenden können • testbasierte Software-Entwicklung
Lehrinhalte
<p>Einführung, Worthäufigkeiten in einer Datei zählen: Datentypen, Steueranweisungen Funktionen und Strukturen Include-Wächter, Header-Dateien, Werte- und Referenz-Semantik, Zeiger, Heap- und Stackspeicher Klassen als Abstrakte Datentypen tiefe und flache Kopie: Kopierkonstruktor, Nutzung von LogTrace Operatoren, Verschiebeoperatoren Templates Programmieren mit der STL (Standard Template Library) Polymorphie, Intelligente Zeiger Container Lambda-Ausdrücke</p>
Literatur

Versionsnummer: Eintrag erstellt am:

2.2.14 RFID - Radio Frequency Identification

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	RFID - Radio Frequency Identification				
LV alte PO (2013):	RFID - Radio Frequency Identification				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr	WPF	5.0	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Alle Prüfungen des 1. bis einschließlich 2. Semesters bestanden	WPF	Prof. J. Kreyszig
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Experimentelle Arbeit oder Vortrag als Teilleistung und Klausur 1,5h oder mündliche Prüfung	Seminaristische Vorlesung und Projektarbeit oder Vortrag	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen und verstehen die Grundlagen der RFID-Technik (Radio Frequency Identification) • haben erste praktische Erfahrungen mit RFID-Systemen
Lehrinhalte
<p>Einführung und Überblick</p> <ul style="list-style-type: none"> • Anwendungsbeispiele (Logistik, Automobil, Robotik, Zugangskontrolle, Nahrungsmittel, Diebstahlssicherung, Materialwirtschaft, Sport, usw.) • Kommunikation zwischen Lesegerät und Transponder (Aktiv, Passiv, Semi-Aktiv, Bluetooth-Beacon usw.) • Vergleich der Systeme und ihrer Einsatzmöglichkeiten • Energie und Datenübertragung; Einschränkungen; Kollisionserkennung; Reichweiten • Beispiele von implementierten RFID Systemen <p>Experimentelle Arbeit:</p> <ul style="list-style-type: none"> • praktische Erfahrung mit RFID
Literatur
<ul style="list-style-type: none"> • Finkenzeller, K.: RFID-Handbuch: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC, Carl Hanser Verlag • Kern, C.: Anwendung von RFID-Systemen, Springer Verlag

Versionsnummer: Eintrag erstellt am:

2.2.15 Software für sichere Systeme

Studiengang: Informatik (M. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Software für sichere Systeme				
LV alte PO (2013):					
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	sporadisch	WPF	5.0	150h, davon ca. 30% Kontaktstudium, ca. 70% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
Grundwissen über Softwaretechnik	WPF	Prof. M. Huhn
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
50 % M, 50 % ED + RP (Teilleistung d. Bearbeitung der Laboraufgaben: Erstellung+Dokumentation)	Vorlesung (2 SWS) und Labor (2 SWS) mit praktischen Aufgaben zur Bearbeitung im Eigenstudium	

Kompetenzziele (nach Bloom)
<p>Die Studierenden</p> <ul style="list-style-type: none"> • beherrschen die Grundbegriffe der funktionalen Sicherheit und verwenden Prozesse und Anforderungen zur Entwicklung sicherheitskritischer Software gemäß der geltenden europäischen und amerikanischen Sicherheitsnormen bei Entwicklungstätigkeiten für sicherheitskritische Systeme • können Methoden zur Sicherheitsanalyse anwenden • planen, entwerfen und verifizieren sicherheitskritische Software mit SCADE, einer modellbasierten, integrierten Entwicklungsumgebung (IDE) aus der Bahn-, Automotive und Luftfahrtbranche.
Lehrinhalte
<ul style="list-style-type: none"> • Funktionale Sicherheit und der Begriff des Risikos • Methoden zur Gefährdungs- und Sicherheitsanalyse (FMEA, FTA, STAMP) • Sicherheitsnormen • Softwaresicherheit und Architekturmuster für sichere Software • Anforderungen an sicherheitskritische SW • Scade Suite <ul style="list-style-type: none"> - die synchrone Sprache Scade: Konzepte, Syntax, Semantik - die Scade Entwicklungsumgebung: Vorgehensmodell, Scade Suite, Simulation, Code-Generierung mit dem zertifizierten Code Generator, Testen • Validierung und Verifikation • Sicherheitsnachweisführung
Literatur

