

Modulhandbuch für den Studiengang Bachelor Medieninformatik

Begriffserläuterungen	3
1. Studienabschnitt	4
Theoretische Grundlagen	4
Grundlagen der Informatik	5
Mathematik I (Algebra)	7
Mathematik II (Analysis)	8
Theoretische Informatik	9
Praktische Informatik	10
Programmieren I	11
Programmieren II	13
Digitale Medien	13
Gestaltungs- und Medienlehre I	14
Gestaltungs- und Medienlehre II	16
Grundlagen der Medieninformatik	18
Fremdsprache	18
Englisch I/II	19
2. Studienabschnitt	19
Theoretische Grundlagen	19
Algorithmen und Datenstrukturen	20
Mathematik III (Geometrie, Transformationen)	22
Mathematik IV (Statistik)	23
Praktische Informatik	23
Betriebssysteme und Rechnerarchitektur	24
Datenbanken	26
Rechnerkommunikation	28
Software Engineering	30
Softwarearchitektur	32
Web-Programmierung	34
Digitale Medien	35
Interdisziplinäre Projektarbeit (9LP)	36
Medienanalyse	36
Medienverarbeitung	37
Mediensynthese	37
Computergraphik	38
Medienkonzeption	39
Mensch-Computer-Interaktion	39
Interaktive Systeme	40
Software-Ergonomie	41
Praktisches Studiensemester	41
Praktikum	42
Praxisseminar	43
Projektmanagement	44
Fachwissenschaftliche Wahlpflichtfächer	45
Schwerpunkt-FWPF (min. 12SWS/15LP, max. SWS/LP)	45
Agile Methoden in ausgewählten Anwendungsbereichen	46
Aktuelle Trends im Einsatz von Projektmanagement-Methoden	48
Algorithmische Graphentheorie	49
Clean Code in Java-Projekten	50
Conversational Interfaces - Design und Implementierung	52
Datenmodellierung mit XML	54
Datenschutz	56
Digitale Musikinstrumente mit Smartphones und PureData	58
E-Commerce-Recht	60
Einführung in die Funktionale Programmierung	61
Einführung in Operations-Research	63
Einstieg in C++ für C#-Programmierer	65

Entwicklung eines autonom fahrenden Modellfahrzeugs	67
Graphical User Interface Design and Information Visualization	69
Grundlagen Content-Management-Systeme	70
Grundlagen der Computerspielentwicklung	72
Grundlagen des maschinellen Lernens	74
Grundlagen des Softwaretests	75
In-Memory Computing am Beispiel von SAP HANA	77
Lean Startup	79
Logik	81
NoSQL-Datenbanken	82
Programmierung und Technologie betrieblicher Standardsoftware (SAP-Praktikum)	83
Selbstlokalisierung	84
Software-Zuverlässigkeit: Theorie und Praxis	86
Technical Computing	87
Text Analytics	88
Verteilte Systeme - Grundlagen und Algorithmen	90
Virtuelle Maschinen	92
Visualisierung	94
Voice-over-IP	96
wählbar außerhalb Schwerpunkt	97
Betriebssysteme für einfache Mikrocontroller	98
Datenbanken II in der Wirtschaftsinformatik	100
Digitalisierung und Nachhaltigkeit	102
Einsatz von Simulations-Software	105
Electronic Commerce	107
Embedded Systems	109
Finanzmathematik	111
Integrierte Warenwirtschaft mit einem ERP-System	112
Praxis des Software-Tests eingebetteter Systeme	114
Programmieren von Mikrocontrollern	116
Spezielle Rechnerarchitekturen - vom Embedded System zum Supercomputer	118
Spieltheorie	119
Abschlussarbeit	120
Bachelorarbeit	121
Seminar zur Bachelorarbeit	122
Hilfsmittel	123

Begriffserläuterungen

ECTS	<p>European Credit Transfer System: Diese Vereinbarungen zur Anrechnung, Übertragung und Akkumulierung von Studienleistungen basieren auf dem Arbeitspensum, das Studierende durchzuführen haben, um die Ziele des Lernprogramms zu erreichen. Für jede studienbezogene Leistung wird der voraussichtliche durchschnittliche Arbeitsaufwand angesetzt und auf das Studienvolumen angerechnet. Der Arbeitsaufwand umfasst Präsenzzeit und Selbststudium ebenso wie die Zeit für die Prüfungsleistungen, die notwendig sind, um die Ziele des vorher definierten Lernprogramms zu erreichen. Mit dem ECTS können Studienleistungen international angerechnet und übertragen werden.</p>
Arbeitsaufwand (Workload) und Leistungspunkte (ECTS-LP)	<p>Der Arbeitsaufwand der Studierenden wird im ECTS in credits (credit points) angegeben.</p> <p>Deutsche Übersetzungen für credit point sind die Begriffe Leistungspunkt oder ECTS-Punkt. Ein Arbeitsaufwand von 30 Zeitstunden bedeutet einen Leistungspunkt.</p> <p>Der Arbeitsaufwand von Vollzeitstudierenden entspricht 60 Leistungspunkten pro Studienjahr, also 30 Leistungspunkten pro Semester. Das sind 1.800 Stunden pro Jahr oder 45 Wochen/Jahr mit 40 Stunden/Woche.</p> <p>Der Arbeitsaufwand setzt sich aus verschiedenen Bestandteilen zusammen:</p> <ul style="list-style-type: none">• Präsenzzeit,• Zeit für die Vor- und Nachbereitung des Vorlesungsstoffs,• Zeit für die Vorbereitung von Vorträgen und Präsentationen,• Zeit für die Erstellung eines Projekts,• Zeit für die Ausarbeitung einer Studienarbeit,• Zeit für notwendiges Selbststudium,• Zeit für die Vorbereitung auf mündliche oder schriftliche Prüfungen. <p>Die siebensemestrigen Bachelorstudiengänge bescheinigen erfolgreichen Studierenden also 210 ECTS-LP, die dreisemestrigen Masterstudiengänge weitere 90 ECTS-LP. Damit ist die Forderung nach 300 ECTS-LP für ein erfolgreich abgeschlossenes Masterstudium erfüllt.</p>
Semesterwochenstunden und Präsenzzeit	<p>Eine Semesterwochenstunde ist die periodisch wiederkehrende Lehrinheit in einem Modul, in der Regel im Rhythmus von einer oder zwei Wochen. Dabei wird eine Präsenz von 45 Minuten plus Wegzeiten gerechnet, sodass die Vorlesungsstunde als eine Zeitstunde gewertet wird.</p> <p>Wir rechnen mit einer Vorlesungszeit von 16 Wochen pro Semester, wodurch sich aus der Zahl der Semesterwochenstunden die geforderte Präsenzzeit („Kontaktzeit“) direkt ableitet:</p> <p>1 SWS entspricht 16 Stunden Präsenzzeit. Natürlich wird gerundet.</p>
Module	<p>Das Studium ist inhaltlich in Module aufgeteilt, die zur besseren Übersicht in Modulgruppen zusammengefasst sind.</p> <p>Jedes Modul wird mit einer Prüfung abgeschlossen und ist mit Leistungspunkten versehen, die dem Studierenden bei erfolgreichem Ablegen der Prüfung gutgeschrieben werden.</p>
Studienbegleitende Prüfungen	<p>Sämtliche Prüfungen erfolgen über das gesamte Studium verteilt studienbegleitend und stehen in direktem Bezug zur Lehrveranstaltung. Prüfungsbestandteile können je nach Lehrveranstaltung veranstaltungsbegleitend oder nach Abschluss des Moduls stattfinden, beispielsweise als Referat, Klausurarbeit, mündliche Prüfung, Hausarbeit mit Kolloquium, Entwurf mit Kolloquium, Laborbericht, Exkursionsbericht oder einer Kombination. In den Beschreibungen der einzelnen Module wird im Modulhandbuch die jeweilige Prüfungsform festgelegt.</p> <p>Gemäß der Studien- und Prüfungsordnung kann Fachwissenschaftliche</p>

Wahlpflichtmodule nur ableisten,
wer alle Module des ersten Studienabschnitts bestanden hat.
Einzelheiten zur Organisation der Fachwissenschaftlichen Wahlpflichtmodule regelt
der Studienplan.

Liste Hilfsmittel

- 1) keine Hilfsmittel
- 2) unbeschränkt
- 3) Vorlesungsmitschrift
- 4) Vorlesungsumdrucke
- 5) selbstgefertigte Arbeitsunterlagen (DinA4 Seitenzahl)
- 6) mathematische Formelsammlung
- 7) Taschenrechner, nicht programmierbar (Taschenrechner mit vollständiger alphanumerischer Tastatur und/oder Graphikdisplay sind nicht erlaubt)
- 8) Gesetzestexte, z.B. BGB, UrhG, PatG, UWG, Betr.VG, BDSG, StGB, Stopp, TKG
- 9) Lehrbuch

Grundlagen der Informatik

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Theoretische Grundlagen
Modul	Grundlagen der Informatik
Modulverantwortliche	Michael Zapf, Florian Gallwitz, Jörg Roth, Thomas Fuhr, Rainer Weber
Dozent(en)	Rainer Weber, Michael Zapf
Vorkenntnisse	Schulwissen
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit 95 Stunden Vor- und Nachbereitung des Lehrstoffs 20 Stunden für Prüfung und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung mit 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Schaffen von Grundlagen, insbesondere im Hinblick auf technische Abläufe in einem Computersystem sowie auf die Repräsentation von Informationen
Lernziel	Kenntnis und Verständnis der Repräsentation von Daten wie ganze und reelle Zahlen und Zeichenketten; Verständnis der mathematischen Grundlagen von Rechenoperationen auf diesen Repräsentationen und selbständige Anwendung; Befähigung zur Konvertierung zwischen Repräsentationen mit besonderem Gewicht auf dem Binär-, Oktal- und Hexadezimalsystem. Analyse von logischen Schaltungen; Verständnis der mathematisch-logischen Grundlagen; selbständige Gestaltung von Logikschaltungen auf Basis eines vorgegebenen Verhaltens; Verständnis für den grundlegenden Entwurf von Computersystemen; Analyse und Erstellung von Maschinensprache- und Assemblerprogrammen.
Schlüsselqualifikation	Fähigkeit zu abstraktem mathematischen Denken und Schließen, technisches Verständnis
Lehrinhalte	<ul style="list-style-type: none">• Repräsentation positiver ganzer Zahlen in verschiedenen Stellenwertsystemen• Negative ganze Zahlen in verschiedenen Stellenwertsystemen• Reelle Zahlen im Standardformat (IEEE-754)• Rechenoperationen und Konvertierungen• Boolesche Algebren• Boolesche Ausdrücke und Funktionen• Schaltnetze und Schaltwerke• Allgemeine Computerarchitektur• Maschinensprache am Beispiel der MIPS-Architektur• Assemblersprache der MIPS-Architektur
Literatur	<ul style="list-style-type: none">• R. Klar: Digitale Rechenautomaten, Berlin, 1988, Walter de Gruyter.

- W. Oberschelp, G. Vossen: Rechneraufbau und Rechnerstrukturen, 9. Auflage, München, 2003, Oldenbourg Verlag.
- Y. Patt, S. Patel: Introduction to Computing Systems - From Bits to Gates and Beyond, 2nd ed., 2004, McGraw Hill.
- D. A. Patterson, J. L. Hennessy: Computer Organization & Design, 2nd ed., San Francisco, CA, 1998, Morgan Kaufman Publishers Inc.
- A. S. Tanenbaum: Computerarchitektur. Strukturen - Konzepte - Grundlagen. 2005. Pearson Studium.
- H. Delfs, H. Knebl, C. Schiedermeier: Grundlagen der Informatik. Nürnberger Hochschulschriften für Technik und Wirtschaft. Nürnberg, 2001, Nano-Verlag.

Leistungsnachweis

Schriftliche Prüfung (90 min)

Mathematik I (Algebra)

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Theoretische Grundlagen
Modul	Mathematik I (Algebra)
Modulverantwortliche	Herbert Urban, Harald Stieber, Alexander Hufnagel, Elke Wilczok, Christian Scherr
Dozent(en)	Alexander Hufnagel, Christian Scherr, Herbert Urban
Vorkenntnisse	Schulwissen
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit 95 Stunden Vor- und Nachbereitung des Lehrstoffs 20 Stunden für Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung mathematischer Grundlagen
Lernziel	Begriffe und Strukturen aus verschiedenen grundlegenden Gebieten der Mathematik (Zahlen, Aussagenlogik, Mengen, Relationen und Funktionen) definieren und verstehen. Methoden der Kombinatorik zur Lösung von Zählproblemen einsetzen. Begriffe und Aussagen der elementaren Zahlentheorie aufstellen und Verständnis für spätere Anwendungen (insbesondere in der Kryptographie) erlangen. Rechenverfahren im Umgang mit komplexen Zahlen einüben und durchführen. Grundlegende Begriffe der Algebra kennen und in ihrer Bedeutung verstehen.
Schlüsselqualifikation	Fähigkeit zu abstraktem mathematischen Denken und Schließen, Fähigkeit zu komplexem Denken
Lehrinhalte	Zahlen, Aussagenlogik und Mengen, Relationen und Funktionen, Kombinatorik, Elemente der Zahlentheorie, komplexe Zahlen, Algebra (Gruppen, Ringe, Körper)
Literatur	G. Teschl, S. Teschl: Mathematik für Informatiker (Springer)
Leistungsnachweis	Schriftliche Prüfung (90 min.)

Mathematik II (Analysis)

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Theoretische Grundlagen
Modul	Mathematik II (Analysis)
Modulverantwortliche	Harald Stieber, Alexander Hufnagel, Herbert Urban, Elke Wilczok, Christian Scherr
Dozent(en)	Jürgen Bolik, Alexander Hufnagel, Christian Scherr, Herbert Urban, Elke Wilczok
Vorkenntnisse	Schulwissen, Mathematik I
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit 95 Stunden Vor- und Nachbereitung des Lehrstoffs 20 Stunden für Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung mathematischer Grundlagen
Lernziel	Grundlegende Begriffe und Strukturen der Linearen Algebra und der Analysis definieren und verstehen. Methoden des Rechnens mit Vektoren, Matrizen, Folgen und Reihen einüben und durchführen. Rechenverfahren der Differential- und Integralrechnung benutzen und in ausgewählten Problemstellungen anwenden. Differenzen- und Differentialgleichungen verstehen und Lösungsmethoden durchführen.
Schlüsselqualifikation	Fähigkeit zu abstraktem mathematischen Denken und Schließen, Fähigkeit zu komplexem Denken
Lehrinhalte	Lineare Algebra, Folgen und Reihen, Differential- und Integralrechnung, Differenzengleichungen, Differentialgleichungen
Literatur	<ul style="list-style-type: none">• G. Teschl, S. Teschl: Mathematik für Informatiker (Springer)• K. Meyberg, P. Vachenauer: Höhere Mathematik (Springer)
Leistungsnachweis	Schriftliche Prüfung (90 min.)

Theoretische Informatik

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Theoretische Grundlagen
Modul	Theoretische Informatik
Modulverantwortliche	Alexander Kröner, Thomas Fuhr
Dozent(en)	Thomas Fuhr, Alexander Kröner
Vorkenntnisse	Schulwissen, Programmieren I
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit 115 Stunden Vor- und Nachbereitung des Lehrstoffs, Lösung der Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung mit 2 SWS Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Schaffen von Grundlagen
Lernziel	Verständnis grundlegender Konzepte formaler Sprachen, der Automatentheorie sowie formaler Grammatiken und ihrer Zusammenhänge. Einordnung formaler Sprachen in die Chomsky-Hierarchie. Fähigkeit zur Abgrenzung regulärer und nicht-regulärer Sprachen. Analysieren von Automaten und Chomsky-Grammatiken. Formulieren von Grammatiken, regulären Ausdrücken und Automaten für formale Sprachen. Anwenden von Transformationen zur Überführung dieser Repräsentationen formaler Sprachen in äquivalente Modelle. Kenntnis des Berechenbarkeitsbegriffs und ausgewählter Entscheidbarkeitsprobleme.
Schlüsselqualifikation	Fähigkeit zu abstraktem mathematischen Denken und Schließen, Fähigkeit zur Bildung und zum Verständnis präziser formaler Modelle
Lehrinhalte	Endliche Automaten und formale Sprachen, Chomsky-Grammatiken und -Hierarchie, Kellerautomaten, Turingautomaten, Determinismus vs. Nichtdeterminismus, Berechenbarkeit und Entscheidbarkeit
Literatur	<ul style="list-style-type: none">• Dirk. W. Hoffmann: Theoretische Informatik, 3. Auflage, 2015, Carl Hanser Verlag.• G. Vossen, K.-U. Witt: Grundkurs theoretische Informatik: Eine anwendungsbezogene Einführung, 6. Auflage. Braunschweig, 2016, Springer Vieweg.• U. Hedstück: Einführung in die theoretische Informatik: Formale Sprachen und Automatentheorie, 5. Auflage. 2012, Oldenbourg Wissenschaftsverlag.• J.E.Hopcroft & R. Motwani & J.D.Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie; 3. Auflage, 2011, Pearson Studium.• H. Knebl: Informationsverarbeitende Systeme. Nürnberger Hochschulschriften für Technik und Wirtschaft. Nürnberg, 2003, Nano-Verlag.

Programmieren I

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Praktische Informatik
Modul	Programmieren I
Modulverantwortliche	Uwe Wienkop, Friedhelm Stappert, Jens Albrecht
Dozent(en)	Jens Albrecht, Ilse Hartmann, Friedhelm Stappert, Uwe Wienkop
Vorkenntnisse	Schulwissen
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit, 115 Stunden Vor- und Nachbereitung und Einübung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung mit 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Nahezu alle Berufsfelder für Absolventen und damit sehr viele Module der Studienprogramme der Studiengänge Informatik, Medieninformatik und Wirtschaftsinformatik haben mit der Anwendungsentwicklung von Software zu tun. Das Modul Programmieren I schafft dafür Grundlagen.
Lernziel	Nach dem erfolgreich Abschluss des Moduls Programmieren I sind die Studierenden in der Lage <ul style="list-style-type: none">• grundlegende Kontroll- und Datenstrukturen einer Programmiersprache sicher einzusetzen• einfache algorithmische Probleme zu analysieren und Lösungen dafür zu implementieren• wesentliche Konzepte objektorientierter Programmierung zu erläutern und anzuwenden• Elemente von Programmiersprachen wie Syntax, Namensbindung, Typsystem, Speicherstrukturen, Funktionsaufrufe und Parameterübergabe zu erkennen und zu erklären
Schlüsselqualifikation	<ul style="list-style-type: none">• Fähigkeit zu komplexem Denken• Ausdauer bei Problemlösungen
Lehrinhalte	<ul style="list-style-type: none">• Begriffe zur Datenverarbeitung, Syntax und Semantik der Sprachelemente: Ablaufstrukturen, Datenstrukturen, Objekte, Module, Iteration und Rekursion, Zeiger.• Entwicklungsmethoden: Entwicklungsumgebung, Entwicklung und Darstellung von Daten- und Ablaufstrukturen, strukturierter Entwurf und Implementierung, Dokumentation, Test.
Literatur	<ul style="list-style-type: none">• Kühnel, Andreas, <i>Visual C# 2012</i>, Rheinwerk Computing• Doberenz, Walter, <i>Visual C# 2015</i>, Hanser Verlag

- Hanspeter Mössenböck: *Kompaktkurs C# 6.0*, dpunkt Verlag
- Skript.

Leistungsnachweis	Schriftliche Prüfung (90 min)
Hilfsmittel	6 selbst erstellte Seiten
Zulassungsvoraussetzung	Erfolgreiche Teilnahme am Praktikum.

Programmieren II

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Praktische Informatik
Modul	Programmieren II
Modulverantwortliche	Uwe Wienkop, Timo Götzelmann, Bartosz von Rymon Lipinski
Dozent(en)	Timo Götzelmann, Ilse Hartmann, Alexander Schneider, Bartosz von Rymon Lipinski
Vorkenntnisse	Programmieren I
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit, 115 Stunden Vor- und Nachbereitung und Einübung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung mit 2 SWS Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Nahezu alle Berufsfelder für Absolventen und damit sehr viele Module der Studienprogramme der Studiengänge Informatik, Medieninformatik und Wirtschaftsinformatik haben mit der Anwendungsentwicklung von Software zu tun. Das Modul Programmieren II führt die in Programmieren I gelegten Grundlagen um i.w. objektorientierte Aspekte fort.
Lernziel	Vertiefung der Fähigkeiten, die in Programmieren I erworben wurden.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken Ausdauer bei Problemlösungen
Lehrinhalte	Fortsetzung der Lehrinhalte von Programmieren I: dynamische Datenstrukturen, insbesondere verkettete Listen, Operatoren, Nutzung von Klassenbibliotheken, Ausnahmen und ihre Behandlung.
Literatur	<ul style="list-style-type: none">• Kühnel, Andreas, <i>Visual C# 2012</i>, Rheinwerk Computing• Doberenz, Walter, <i>Visual C# 2015</i>, Hanser Verlag• Hanspeter Mössenböck: <i>Kompaktkurs C# 6.0</i>, dpunkt Verlag• Skript.
Leistungsnachweis	Schriftliche Prüfung (90 min)

Gestaltungs- und Medienlehre I

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Digitale Medien
Modul	Gestaltungs- und Medienlehre I
Modulverantwortliche	Malte Burdekat , Leo Glomann
Dozent(en)	Malte Burdekat , Leo Glomann
Vorkenntnisse	Schulwissen
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz und 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS seminaristischer Unterricht
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Grundlage für die konzeptionell geplante und medienintegrierte Entwurfsarbeit ist das Verstehen des gesamten Gestaltungsprozesses als Auseinandersetzung mit inhaltlichen Vorgaben, klar definierten Aussagen und der den Zielgruppen entsprechenden Ausgestaltung mit der jeweils angemessenen Medienwahl.
Lernziel	Teil 1: <ul style="list-style-type: none">• Kenntnisse über Grundlagen von Interaktionsdesign• Basiskenntnisse der Mensch-Technik-Kommunikation aus arbeitspsychologischer, ergonomischer und gestalterischer Perspektive• Grundlegende Fähigkeiten, diese in einem gegebenen Kontext anzuwenden (insbes. Evaluation und Konzeptualisierung)• Steigerung des visuellen Wahrnehmungsvermögens und Sensibilisierung für die Grundproblematik insbesondere im Darstellungsmedium Layout / Typografie.• Fähigkeit zur Beurteilung ausgewählter Gestaltungsprozesse hinsichtlich ihrer inhaltlichen Klarheit, der eindeutigen Verständlichkeit und der passenden Gestaltung. Dabei richtet sich der Blick auf die wahrnehmungsbezogene, kommunikative und ästhetische Wirkungsebene von Gestaltung. Teil2: <ul style="list-style-type: none">• Einordnen der Fotografie in den Kontext der technischen Kompetenzen und künstlerischen Entwicklungen des 19. und 20. Jahrhunderts.• Erkennen des Unterschieds zwischen Beschreiben und Bewerten bei fotografischen Bilderzeugnissen.• Erstellen von Begründungen zu Werturteilen über fotografische Bilder.
Schlüsselqualifikation	<ul style="list-style-type: none">• Erkennen der unterschiedlichen Prozessebenen von Gestaltung• Fähigkeit, medial aufbereitete Gestaltungsprodukte (Print, Film, Web) mehrschichtig zu analysieren und zu verstehen• Reflektion formaler Gestaltungslösungen

Lehrinhalte

Teil 1:

- Vorlesung: Basiskonzepte der Mensch-Technik-Kommunikation aus arbeitspsychologischer, ergonomischer und gestalterischer Perspektive
- Seminar: Übungen zu den Prozessebenen Analyse, Recherche, Ideenfindung, Konzeptualisierung, Darstellung, Evaluation.