Versionsnummer: Eintrag erstellt am:

2.2.16 Virtualisierung

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Virtualisierung				
LV alte PO (2013):	Virtualisierung				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	1x pro Jahr	WPF	5.0	150h, davon ca. 40% Kontaktstudium, ca. 60% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	G. Kircher
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
60% Klausur / mündliche Prüfung 40% Referat / experimentelle Arbeit	Unterricht, Labor mit Projektvorträgen, Projektarbeit (4 SWS)	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • kennen die Virtualisierungskonzepte und deren Anwendungsbereiche • entwerfen, implementieren und betreiben Serversysteme mit virtualisierten Komponenten • kennen die Einsatzbereich für Konsolidierung, FailOver u./o. Replikation • kennen Sicherheitskonzepte und Anwendungsbereiche bei Embedded Systems
Lehrinhalte
<ul style="list-style-type: none"> • Grundlagen und technische Funktionsweise • Virtualisierungskonzepte • Einsatzgebiete und Beispielanwendungen, Cloud • Installation und Anwendung kommerziell verfügbarer Virtualisierungslösungen (z.B. VMware und HyperV) • Administration von Virtuellen Maschinen, Datensicherung, Sicherheit • Virtualisierung von eingebetteten Systemen
Literatur
<ul style="list-style-type: none"> • J.M.Portnoy, Virtualisierung für Einsteiger, Wiley 2012, ISBN 3-527-76023-7 • H.Rathod / J.Townsend, Virtualization 2.0, Wiley 2014, ISBN: 978-1-119-02432-3

Versionsnummer: Eintrag erstellt am:

2.2.17 Wissenschaftstheorie

Studiengang: Informatik (B. Sc.), Informatik im Praxisverbund (B. Sc.), Wirtschaftsinformatik (B. Sc.), Wirtschaftsinformatik im Praxisverbund (B. Sc.)					
Modul:	Qualifikationsmodul				
Modul alte PO (2013):	Qualifikationsmodul				
Lehrveranstaltung:	Wissenschaftstheorie				
LV alte PO (2013):	Wissenschaftstheorie				
Semester	Dauer (Sem.)	Häufigkeit (pro Jahr)	Art	ECTS-Punkte	Studentische Arbeitsbelastung
4./5. Sem.	1	2x pro Jahr	WPF	3.0	75h, davon ca. 5% Kontaktstudium, ca. 95% Eigenstudium

Voraussetzungen für die Teilnahme	Verwendbarkeit	Modulverantwortliche(r)
keine	WPF	Prof. W. Pekrun
Prüfungsform / Prüfungsdauer (Voraussetzung für die Vergabe von ECTS-Punkten)	Vorgesehene Lehr- und Lernmethoden/-formen	
Referat	Seminar	

Kompetenzziele (nach Bloom)
<p>Studierende</p> <ul style="list-style-type: none"> • können in einem Teilbereich der unter „Lehrinhalte“ genannten Bereiche kompetent Inhalte recherchieren, aufbereiten, vortragen und in der Diskussion vertreten
Lehrinhalte
<p>Aus folgenden Bereichen können Inhalte gewählt werden:</p> <ol style="list-style-type: none"> 1. Übergreifende methodische Fragen der Wissenschaft (grundlegende Methoden der Wissenschaften, Methodenübertragungen zwischen Wissenschaften, etc.) 2. Themen rund um Wissenschaft (z.B. Wissenschaft und Öffentlichkeit, ethische Fragen, Zukunft der Wissenschaft(en), etc.) 3. Sehr fundamentale wissenschaftliche Erkenntnisse (z.B. in Quantenmechanik, Kosmologie, Evolutionstheorie, etc.)
Literatur

Versionsnummer: Eintrag erstellt am:

3 Dokumenthistorie

- 25.09.2018 Ersterstellung
- 13.09.2019 Korrekturen bei Verwendbarkeit, Aufnahme „Autonomes Fahren“ und „Pervasive Systeme“
- 25.09.2020 Aufnahme „Hard- und Software intelligenter Systeme (HuSiS)“, „Medizinische Informatik“,
„Software für sichere Systeme“, „Programmierparadigmen C++“, Wegfall „Quantum
Computing“