Teil 2:

- Vorlesung mit Seminar, Referate zur Geschichte der Fotografie
- Historischer Abriss über die gestalterische und technische Entwicklung der Fotografie, die gleichzeitig als Übung zur Entwicklung von Beurteilungskriterien und Bildanalyse fotografischer Bilder dient. Mit einer Einführung zu Grundlagen der visuellen Kommunikation/Bildwissenschaft.
- Praktische Übungen

Literatur

- Stapelkamp, T. (2010): Interaction- und Interfacedesign. Heidelberg: Springer
- Wandmacher, J. (1993). Software-Ergonomie. Berlin: de Gruyter
- Sarodnick, F., Brau, H. (2011): Methoden der Usability-Evaluation. Bern: Hans Huber
- Michael Frizot: Neue Geschichte der Fotografie. Köln: Könemann, 1988
- Einzelbiographien und Autorenbildbände

Leistungsnachweis

Praktische Studienarbeit/Übungen

Gestaltungs- und Medienlehre II

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Digitale Medien
Modul	Gestaltungs- und Medienlehre II
Modulverantwortliche	Peter Krüll, Fiona Pleasance
Dozent(en)	Hendrik Leyendecker, Fiona Pleasance
Vorkenntnisse	Gestaltungs- und Medienlehre I
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz und 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS Vorlesung / seminaristischer Unterricht
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Grundlage für die konzeptionell geplante und medienintegrierte Entwurfsarbeit ist das Verstehen des gesamten Gestaltungsprozesses als Auseinandersetzung mit inhaltlichen Vorgaben, klar definierten Aussagen und der den Zielgruppen entsprechenden Ausgestaltung mit der jeweils angemessenen Medienwahl.
Lernziel	<ul style="list-style-type: none">• Bedeutung und Stellenwert im zeitgemäßen Kommunikations Design mit der Wechselwirkung und Refektion von Form und Inhalt einer gestalterischen und kommunikativen Botschaft. (Teil 1).• Verstehen von filmtechnischen und filmhistorischen Gestaltungsmitteln.• Anwenden der Werkzeuge der Filmanalyse. Die Fähigkeit, einzelne Eigenschaften des Filmemachens zu erkennen und zu deuten.• Das Analysieren von Filmsequenzen, um die gestalterischen Mitteln von bewegten Bildern aufzuschlüsseln. (Teil 2).
Schlüsselqualifikation	<ul style="list-style-type: none">• Erkenntnisse im Rahmen der Design- und Kommunikations- Projekte – unter Berücksichtigung von Bild und Schrift. (Teil 1).• Fähigkeit, medial aufbereitete Gestaltungsprodukte (bewegte Bilder) mehrschichtig zu analysieren und zu verstehen (Teil 2).• Reflektion formaler Gestaltungslösungen.
Lehrinhalte	<ul style="list-style-type: none">• Vorlesung und Übung. Definition und Grundlagen mit Sichtung und Beurteilung beispielhafter Arbeiten aus Bereichen des Kommunikations Designs. Entwurf einer Semester-Teamarbeit mit gemeinsamen Besprechungen und Korrekturen bzgl. gestalterischer und inhaltlicher Qualita?t. (Teil 1).• Filmgestalterische Grundlagen auf der Basis filmhistorischer und filmanalytischer Kenntnisse. Die praktische Anwendung des Erlernten. (Teil 2).
Literatur	<ul style="list-style-type: none">• Film Verstehen: Kunst, Technik, Sprache, Geschichte und Theorie des Films und der Neuen Medien, James Monaco, Rowohlt Taschenbuch Verlag 2009-• Film Art: An Introduction, David Bordwell & Kristin Thompson, McGraw-Hill 2017 (Teil 2).

Grundlagen der Medieninformatik

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Digitale Medien
Modul	Grundlagen der Medieninformatik
Modulverantwortliche	Florian Gallwitz
Dozent(en)	Florian Gallwitz
Vorkenntnisse	Schulwissen
Arbeitsaufwand	120 Stunden, davon: 65 Stunden Präsenz 55 Stunden Vor- und Nachbereitung des Lehrstoffs, Prüfungsvorbereitung und Prüfung
Leistungspunkte	4
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS Vorlesung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung von Grundlagen der Codierung, Darstellung und Bearbeitung von Digitalen Medien.
Lernziel	Kenntnis der wichtigsten Technologien, Verfahren und Vorgehensweisen bei der Erzeugung und Bearbeitung multimedialer Anwendungen, Fähigkeit zur Einordnung und zum Umgang mit verschiedenen Medienformaten.
Schlüsselqualifikation	Logisches Denken, interdisziplinäres Arbeiten, Transferfähigkeit
Lehrinhalte	<ul style="list-style-type: none">• Mediale Komponenten und deren Standards, Datenformate und Datenkompression• Grundlagen der Audio-, Foto- und Videotechnik, Farbsysteme• Erstellung und Bearbeitung von medialen Komponenten und Multimedia-Anwendungen• Skriptsprachen für Multimediaanwendungen• Multimedia im Netz• Auszeichnungssprachen (XML, XHTML, ...)• Gerätetechnik• Interaktion• Virtuelle Realität• Beispiele multimedialer Produktionen
Literatur	<ul style="list-style-type: none">• Butz, A., Hussmann, H., Malaka, R.: Medieninformatik - Eine Einführung, Pearson Studium, 2009• Bruns, K., Meyer-Wegener, K.: Taschenbuch der Medieninformatik, Fachbuchverlag Leipzig, 2005• Henning P.A., Taschenbuch Multimedia, Hanser Verlag, 2003
Leistungsnachweis	Schriftliche Prüfung (90 min)

Englisch I/II

Studiengang	Bachelor Medieninformatik 1. Studienabschnitt Fremdsprache
Modul	Englisch I/II
Modulverantwortliche	Thomas Huke
Dozent(en)	William Coleman
Vorkenntnisse	Schulwissen
Arbeitsaufwand	120 Stunden, davon: 65 Stunden Präsenz 55 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	4
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS seminaristischer Unterricht
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Englisch
Beitrag zu den Zielen des Studiums	Die Studierenden identifizieren und erklären technische Originalliteratur und wenden die technische Sprache kontextgerecht in Wort und Schrift an.
Lernziel	Die Studierenden erwerben die Fähigkeit, gesprochenes und geschriebenes Englisch mit allgemeinsprachlichen und fachlichen Inhalten zu verstehen, sowie sich in der Fremdsprache mündlich und schriftlich korrekt auszudrücken.
Schlüsselqualifikation	Fremdsprachenkompetenz
Lehrinhalte	Fachbezogene Texte; allgemeine sowie fachbezogene Korrespondenz und Konversation; Erweiterung und Festigung der Vokabelkenntnisse; Umgang mit Hilfsmitteln
Literatur	<ul style="list-style-type: none">• Bruckenmaier, Stephan. 2008. <i>Grammatik kurz und bündig – Englisch</i>. Stuttgart: Klett Verlag (Pons).• Clark, David. 1997. <i>Englisch Grammatik</i>. Pocket Teacher. Berlin: Cornelsen Verlag.• Hoffmann, Hans & Marion Hoffmann. 2006. <i>Kurzgrammatik Englisch. Zum Nachschlagen und Üben</i>. München: Max Hueber.• Jacob, Rainer. 2008. <i>Kurzgrammatik. Kompakt-Wissen Englisch</i>. Freising: Stark Verlag.• Walther, Lutz. 2008. <i>Kurzgrammatik Englisch</i>. Berlin: Langenscheidt.• Weis, Erich. 1977. <i>Grund- und Aufbauwortschatz Englisch</i>. Stuttgart: Klett. (Nachdruck – 1986)
Leistungsnachweis	Schriftliche Prüfung (90 min)

Algorithmen und Datenstrukturen

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Theoretische Grundlagen
Modul	Algorithmen und Datenstrukturen
Modulverantwortliche	Thomas Fuhr, Peter Trommler
Dozent(en)	Thomas Fuhr
Vorkenntnisse	Inhalte der Lehrveranstaltungen des ersten Studienjahres, insbesondere Mathematik, Grundlagen der Informatik, Theoretische Informatik und Programmieren
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Lösung der Praktikumsaufgaben und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung/Praktikum
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Veranstaltung mit Grundlagen- und Querschnittcharakter. Behandelte Themen sind grundlegend für das Verständnis der Realisierung von Standardsoftware (z.B. Datenbanken, Systemsoftware) sowie für die Entwicklung adäquater Softwarelösungen aller Art.
Lernziel	Grundlegende Datenstrukturen unter Einbeziehung externer Speichermedien kennen; zugehörige Algorithmen und ihre Zusammenhänge verstehen sowie diese korrekt für konkrete Beispiele ausführen können; die Aspekte, Korrektheit, Komplexität und Effizienz von Algorithmen sowie übliche Entwurfsprinzipien kennen; einfache Algorithmen analysieren und ihren Aufwand mathematisch beschreiben und kategorisieren bzw. deren Korrektheit beweisen können; Algorithmen hinsichtlich ihres Aufwands einordnen und vergleichend bewerten können.
Schlüsselqualifikation	Fähigkeit zu komplexem, abstraktem mathematischen Denken und Schließen, selbständiges Arbeiten, Analyse und Klassifikation von Problemen, kreatives Problemlösen, Ausdauer bei Problemlösungen.
Lehrinhalte	Korrektheit von Algorithmen, Komplexität und Effizienzbetrachtungen, Entwurfsprinzipien von Algorithmen. Grundlegende Algorithmen aus den Bereichen Sortieren und Suchen, Suchbäume, Hashverfahren, Organisation von Wörterbüchern.
Literatur	<ul style="list-style-type: none">• T.H.Cormen, C.E.Leiserson, R.L.Rivest, C.Stein: Algorithmen – Eine Einführung, 4. Aufl., 2013, De Gruyter Oldenbourg• T. Ottmann, P. Widmayer: Algorithmen und Datenstrukturen, 5. Aufl. 2012, Spektrum Akademischer Verlag.• V. Heun: Grundlegende Algorithmen: Einführung in den Entwurf und die Analyse effizienter Algorithmen, 2.Aufl., Braunschweig, Wiesbaden, 2003, Vieweg.

- D.E. Knuth: The Art of Computer Programming, Volume 1+3: Fundamental Algorithms + Searching and Sorting. Reading, MA, 1998, Addison-Wesley Publishing Company.
- H. Knebl: Algorithmen und Datenstrukturen. Nürnberg, 2005, Skript.

Leistungsnachweis

Schriftliche Prüfung (90 min). Zulassungsvoraussetzung für die Teilnahme an der schriftlichen Prüfung ist die erfolgreiche Teilnahme am Praktikum.

Mathematik III (Geometrie, Transformationen)

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Theoretische Grundlagen
Modul	Mathematik III (Geometrie, Transformationen)
Modulverantwortliche	Hans Delfs
Dozent(en)	Edgar Wermuth
Vorkenntnisse	Mathematik I und Mathematik II
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Prüfungsvorbereitung, Prüfung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Modellierung und Darstellung dreidimensionaler Objekte erfordern Kenntnisse in analytischer Geometrie.
Lernziel	Wesentliche mathematische Grundlagen der Computergraphik und der Medienverarbeitung kennen und verstehen; Geometrische Transformationen im affinen und projektiven Raum definieren und anwenden können; die Transformation von Signalen aus dem Zeit-/Orts- in den Frequenzbereich kennen und verstehen; Algorithmen der Bildsynthese und der Bildanalyse auf mathematischer Grundlage formulieren können.
Schlüsselqualifikation	Fähigkeit des guten räumlichen Vorstellungsvermögens. Fähigkeit zu abstraktem mathematischen Denken und Schließen.
Lehrinhalte	Analytische Geometrie: Affiner Raum und affine Transformationen, Koordinatensysteme und Basiswechsel, Orientierung, projektiver Raum und homogene Koordinaten, orthogonale und perspektivische Projektion, Drehungen und Quaternionen, Skalierung. Orts/Zeit- und Frequenzbereich, Fourier-Transformation, diskrete Fourier-Transformation, Kosinustransformation und Datenkompression.
Literatur	R. Brigola: Fourieranalysis, Distributionen und Anwendungen. Vieweg, 1996. G. Fischer: Analytische Geometrie. Eine Einführung für Studienanfänger. 7. Auflage. Vieweg, 2001. E. Kreyszig: Advanced Engineering Mathematics. 9th edition. Wiley, 2006. G. Teschl, S. Teschl: Mathematik für Informatiker 1 und 2, Springer-Verlag, 2008/2006. J. Vince: Quaternions for Computer Graphics. Springer, 2011.
Leistungsnachweis	Schriftliche Prüfung (90 min)

Mathematik IV (Statistik)

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Theoretische Grundlagen
Modul	Mathematik IV (Statistik)
Modulverantwortliche	Harald Stieber, Hans Delfs, Alexander Hufnagel
Dozent(en)	Hans Delfs, Alexander Hufnagel, Edgar Wermuth
Vorkenntnisse	Mathematik I und II
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 65 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung 20 Stunden für Prüfung und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS Vorlesung
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Wahrscheinlichkeitsrechnung und Statistik sind wichtige Grundlagen für verschiedene Gebiete der Informatik.
Lernziel	Grundlegende Methoden der Wahrscheinlichkeitsrechnung und der beschreibenden und schließenden Statistik verstehen und eigenständig anwenden, die Voraussetzungen ihrer Anwendung prüfen, statistische Modelle vergleichen und zur Anwendungssituation passende auswählen, Ergebnisse sinnvoll interpretieren können.
Schlüsselqualifikation	Fähigkeit zu abstraktem mathematischen Denken und Schließen
Lehrinhalte	Deskriptive Statistik: Lage- und Streuungsmaße, Korrelation und Regression; Elemente der Wahrscheinlichkeitsrechnung: Zufallsvariable, Bedingte Wahrscheinlichkeiten, Wahrscheinlichkeitsverteilungen, Poisson-Prozesse; Induktive Statistik: Punktschätzung, Intervallschätzung, Konfidenzintervalle, Testen von Hypothesen.
Literatur	G. Teschl, S. Teschl: Mathematik für Informatiker, Band 2. 3. Auflage. Berlin, Heidelberg: Springer 2014. J. Lehn, H. Wegmann: Einführung in die Statistik. 5. Auflage. Wiesbaden: Teubner 2006.
Leistungsnachweis	Schriftliche Prüfung (90 min)

Betriebssysteme und Rechnerarchitektur

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Betriebssysteme und Rechnerarchitektur
Modulverantwortliche	Axel Hein, Michael Zapf
Dozent(en)	Axel Hein, Michael Zapf
Vorkenntnisse	Grundlagen der Informatik, Theoretische Informatik, Programmieren.
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Lösung von Übungsaufgaben, Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung.
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Das Betriebssystem und die Rechnerarchitektur bilden den Rahmen, in dem Anwendungen auszuführen sind. Ein grundlegendes Verständnis der Vorgänge im Inneren des Computers ist unerlässlich, um beurteilen zu können, ob ein Computersystem den hohen Anforderungen entsprechen kann, die insbesondere von Anwendungen aus der Medieninformatik gestellt werden, wie etwa hohen Datentransferraten, schnellen Antwortzeiten und der Verarbeitung großer Datenmengen.
Lernziel	Kenntnis von Aufbau und Komponenten zentraler und dezentraler Rechnerarchitekturen; Fähigkeit, die Auswirkungen von Systementscheidungen und Komponentenauswahl auf die Programmierung und die Leistungsfähigkeit von Anwendungen zu bewerten. Analyse von Entwurfsdetails gegebener Systemarchitekturen bis hin zur selbständigen Gestaltung von einfachen Computersystemen mit vorgegebenen Anforderungen. Verständnis der Konzepte von Betriebssystemen; exemplarisch verdeutlicht anhand konkreter Fallstudien. Bewertung der Eigenschaften von Betriebssystemen in Bezug auf Einsatzbereiche und Verhalten unter Last für bestimmte Nutzungsprofile.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Analyse und Klassifikation von Problemen; technisches Verständnis und Lösungskompetenz
Lehrinhalte	<ul style="list-style-type: none">• Aufbau von Rechensystemen• Wichtige Rechnerstrukturen, einschließlich Prozessoren, Hauptspeicher, sekundäre Speichertypen, Cachespeicher• Bewertung der Leistungsfähigkeit von Rechnersystemen• Aufgaben und Aufbau von Betriebssystemen• Ressourcenverwaltung (wie Speicher, Prozesse, Scheduling, Dateisystem)• Kommunikations- und Hilfsfunktionen
Literatur	<ul style="list-style-type: none">• Andrew S. Tanenbaum: Modern Operating Systems. 4th edition, Pearson/Prentice Hall, 2014.

- Eduard Glatz: Betriebssysteme, 3. Auflage, dpunkt-Verlag, 2015.
- Andrew S. Tanenbaum: Structured Computer Architecture, 6th edition, Pearson/Prentice Hall, 2012
- John L. Hennessy, David A. Patterson: Computer Organization and Design – The Hardware/Software Interface, 5th edition, Morgan Kaufmann Publishers, 2013.
- William Stallings: Computer Organization and Architecture, 9th edition, Addison-Wesley, 2012.
- Vorlesungsunterlagen

Leistungsnachweis

Schriftliche Prüfung (90 min)

Hilfsmittel

5(10), 7

Datenbanken

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Datenbanken
Modulverantwortliche	Jens Albrecht, Alfred Holl, Patricia Brockmann
Dozent(en)	Jens Albrecht, Patricia Brockmann, Alfred Holl
Vorkenntnisse	Lehrveranstaltungen des ersten Studienjahres, insbesondere Grundlagen der Informatik, Programmieren.
Arbeitsaufwand	180 Stunden, davon: 90 Stunden Präsenzzeit, 90 Stunden für Vor- und Nachbereitung des Lehrstoffs, Bearbeitung der Übungs- und Praktikumsaufgaben und Prüfungsvorbereitung
Leistungspunkte	6
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung, 2 SWS Übung mit Praktikum
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	In der Anwendungsentwicklung und im Systemmanagement werden gute Kenntnisse über die Funktionsweise und Leistungsfähigkeit von Datenbanksystemen benötigt. Methoden des systematischen Entwurfs von Datenmodellen für Datenbanken zu beherrschen, Datenbanken implementieren und den Zugriff auf die gespeicherten Daten aus Anwendungssystemen realisieren zu können, sind wesentliche Fähigkeiten eines Informatikers.
Lernziel	Nach dem erfolgreichen Abschluss des Moduls Datenbanken sind die Studierenden in der Lage <ul style="list-style-type: none">• die Bedeutung von Datenbanken in Software-Systemen zu bewerten• verschiedene Datenbankmodelle voneinander abzugrenzen• die Grundlagen relationaler Datenbanksysteme zu erklären• konzeptionelle Datenmodelle zu erstellen und in normalisierte relationale Datenbankschemata zu überführen• komplexe SQL-Anfragen und SQL-Änderungsoperationen zu formulieren• die Qualität eines entworfenen Datenbankschemas zu bewerten• die Eignung eines relationalen Datenbanksystems für ein gegebenes Problem einzuschätzen
Schlüsselqualifikation	Fähigkeit zu komplexem, abstraktem mathematischen Denken und Schließen, Analyse und Klassifikation von Problemen, kreatives Problemlösen.
Lehrinhalte	Aufbau und Funktionen eines Datenbanksystems; objektorientiertes und relationales Datenmodell, objektrelationale Abbildung; relationale, objektrelationale, föderative, verteilte Datenbanksysteme; Datenmodellierung und Datenbankentwurf, Referenzmodelle, Unternehmensdatenmodelle, Datenintegration; Datenbankabfragesprachen: Datendefinition und –manipulation mit der Structured Query Language SQL; Schnittstellen für die Anwendungsprogrammierung; Transaktionsmanagement, Concurrency und Recovery. Einführung in NoSQL-

Datenbanken und Big Data.

Literatur

- Kemper, A.; Eickler, A.: Datenbanksysteme - Eine Einführung, De Gruyter Studium, 2015
- Sattler, K.; Heuer, A.: Datenbanken – Konzepte und Sprachen. mitp, 2013
- Saake, G.; Heuer, K.; A. Sattler: Datenbanken – Implementierungstechniken. mitp, 2011
- Elmasri, R. , Navathe, S., "Grundlagen von Datenbanksystemen", Bachelorausgabe, Pearson 2010

Leistungsnachweis

Schriftliche Prüfung (90 min).

Hilfsmittel

10 selbst erstellte Seiten

Zulassungsvoraussetzung

Erfolgreiche Teilnahme am Praktikum.

Rechnerkommunikation

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Rechnerkommunikation
Modulverantwortliche	Peter Trommler, Michael Zapf
Dozent(en)	Peter Trommler, Michael Zapf
Vorkenntnisse	Grundlagen der Informatik, Theoretische Informatik, Mathematik I und II
Arbeitsaufwand	120 Stunden, davon: 65 Stunden Präsenz 55 Stunden Vor- und Nachbereitung des Lehrstoffs, Lösung der Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	4
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Vernetzung von Rechnern ist eine grundlegende Anforderung an verteilte Dienste, die gerade in der Medieninformatik bedeutsam sind. Audio- und Videoströme werden heutzutage wie selbstverständlich genutzt, doch müssen die Daten von einem möglicherweise weit entfernten Rechner zunächst durch das Internet transportiert werden, bevor sie auf dem heimischen Rechner oder auf dem Mobilgerät präsentiert werden können. In dieser Veranstaltung wird die Funktionsweise von Kommunikationsnetzen, insbesondere dem Internet, detailliert besprochen.
Lernziel	Kenntnis der Architektur von Kommunikationssystemen; Verständnis der Architekturkonzepte; Verständnis und Bewertung der Anforderungen an Datentransporttechniken; Kenntnis der typischen Kommunikationsprotokolle im Internet und Verständnis deren typischen Eigenschaften; Beurteilung der Einsetzbarkeit von Techniken für gegebene Anwendungsszenarien; Realisierung eigener Lösungen zur Kommunikation mit entfernten Diensten unter Nutzung englischsprachiger Spezifikationstexte.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Analyse und Klassifikation von Problemen. Fähigkeit zum Umgang mit englischsprachigen Standardtexten. Technisches Verständnis und Kompetenz.
Lehrinhalte	<ul style="list-style-type: none">• Request for Comments: Bedeutung, Verständnis, Umsetzung• Schichtenmodell (Referenzmodelle)• Kommunikation in lokalen Netzen• Routing mit den Internetprotokollen v4 und v6• Broadcast und Multicast• Transportschichtprotokolle• Hierarchische Namensräume und zugehörige Dienste (insbesondere DNS)• Typische Anwendungen im Internet: E-Mail, World Wide Web• Audio- und Videostreaming

Literatur	<ul style="list-style-type: none"> • Peterson L., Davie B., Computer Networks: A Systems Approach, 4th Edition, 2008, Morgan Kaufmann Publishers. (deutsche Übersetzung "Computernetze" erschienen im dPunkt Verlag, Heidelberg, 2008) • Andrew S. Tanenbaum, Computer Networks, 4th Edition, Upper Saddle River, NJ, 2002, Prentice Hall PTR (auch deutsche Übersetzung erschienen bei Pearson Studium, 2003) • Roth J., Prüfungstrainer Rechnernetze, Vieweg+Teubner, 2010
Leistungsnachweis	<p>Schriftliche Prüfung (90 min) Voraussetzungen: erfolgreiche Teilnahme am Praktikum (nur bei Studienbeginn vor WS13/14)</p>
Hilfsmittel	5(10), 7

Software Engineering

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Software Engineering
Modulverantwortliche	Ramin Tavakoli Kolagari, Bartosz von Rymon Lipinski, Alfred Holl
Dozent(en)	Alfred Holl, Bartosz von Rymon Lipinski
Vorkenntnisse	Programmieren
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Bearbeitung der Übungs- und Praktikumsaufgaben und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung und Praktikum
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	In den Berufsfeldern Anwendungsentwicklung und Entwicklung systemnaher Software gehört das Software Engineering zum unverzichtbaren Handwerkszeug.
Lernziel	Bei Abschluss des Lernprozesses wird der erfolgreiche Studierende in der Lage sein, Ziele, Methoden, Techniken und Verfahren des Software Engineering zu kennen und diese für konkrete Beispiele anwenden zu können. Die Studierenden werden Prozesse des Software Engineering verstehen und ausgewählte Techniken und Methoden für verschiedene Entwicklungsphasen auswählen können. Im Rahmen des begleitenden Praktikums werden sie teambasiert eine Beispielanwendung bzw. betriebliches Informationssystem im Rahmen eines vorgegeben Projektgerüsts prototypisch entwickeln können.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Analyse und Klassifikation von Problemen.
Lehrinhalte	<ul style="list-style-type: none">• Probleme der industriellen Softwareerstellung;• Phasenmodelle;• Methoden zur Anforderungsspezifikation; Entwurfsmethoden;• Methoden zur Systemkonstruktion; Systemintegration und Test; Software-Ergonomie;• Qualitätssicherung; Softwaremetriken; Projektmanagement; DV-gestützte Entwicklungsumgebungen;• Aufwandsschätzungen; Wirtschaftlichkeitsbetrachtungen.
Literatur	<ul style="list-style-type: none">• C. Rupp, S. Queins, UML2 glasklar, 2012, Hanser.• G. Pomberger, W. Pree, Software Engineering, München, 2004, Hanser.• I. Sommerville, Software Engineering, München, 2012, Pearson Studium.
Leistungsnachweis	Schriftliche Prüfung (90 min). Erfolgreiche Teilnahme am Praktikum ist Voraussetzung für die Zulassung zur Prüfung.
Hilfsmittel	

Softwarearchitektur

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Softwarearchitektur
Modulverantwortliche	Korbinian Riedhammer, Ramin Tavakoli Kolagari
Dozent(en)	Korbinian Riedhammer
Vorkenntnisse	Praktisches Studiensemester
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit Übungen
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Erfahrungsgemäß gehören der Entwurf und die Architektur großer Softwaresysteme zu den schwierigsten Aufgaben der praktischen Informatik. Diese Lehrveranstaltung bereitet die Studierenden darauf vor, indem sie architektonische Alternativen und Kriterien für ihre Bewertung kennen lernen.
Lernziel	Bei Abschluss des Lernprozesses wird der erfolgreiche Studierende in der Lage sein, Konzepte guter Softwarearchitektur, insbesondere Clean Code, Programmieridiome, Entwurfsmuster, Architekturmuster und Komponenten zu kennen und diese für konkrete Szenarien anwenden, die unterschiedlichen Prinzipien der Softwarearchitektur für den Entwurf eines Systems auswählen und kleine Beispielanwendung unter Vermeidung bekannter Architekturprobleme entwickeln zu können.
Schlüsselqualifikation	Fähigkeit zum Verständnis großer komplexer Systeme; Analyse und Klassifikation von Problemen; kreatives Problemlösen.
Lehrinhalte	Dieser Kurs vermittelt grundlegende Techniken zum Entwurf, zur Beschreibung und zur Implementierung von großen, modularen Softwaresystemen. Zentraler Begriff der Vorlesung sind Patterns, welche als Grundprinzipien und Bausteine einer Softwarearchitektur verstanden werden können. Der Kurs führt Patterns auf verschiedenen Ebenen eines Softwaresystems ein: Auf Implementierungsebene (Programming Idioms), auf Ebene verschiedener Softwaremodule, Klassen und deren Zusammenspiel (Design Patterns), und schließlich als prototypische Struktur für ganze Anwendungen (Architectural Patterns). Muster werden anhand von Programmbeispielen veranschaulicht. Es werden mehrere Programmierübungen im Labor bearbeitet.
Literatur	<ul style="list-style-type: none">• Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides: "Design Patterns, Elements of Reusable Object-Oriented Software", Addison Wesley, 1994.• Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: "Pattern-Oriented Software Architecture", Wiley, 1996.

- Alan Shalloway, James R. Trott: "Design Patterns Explained: A New Perspective on Object-Oriented Design", Addison Wesley, 2005.
- Matthias Geirhos: "Entwurfsmuster: Das umfassende Handbuch", Rheinwerk Computing, 2015.
- Mahboub Gharbi, Arne Koschel, Andreas Rausch, Gernot Starke: "Basiswissen für Softwarearchitekten", dpunkt.verlag, 2013.

Leistungsnachweis

Schriftliche Prüfung (90 Minuten)

Web-Programmierung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktische Informatik
Modul	Web-Programmierung
Modulverantwortliche	Matthias Teßmann
Dozent(en)	Matthias Teßmann, Thomas Ulrich
Vorkenntnisse	Programmieren I und II, Software Engineering, Grundlagen der Informatik
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbereiten des Lehrstoffs sowie Übungsaufgaben/Praktikum
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übungen und studienbegleitendes Praktikum
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Nahezu alle Berufsfelder für Absolventen und damit sehr viele Module der Studienprogramme der Studiengänge Informatik, Medieninformatik und Wirtschaftsinformatik haben mit der Anwendungsentwicklung von Software zu tun. Das Modul Web-Programmierung führt die in Programmieren I und II gelegten Grundlagen mit einem besonderem Fokus auf Webanwendungen fort.
Lernziel	Durch das Modul Web-Programmierung lernen die Studierenden die wichtigsten Schlüsseltechniken des World Wide Web kennen. Dies umfasst sowohl client- als auch serverseitige Technologien, wie unter anderem HTML5, CSS3, JavaScript, HTTP, Webserver (Apache2, Nginx), PHP, REST-Schnittstellen und asynchrone Anfragen. Dadurch erlangen Sie ein Verständnis für die Zusammenhänge dieser und werden dazu befähigt, deren Vor- und Nachteile gegeneinander abzuwägen. Die vermittelten theoretischen Kenntnisse werden im Rahmen des Pflichtpraktikums durch die selbstständige Konzeption und Umsetzung einer vollständigen Client-Server-Web-Anwendung (z.B. Online-Shop, soziales Netzwerk, ...) modulbegleitend praktisch gefestigt. Dabei kommt die Lernform des problembasierten Lernens (PBL) zum Einsatz. Das Modul schließt mit der Vermittlung grundlegender Kenntnisse über die wichtigsten Sicherheitsrisiken von Web-Anwendungen.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, selbständiges Arbeiten, Fähigkeit zu komplexem Denken, Fähigkeit zum Verständnis großer komplexer Systeme, Lernen am Beispiel, Transferfähigkeit
Lehrinhalte	Die Inhalte der Lehrveranstaltung sind u.a.: <ul style="list-style-type: none">• Grundlagen<ul style="list-style-type: none">◦ HTML5, CSS3, JavaScript• Client-Server Modell / Kommunikationsprotokolle und Datenformate<ul style="list-style-type: none">◦ HTTP/HTTPS/HTTP2◦ JSON• Front-End, Back-End und "Full-Stack"-Anwendungsentwicklung

- PHP, Application-Server, node.js
- Cookies und Sessions
- Asynchrone Kommunikation
- Browser- und Server-Architekturen
- Single-Page-Anwendungen

- Der Browser als universelle Anwendungsplattform
- Web-Anwendungssicherheit (OWASP Top 10)

Literatur

- The World Wide Web Consortium (W3C), Standards and Recommendations, Online: <http://w3.org>
- Jürgen Wolf, "HTML5 und CSS3: Das umfassende Handbuch.", 2. Auflage, Rheinwerk Computing, 2016
- David Flanagan, "JavaScript: The Definitive Guide", 6th. Ed.. O'Reilly, 2011
- Douglas Crockford, "JavaScript: The Good Parts", O'Reilly, 2008
- Steve Prettyman, "Learn PHP 7", Apress, 2016
- The Open Web Application Security Project, Documents and Recommendations, Online: <http://www.owasp.org>

Leistungsnachweis

Schriftliche Prüfung (90 Minuten), erfolgreiche Teilnahme am Praktikum für die Zulassung erforderlich

Hilfsmittel

Zulassungsvoraussetzung

Interdisziplinäre Projektarbeit (9LP)

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien
Modul	Interdisziplinäre Projektarbeit (9LP)
Modulverantwortliche	Peter Trommler
Dozent(en)	Florian Gallwitz, Timo Götzelmann, Alexander Kröner, Jörg Roth, Friedhelm Stappert, Matthias Teßmann, Peter Trommler, Bartosz von Rymon Lipinski, Michael Zapf
Vorkenntnisse	Programmieren, Software Engineering, Medienkonzeption, Interaktive Systeme, Software Ergonomie, Projektmanagement
Arbeitsaufwand	270 Stunden Projektarbeit, davon: 90 Stunden Präsenz (Projektbesprechungen, gemeinsam in der Projektgruppe zu leistende Diskussion der Ziele, Planungsarbeiten, Integration von Arbeitsergebnissen, Integrationstest, Präsentation des Projektergebnisses, Vortragsveranstaltungen),
Leistungspunkte	9
Semesterwochenstunden	6
Veranstaltungstyp	Projektarbeit
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Das Projekt bereitet auf alle Berufsfelder vor, die mit der Entwicklung von medienbasierter Software verbunden sind. Interdisziplinäre Anwendungsfelder und/oder Teams fördern fachübergreifendes Verständnis und Kommunikationsfähigkeit.
Lernziel	Einblick in die Vielgestaltigkeit von Anwendungen der Medieninformatik. Fähigkeit zur Umsetzung einer Problemstellung in eine anwendergerechte Lösung. Fähigkeit in einem interdisziplinären Team zu arbeiten.
Schlüsselqualifikation	Sozialkompetenz (Kooperationsfähigkeit, Teamfähigkeit), Organisationskompetenz, Projekterfahrung, Projektmanagement, Zeitmanagement, Ausdauer bei Problemlösungen, fachübergreifendes Wissen, Interdisziplinarität
Lehrinhalte	Die Studierenden bearbeiten in bevorzugt interdisziplinären Teams von ca. 4 Mitgliedern jeweils ein Thema aus der Medieninformatik. Sie werden dabei von einem Dozenten betreut. Typischerweise ist für eine ausgewählte Problemstellung eine medienorientierte IT-Lösung in Form einer Software-Anwendung zu planen, zu entwerfen und zu implementieren, oder es sind implementierte Lösungen zu beurteilen und anzupassen. Die Projektarbeit erstreckt sich über 2 Semester. Im ersten der beiden Semester analysiert das Team die gestellte Aufgabe, entwickelt einen Projektplan und beginnt mit der Umsetzung des Projekts, beispielsweise dem Entwurf oder einem Prototypen zu einer Softwarelösung. Im zweiten Semester wird das Projekt fertiggestellt, mit einem Bericht dokumentiert und dem Betreuer und Mitstudierenden in einem begleitenden Seminar vorgestellt.
Literatur	Abhängig vom Projektthema
Leistungsnachweis	Studienarbeit, Referat

Medienverarbeitung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien Medienanalyse
Modul	Medienverarbeitung
Modulverantwortliche	Florian Gallwitz
Dozent(en)	Florian Gallwitz
Vorkenntnisse	Mathematik III
Arbeitsaufwand	210 Stunden, davon: 95 Stunden Präsenzzeit, 115 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Fähigkeiten zur spezialisierten Anwendungsentwicklung werden maßgeblich entwickelt.
Lernziel	Kenntnis der Grundlagen und Methoden der Bild-, Video- und Audioverarbeitung, Fähigkeit zur Entwicklung von Anwendungen zur Aufnahme, Verarbeitung und Analyse von Mediendaten.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen.
Lehrinhalte	Bild-, Video-, Audioverarbeitung (Erfassung und Verbesserung von Medien). Mustererkennung (automatische Klassifikation mit extrahierten Merkmalen). Bild- und Sprachverstehen (Computer Vision, rechnergestütztes Erkennen von Bildinhalten).
Leistungsnachweis	Schriftliche Prüfung (90 min)

Computergraphik

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien Mediensynthese
Modul	Computergraphik
Modulverantwortliche	Christian Schiedermeier
Dozent(en)	Christian Schiedermeier
Vorkenntnisse	Mathematik I, Mathematik II, Programmieren I, Programmieren II
Arbeitsaufwand	240 Stunden, davon: 120 Stunden Präsenzzeit, 120 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	8
Semesterwochenstunden	8
Veranstaltungstyp	4 SWS Vorlesung, 4 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Computergraphik ist die wichtigste Art der Mediensynthese. Diese Kenntnisse und Fähigkeiten sind für Medieninformatiker/innen unverzichtbar. Sie ermöglichen berufliche Tätigkeiten in den Bereichen CAD, Medizinsysteme, Computerspiele und Entertainment-Systeme, computergestützte Produktgestaltung und Werbung, Training und Schulung, und anderen.
Lernziel	Kenntnis und Verstehen der polygonbasierten Computergraphik, insbesondere das hardware-unterstützte Rendering mit lokalen Beleuchtungsmodellen, und die Fähigkeit, graphische Anwendungen zu programmieren.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Fähigkeit zu abstraktem mathematischen Denken, Fähigkeit zu selbständigem Arbeiten, Analyse und Klassifikation von Problemen, kreatives Problemlösen, praktische Erfahrung
Lehrinhalte	Softwarearchitektur graphischer Anwendungen, Modellierung geometrischer Objekte mit Primitiven, 3-dimensionale Modell-Transformation, View-Transformation und Projektion, lokale Beleuchtung und Schattierung, Standard-Vertexshader und -Pixelshader, Texturen für reichhaltige Oberflächen, Graphik-Pipeline. Animation.
Literatur	<ul style="list-style-type: none">• Christian Schiedermeier, Foliensatz Computergraphik• Peter Shirley, Michael Ashikhmin, Michael Gleicher, Fundamentals of Computer Graphics (second edition), AK Peters, Natick, MA, 2005• Microsoft Windows Presentation Foundation Dokumentation• Adam Nathan: WPF 4, Kapitel 3D Graphics, Sams• Hanspeter Mössenböck: Kompaktkurs C# 6.0, dpunkt
Leistungsnachweis	Schriftliche Prüfung (90 min)

Medienkonzeption

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien Mediensynthese
Modul	Medienkonzeption
Modulverantwortliche	Max Ackermann
Dozent(en)	Peter Krüll, Joerg Meister
Vorkenntnisse	Gestaltungs- und Medienlehre I und II
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz 85 Stunden Vor- und Nachbereitung des Lehrstoffs, Übungsaufgaben und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS seminaristischer Unterricht
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Bildanalytische Fähigkeiten zählen zu den Grundvoraussetzungen jeder gestalterischen Arbeit. Hierzu eröffnet das Seminar einen basalen Reflektionshorizont, der dialogisch mit den Studierenden anhand von spezifischen Fragestellungen entwickelt wird. Schwerpunkte bilden ästhetische, funktionale und gestalterische Parameter.
Lernziel	<ul style="list-style-type: none">• Schriftliche Definition von Bildbestandteilen (ikonischen, ästhetischen, farblichen, kompositorischen Mustern) in Gruppenarbeit• Übernahme dieser Definitionen in die eigenständige Analyse unbekannter Bilder aus den Gebieten Kunst, Werbung und Design• Eigenständiger schriftlicher Aufbau von Zielkriterien für Bilder, die in der eigenen Mediengestaltung benötigt werden
Schlüsselqualifikation	<ul style="list-style-type: none">• Entwicklung eines bildreflexiven Denkens• Befähigung, medial aufbereitete Bilder mehrschichtig zu analysieren• Bewusstmachung formaler Gestaltungslösungen des Bildes
Lehrinhalte	<ul style="list-style-type: none">• Vermittlung von Grundkenntnissen zur Phänomenologie und Ontologie des Bildes• Einführung und Anwendung bildanalytischer Rezeptionstechniken
Literatur	<ul style="list-style-type: none">• Werner Faulstich: Einführung in die Medienwissenschaft. Probleme ? Methoden ? Domänen, München 2002• W.J.T. Mitchell: Das Leben der Bilder. Eine Theorie der visuellen Kultur, München 2008• Birgit Recki, Lambert Wiesing (Hg.): Bild und Reflexion. Paradigmen und Perspektiven gegenwärtiger Ästhetik, München 1997• Jürgen Stöhr (Hg.): Ästhetische Erfahrung heute, Köln 1996
Leistungsnachweis	Schriftliche Prüfung (90 min)

Interaktive Systeme

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien Mensch-Computer-Interaktion
Modul	Interaktive Systeme
Modulverantwortliche	Timo Götzelmann, Alexander Kröner
Dozent(en)	Timo Götzelmann, Alexander Kröner
Vorkenntnisse	Grundständige Programmierkenntnisse
Arbeitsaufwand	210 Stunden, davon: 90 Stunden Präsenz 120 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	7
Semesterwochenstunden	6
Veranstaltungstyp	4 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und Englisch
Beitrag zu den Zielen des Studiums	Die Studierenden lernen spezielle Problemstellungen der Entwicklung interaktiver Mensch-Computer Systeme kennen. Sie können wesentliche Anforderungen interaktiver Systeme identifizieren. Aufbauend auf grundständigen Programmierkenntnissen können sie damit selbstständig in Verbindung stehende Lösungsansätze entwickeln.
Lernziel	Kenntnis von Historie und Merkmalen interaktiver Systeme. Fähigkeit zur Klassifizierung solcher Systeme. Fähigkeit zur Beurteilung der Gebrauchstauglichkeit von Benutzerschnittstellen. Kenntnis von Entwicklungsprozess und Architekturmustern interaktiver Systeme. Fähigkeit zur Anwendung vorher genannter Kenntnisse zur Implementierung interaktiver Systeme.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, fachübergreifendes Wissen, praktische Erfahrung
Lehrinhalte	<ul style="list-style-type: none">• Historische Entwicklung und Arten von Benutzerschnittstellen• Interaktionstechnologien und deren Interaktionstechniken• Graphische Dialogsysteme• Modelle und Architekturmuster für interaktive Systeme• Entwicklungsframeworks
Leistungsnachweis	Schriftliche Prüfung (90 min.)

Software-Ergonomie

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Digitale Medien Mensch-Computer-Interaktion
Modul	Software-Ergonomie
Modulverantwortliche	Timo Götzelmann
Dozent(en)	Timo Götzelmann
Vorkenntnisse	Gestaltungs- und Medienlehre I und II
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz 85 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und Englisch
Beitrag zu den Zielen des Studiums	Urteilsfähigkeit über benutzerfreundliche Software und Kompetenz, Benutzerschnittstellen unter Berücksichtigung ergonomischer Kriterien zu entwerfen, zu analysieren und zu Verbesserungsmöglichkeiten zu identifizieren.
Lernziel	Kenntnis grundlegender kognitionspsychologischer Sachverhalte und der Mensch-Computer-Interaktion. Fähigkeit Konzepte für die Beurteilung und Validierung ergonomischer graphischer Dialogsysteme zu recherchieren und anzuwenden. Damit wird die Fähigkeit vertieft, gebrauchstaugliche Softwaresysteme zu entwerfen.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, theoretische Grundlagen, kreatives Problemlösen, fachübergreifendes Wissen
Lehrinhalte	<ul style="list-style-type: none">• Grundlagen der menschlichen Wahrnehmung und des Gedächtnisses• Menschliche Handlungsprozesse• Methoden zur Bewertung und Validierung benutzerfreundlicher Software• Hypothesentests sowie methodische Auswertung empirischer Daten zur Evaluation• Formen und Merkmale der Kommunikation und Handlung zwischen Mensch und Maschine• Richtlinien und gesetzliche Regelungen zur Software-Ergonomie
Leistungsnachweis	Schriftliche Prüfung (90 min)

Praktikum

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktisches Studiensemester
Modul	Praktikum
Modulverantwortliche	Wolfgang Bremer
Vorkenntnisse	Insbesondere Kenntnisse für die Anwendungsentwicklung
Arbeitsaufwand	20 Wochen in der Firma
Leistungspunkte	22
Semesterwochenstunden	0
Veranstaltungstyp	20 Wochen praktische Tätigkeit in einem Betrieb
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Praktische Erfahrung bei Anwendern und Dienstleistern
Lernziel	Kenntnisse erlangen in der Projektarbeit oder im Servicegeschäft von Informatik- oder Medien-Dienstleistern bzw. in Informatik- oder Medienbereichen. Alternativ sammeln von Erfahrungen in der Projektarbeit in Forschungs- und Entwicklungsbereichen für Informations-, Kommunikations- und Softwaretechnologien.
Schlüsselqualifikation	- Projekterfahrung - Zeitmanagement - Sozialkompetenz (Kooperationsfähigkeit, Teamfähigkeit)
Lehrinhalte	Verantwortliche Durchführung von Projekten/Teilprojekten der Softwareentwicklung oder verantwortliche Übernahme von Aufgaben/Teilaufgaben bei der Abwicklung von Informatik-Dienstleistungen.
Leistungsnachweis	Ausbildungsbestätigung, Zeugnis

Praxisseminar

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktisches Studiensemester
Modul	Praxisseminar
Modulverantwortliche	Wolfgang Bremer
Dozent(en)	Tobias Bocklet, Wolfgang Bremer, Florian Gallwitz, Joachim Scheja, Peter Trommler, Bartosz von Rymon Lipinski
Vorkenntnisse	keine
Arbeitsaufwand	60 Stunden, davon: 30 Stunden Präsenzzeit, 30 Stunden Vor- und Nachbereitung der Themen
Leistungspunkte	3
Semesterwochenstunden	2
Veranstaltungstyp	2 SWS Seminar mit Vorträgen
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Grundlagen projektorientierter Arbeit
Lernziel	Erfahrungen im Praktikum bei der Projektarbeit oder im Servicegeschäft von Anwendern, Informatik- oder Medien-Dienstleistern reflektieren, sowie mündlich und schriftlich darstellen und diskutieren können. Ergebnisse und erworbene Kompetenzen demonstrieren und dokumentieren können.
Schlüsselqualifikation	Präsentationskompetenz, technisches Schreiben
Lehrinhalte	<ul style="list-style-type: none">• Vermitteln und Einüben von Präsentationstechniken• Feedback und Diskussion• Gestaltung von Berichten
Literatur	Seifert, J.W.: Visualisieren, Präsentieren, Moderieren, 20. Auflage, Gabal 2004
Leistungsnachweis	Referate und Praxisbericht

Projektmanagement

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Praktisches Studiensemester
Modul	Projektmanagement
Modulverantwortliche	Thomas Voit, Peter Rausch
Dozent(en)	Martin Geier, Jörg Meier
Vorkenntnisse	Schulwissen
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbereiten des Lehrstoffs sowie Übungsaufgaben
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS Vorlesung mit Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Fähigkeiten in der Organisation und Koordination von zeitlich limitierten Projekten
Lernziel	Methoden der Projektplanung, -steuerung und -kontrolle kennen und anwenden können. Projekte anhand charakteristischer Merkmale definieren und von sonstigen Vorhaben und Formen der Organisationsgestaltung unterscheiden können. Die Werte, Prinzipien und Vorgehensmodelle iterativer und agiler Ansätze kennen und deren spezifische Vor- und Nachteile gegenüber traditionellen Projektmanagement-Ansätzen beurteilen können.
Schlüsselqualifikation	Projektmanagement, Organisationskompetenz, Zeitmanagement
Lehrinhalte	<ul style="list-style-type: none">• Projektplanung, -steuerung und -kontrolle mit Prozessmodellwahl• Meilensteinplanung• Werte, Prinzipien und Vorgehensmodelle des agilen Projektmanagements• Terminplanung mit Netzplantechnik• Kosten- und Aufwandsschätzung• Ressourcenzuordnung• Steuerung und Überwachung• Dokumentation• Überprüfung der Zielerreichung• Lernen aus Projekten
Literatur	<ul style="list-style-type: none">• Hindel, B. u. a.: Basiswissen Projektmanagement, Heidelberg 2004, dpunkt.Verlag• Tumascheit, K. D.: Überleben im Projekt. 10 Projektfallen und wie man sie umgeht, München (Redline) 2014• Heintel, P.; Krainz, E. E.: Projektmanagement. Hierarchiekriese. Systemabwehr. Komplexitätsbewältigung, 5. Auflage, Wiesbaden (Gabler) 2011• Schelle, H.; Ottmann, R.; Pfeiffer, A.: ProjektManager, 3. Auflage, Nürnberg (Deutsche Gesellschaft für Projektmanagement) 2008
Leistungsnachweis	Dieser Leistungsnachweis kann nur praxisbegleitend abgelegt werden. (Klausur 90 Min.)

Hilfsmittel

Zulassungsvoraussetzung

Agile Methoden in ausgewählten Anwendungsbereichen

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Agile Methoden in ausgewählten Anwendungsbereichen
Modulverantwortliche	Claudia Hirschmann
Dozent(en)	Claudia Hirschmann
Vorkenntnisse	Grundkenntnisse in Vorgehensmodellen, Grundkenntnisse in Softwareengineering
Arbeitsaufwand	Umfang 2 SWS und häusliche Arbeiten
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Die Veranstaltung wird auf vier Tagesblöcke im Semester verteilt. Zwischen den Tagesblöcken ist Arbeit in Übungsgruppen einzuplanen und auszuführen. Die Tagesblöcke sind interaktiv gestaltet (Mitarbeiten der Teilnehmenden).
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Grundlagen zu agilen Methoden in Theorie und Praxis, Grundlagen ausgewählter Anwendungsbereiche aus der Praxis, Vertiefungen durch praxisorientierte Übungen, Befähigung zur Einschätzung von Anwendungen agiler Methoden unter verschiedenen Rahmenbedingungen. Förderung des interdisziplinären Verständnisses. Analytische Fähigkeiten & Visualisierung von Prozessen der Softwareentwicklung, von Anforderungen an Vorgehensmodelle und an die Softwareentwicklung. Vertiefung des Verständnisses zum Prozessmanagement und zu agilen Methoden, selbständiges Erarbeiten und Lernen von (kleineren) Teilgebieten des FWPF, Präsentation von Arbeitsergebnissen.
Lernziel	Einführung, Kenntnis, Verständnis und Transfer von Agilen Methoden in ausgewählten Anwendungsbereichen. Agile Methoden der Software-Entwicklung fokussieren u.a. auf zahlreiche Facetten der Organisation der operativen Tätigkeiten bei der Erstellung von Software, wie sie z.B. in den Definitions-, Realisierungs- oder Testphasen auftreten. Ausgewählte Anwendungsbereiche können beispielsweise regulierte Anwendungsbereiche sein, wie z.B. die Medizintechnik oder Automotive, die aufgrund von gesetzlichen und normativen Vorgaben besondere Anforderungen an die Software- Entwicklung stellen, oder verschiedene Szenarien in Unternehmen.
Schlüsselqualifikation	Agile Methoden und Prinzipien, Vorgehens- und Prozessmodelle. Anforderungen an und Analyse von agilen Methoden in ausgewählten Anwendungsbereichen. Fähigkeit zum Transfer von Agilen Modellen auf ausgewählte Anwendungsbereiche, Transfer in die Praxis, Entwicklung von Konzepten zu agilen Methoden, Praxisorientierte Prozessuntersuchungen & Prozessmanagement, Kommunikationsmanagement, Interdisziplinarität, Teamarbeit, Kommunikations- und Teamfähigkeiten, Präsentationsfähigkeit, Erarbeiten von ausgewählten Themen selbständig und im Team .
Lehrinhalte	Die Veranstaltung führt zu einen in wichtige und häufig anzutreffende agile Software-Entwicklungs-Frameworks ein, u.a. in SCRUM. Zum anderen zeigt sie die speziellen Anforderungen auf, die von ausgewählten Anwendungsbereichen an die SW-

Entwicklung gestellt werden. Solche Anwendungsbereiche können z.B. der stark regulierte Bereich der Medizintechnik oder Automotive oder andere Szenarien in Unternehmen sein. Neben den reinen Lerninhalten zur Charakterisierung der agilen Methoden und der ausgewählten Bereiche, nehmen Anwendungs-, Praxis- und Transferbeispiele- und Übungen einen breiten Raum ein.

Literatur

- The Scrum Primer by P. Deemer, G. Benefield, C. Larman, B. Vodde, u.ä.; - Extreme Programming: <http://www.extremeprogramming.org>; - Das "Manifest für Agile Software-Entwicklung", <http://agilemanifesto.org>; - Normen, z.B. IEC/ISO 62304; - FDA Websites, z.B. <http://www.fda.gov/MedicalDevices/default.htm>.

Leistungsnachweis

Referat mit Handout (35 Min.) (60%), schriftliche Hausarbeit (8 Seiten) (40%); Aufgrund des Veranstaltungsdesigns herrscht Anwesenheitspflicht.

Hilfsmittel

Zulassungsvoraussetzung

Aktuelle Trends im Einsatz von Projektmanagement-Methoden

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Aktuelle Trends im Einsatz von Projektmanagement-Methoden
Modulverantwortliche	Martin Geier
Dozent(en)	Martin Geier
Vorkenntnisse	Grundlagen in Vorgehensmodellen Grundlagen in Softwareengineering
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit 40 Stunden Projektarbeit 45 Stunden Vor- und Nachbereitung, Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Lernziel	Kenntnisse in aktuellen Trends des Software-Projektmanagements und deren Einsatz
Schlüsselqualifikation	Projektmanagement Methoden, Prozessmodelle, Qualitätsmanagement
Lehrinhalte	<p>Projektmanagement ist die zentrale Disziplin, um Projekte auszusteuern und durch Analysen und Abschätzungen Projektmetriken messbar und damit kontrollierbar zu machen. Im Laufe der Zeit haben sich Modelle und Mechanismen als „Best Practices“ etabliert, andere haben an Bedeutung verloren. Die Vorlesung geht auf aktuelle Trends und deren praktischen Einsatz ein und präsentiert entsprechende „Best Practices“.</p> <p>In dieser Lehrveranstaltung werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none">• Einsatz aktueller Projektmanagementmethoden• Aktuelle Verfahrensmodelle und deren Entwicklung• Agile Methoden im Projektmanagement und deren Einsatz (am Beispiel SCRUM)• Vergleich von Projektplanung und Kontrolle in verschiedenen Vorgehensmodellen• Softskills und Mitarbeiterführung / „Human Resource Management“• Aufwandsabschätzungen und deren praktischer Einsatz• Projektbegleitendes Qualitätsmanagement• Reifegradmodelle zur Messung von Prozessqualität
Literatur	<ul style="list-style-type: none">• Hindel, Hörmann, Müller, Schmied: Basiswissen Software-Projektmanagement. dpunkt, 2006• Schwaber: Agile Project Management with Scrum. Microsoft Press, 2004• V-Modell XT. www.v-modell-xt.de
Leistungsnachweis	mündliche Prüfung, Dauer 20 Minuten

Algorithmische Graphentheorie

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Algorithmische Graphentheorie
Modulverantwortliche	Alexander Hufnagel
Dozent(en)	Alexander Hufnagel
Vorkenntnisse	Mathematik 1 und 2, insbesondere Kombinatorik, Lineare Algebra und Rekursionsgleichungen
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbearbeitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit Übungen
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Lernziel	Kenntnis der grundlegenden Begriffe der Graphentheorie sowie der Algorithmen auf Graphen, sowie Übertragung dieser Inhalte auf konkrete Anwendungsbeispiele. Analyse der Laufzeit bzw. Komplexität, Entwickeln von Problemstellungen als Modell der Graphentheorie. Bewertung von Lösungsverfahren für konkrete Projekte.
Lehrinhalte	<p>Graphen zählen zu den wichtigsten Modellen der Informatik. Viele Problemstellungen lassen sich graphentheoretisch beschreiben, beispielsweise</p> <ul style="list-style-type: none">• Ausfallsicherheit von Netzen• Suchstrategien• Finden von kürzesten Wegen• Routenplanung• Zuordnungsprobleme• Maximale Flüsse in Netzwerken <p>Nach einer Einführung in die Theorie und Darstellung von Graphen werden die wichtigsten Algorithmen für Graphentheorie vorgestellt, analysiert und bewertet. Die Methoden werden dann auf konkrete Fragestellungen übertragen. Begleitend zur Vorlesung werden Übungen angeboten.</p>
Literatur	<ul style="list-style-type: none">• Christina Büsing, Graphen- und Netzwerkoptimierung, Spektrum 2010• Shimon Even, Graph Algorithms, 2nd ed., Cambridge 2012• Dieter Jungnickel, Graphs, Networks and Algorithms, 3rd ed., Springer 2007• Sven Oliver Krumke, Hartmut Noltemeier: Graphentheoretische Konzepte und Algorithmen, 3. Auflage, Springer 2012• Angelika Steger, Diskrete Strukturen 1, Springer Verlag 2007• Volker Turau, Algorithmische Graphentheorie, Oldenbourg 2009• T. Cormen et al: Introduction to Algorithms, MIT Press, 2009
Leistungsnachweis	Schriftliche Prüfung (90 min)

Clean Code in Java-Projekten

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Clean Code in Java-Projekten
Modulverantwortliche	Martin Hock
Dozent(en)	Martin Hock
Vorkenntnisse	Java-Kenntnisse sind von Vorteil.
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbearbeitung, Übungsaufgaben und Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vertiefung von Kenntnissen und praktischen Fertigkeiten im Bereich Softwarequalität. Kennenlernen von Tools und Vorgehensweisen aus der Praxis.
Lernziel	Die Studierenden erwerben vertiefte Kenntnisse und Fähigkeiten zur Analyse, Beurteilung und Verbesserung von Software-Qualität. Sie sind in der Lage Prinzipien, Patterns, Techniken und Tools, die zum Schreiben von sauberem Code benötigt werden, anzuwenden.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Fähigkeit zu selbständigem Arbeiten, Analyse und Lösung von Problemen.
Lehrinhalte	<p>Die Prüfung von Software ist nicht auf eine bestimmte Phase im Projekt beschränkt. Schon während der Coding-Phase bzw. des System-Build-Prozesses können kritische und schwierig zu findende Softwaredefekte im Quellcode aufgedeckt werden. In der Vorlesung werden die dafür nötigen Verfahren und Tools vorgestellt.</p> <p>Unter anderem werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none">• Übersicht über die Grundlagen der Software-Qualität• Softwaremetriken, Metrikanwendung in der Praxis• Überblick über Prinzipien, Best Practices und Code Smells• Einhaltung und Überprüfung Java Code Conventions• Statische Softwareprüfung, insbesondere Review-Techniken und statische Programmanalyse• Sicherung der Softwarequalität mit Werkzeugen wie SonarQube, PMD, SpotBugs und Checkstyle• Softwaretests mit JUnit• Überprüfen der Testabdeckung (Code Coverage)• Continuous Integration• Design Prinzipien• Design Patterns (GoF)
Literatur	<ul style="list-style-type: none">• Schneider, Kurt: Abenteuer Software Qualität – Grundlagen und Verfahren für Qualitätssicherung und Qualitätsmanagement, dpunkt.verlag, 2007

- Robert, Martin: Clean Code – Refactoring, Patterns, Testen und Techniken für sauberen Code, mitp-Verlag, 2009
- Lilienthal, Carola: Langlebige Software-Architekturen, Dpunkt Verlag, 2015
- Bloch, Joshua: Effective Java – Second Edition, Addison Wesley, 2008
- Roock, Stefan: Refactorings in grossen Softwareprojekten, Dpunkt Verlag, 2004
- Gamma, Erich: Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Professional, 1994
- Robert C. Martin: Agile Software Development: Principles, Patterns and Practices, Prentice Hall, 2003

Leistungsnachweis

Schriftliche Prüfung (90 min)

Hilfsmittel

Zulassungsvoraussetzung

Conversational Interfaces - Design und Implementierung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Conversational Interfaces - Design und Implementierung
Modulverantwortliche	Steffen Blümm
Dozent(en)	Steffen Blümm
Vorkenntnisse	Kenntnisse in JavaScript / TypeScript / Node.js von Vorteil
Arbeitsaufwand	150 Stunden, davon: 50 Stunden Präsenzzeit 100 Stunden Vor- und Nachbereitung der Vorlesung, Bearbeitung der Assignments, Bericht
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit praktischen Übungen und Assignments
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	
Lernziel	Nach der Teilnahme an diesem Modul sind die Studierenden in der Lage <ul style="list-style-type: none">• wichtige Begriffe im Bereich Conversational Interfaces zu definieren• Methoden für die Gestaltung von Conversational Interfaces anzuwenden• Entscheidungen zu Vorgehensweise für Usability Tests zu treffen• In Dialogflow und RASA einfache Interaktionen über Dialoge zu konzipieren• Die Verarbeitung von Interaktionen mit eigener Logik zu konkretisieren
Schlüsselqualifikation	Konzeption von Services für konversationsgestützte Interaktion (Chatbot / Voice Assistants) und Verarbeitung von User Intents. Verständnis und Kenntnis für grundlegende Prinzipien von Conversational Design und Conversational User Experience.
Lehrinhalte	Conversational Interfaces bieten Menschen die Möglichkeit über natürliche Sprache mit digitalen Services zu interagieren. Zum einen ermöglicht dies die Nutzung von Diensten in Situationen, in denen ein physikalischer Kontakt oder Blickkontakt mit einem Device nicht möglich oder erwünscht ist. Zum anderen wird der nicht-lineare Zugriff auf Informationen und, ohne den Kontext zu wechseln, die Verknüpfung von Aufgaben vereinfacht. Im Vergleich mit Mobile-Applications oder Web-Anwendungen wird die Systemarchitektur um einige Standardkomponenten erweitert. Diese Form der Interaktion erfordert weiterhin die Beachtung von neuen Prinzipien in Design und User Experience. Es wird auf folgende Aspekte eingegangen: <ul style="list-style-type: none">• Architektur von Conversational Interfaces

- Conversational Front-Ends
- Natural Language Understanding Services (NLUs)
- Webhooks
- Conversational User Experience Design
 - UX Research
 - UX Design
 - Conversation Design
 - Conversational Design
 - Conversational UX
 - Prototyping
- Implementierung von Conversational Interfaces
 - Verwendung von Dialogflow und RASA
 - Intents, Utterances, Slots und Entities
 - Utterances definieren und verarbeiten
 - Slot-filling
 - Intents verknüpfen
 - Erweiterte Funktionalitäten von Google Assistant

Literatur

- Pearl, C.: Designing Voice User Interfaces, O'Reilly, 2017
- Shevat, A.: Designing Bots, O'Reilly, 2017
- McTear, M., Callejas, Z., Griol, D.: The Conversational Interface: Talking to Smart Devices, Springer, 2016
- Nass, C., Brave, S.: Wired For Speech, MIT Press, 2007
- Reeves, B., Nass, C.: The media equation: how people treat computers, television and new media like real people and places. Cambridge University Press, New York, 1996

Leistungsnachweis

Präsentationen der Assignments (40%) und Bericht (60%)

Hilfsmittel

Zulassungsvoraussetzung

Datenmodellierung mit XML

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Datenmodellierung mit XML
Modulverantwortliche	Alexander Kröner
Vorkenntnisse	Erforderlich <ul style="list-style-type: none">• Programmieren I, II• Web-Programmierung (insbesondere HTML, PHP) Empfohlen <ul style="list-style-type: none">• Javascript (Grundkenntnisse)• ERM
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit 85 Stunden Vor- und Nachbereitung, Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung, Übungen
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Beherrschung der weit verbreiteten XML-Technologien trägt zur praxisorientierten Ausbildung bei. Die für das Fach charakteristischen semi-strukturierte Datenmodelle sind ein Bestandteil moderner Datenbankkonzepte (NoSQL). Der Umgang mit XML Schema und XSLT ermöglicht einen Einblick in Grammatiken und regelbasierte Systeme als Komponenten fortgeschrittener Programmiersprachen.
Lernziel	Kenntnis der Eigenschaften von semi-strukturierten Datenmodellen und well-formed XML. Verstehen der Auswirkungen dieses Modellierungsansatzes. Anwendung dieses Wissens um konzeptionelle Datenmodelle zu analysieren und ein korrespondierendes Model mittels XML Schema zu entwickeln. Fähigkeit, XSLT anzuwenden und zu entwickeln als Werkzeug zur Transformation von Modellen.
Schlüsselqualifikation	Theoretisches Verständnis und praktische Beherrschung ausgewählter XML-Technologien.
Lehrinhalte	Die Extensible Markup Language (XML) ist eine Auszeichnungssprache zur Repräsentation semi-strukturierter Daten in Form von Textdateien. XML bildet im Internet eine der Basistechnologien für den plattform- und implementationsunabhängigen Austausch von Daten. Schwerpunkt der Vorlesung bilden u.a.: <ul style="list-style-type: none">• Grundlagen semi-strukturierter Datenmodelle: Strategien zur Serialisierung relationaler Modelle, well-formed XML (u.a. Namenskonventionen, Whitespace-

Behandlung, Namensräume)

- Dokumentstrukturen validieren durch Schema-Sprachen: XML Schema Definition (u.a. Erweiterbarkeit, Namensräume, Validierung von Verweisstrukturen), JSON Schema
- Zugriff auf XML-Dokumente mit XPath
- Transformation von XML mit XSLT (u.a. Template-Regeln, Template-Prozessor, Template-Design-Strategien, Modell-Transformation, View-Transformation, Client- und Server-seitige Transformation)
- JSON als Transport-View

Leistungsnachweis

Klausur (90 min)

Datenschutz

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Datenschutz
Vorkenntnisse	
Arbeitsaufwand	
Leistungspunkte	5
Semesterwochenstunden	4
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	
Lernziel	Die Studierenden haben einen Überblick über die geltenden Gesetze zum Datenschutz. Sie haben die Kompetenz, datenschutzrechtliche Probleme in der betrieblichen Praxis zu erkennen und Handlungsanweisungen daraus abzuleiten.
Schlüsselqualifikation	
Lehrinhalte	<p>Neben einer Einführung in das Datenschutzrecht steht vor allem der technische Datenschutz im Vordergrund. Die Themen werden sehr praxisnah behandelt.</p> <ul style="list-style-type: none">- EU-Datenschutz-Grundverordnung und ePrivacy-Verordnung- Überblick über kryptographische Verfahren- Privacy by Design- Anonymisierung und Pseudonymisierung in der Praxis- Sichere Kommunikation in der Praxis: E-Mail und Messenger- Datenschutz im Web: Tracking, Social Plugins und co.- Identitätsmanagement- Anonymität im Internet- Anonyme Bezahlverfahren / Bitcoin- Datenschutz-Folgenabschätzung
Literatur	<ul style="list-style-type: none">- Petrlc, Sorge: "Datenschutz: Einführung in technischen Datenschutz, Datenschutzrecht und angewandte Kryptographie", Springer-Vieweg, 2017.- Wybitul: „EU-Datenschutz-Grundverordnung im Unternehmen: Praxisleitfaden (Kommunikation & Recht)“, 2016.

Leistungsnachweis

Hilfsmittel

Zulassungsvoraussetzung

Digitale Musikinstrumente mit Smartphones und PureData

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Digitale Musikinstrumente mit Smartphones und PureData
Modulverantwortliche	Sebastian Trump
Dozent(en)	Sebastian Trump
Vorkenntnisse	
Arbeitsaufwand	75 Stunden, davon: 30 Stunden Präsenz und 45 Stunden Vor- und Nachbearbeitung in Projektgruppen
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Seminar
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	
Lernziel	In diesem hochschulübergreifenden Seminar für Studierende der Hochschule für Musik und der TH Nürnberg werden gemischte Teams Ideen für digitale Musikinstrumente entwickeln, prototypisch umsetzen und evaluieren.
Schlüsselqualifikation	<ul style="list-style-type: none">• Gemeinsame Projektarbeit in interdisziplinären Teams
Lehrinhalte	<ul style="list-style-type: none">• Klangsynthese mit PureData• Open-Sound-Control over WiFi• Mapping von Smartphone-Sensordaten auf Syntheseparameter• Interaktionsdesign• Evaluation der entwickelten Prototypen anhand kleiner Performances
Literatur	<ul style="list-style-type: none">• J. Ryan. Some remarks on musical instrument design at steim. Contemporary music review, 6(1):3–17, 1991.• V. Bellotti, M. Back, W. K. Edwards, R. E. Grinter, A. Henderson, and C. Lopes. Making sense of sensing systems: five questions for designers and researchers. In Proc. of CHI, pages 415–422. ACM, 2002.• D. Svanaes. Philosophy of Interaction. The Interaction Design Foundation, Aarhus, Denmark, 2013.• K. Tahiroglu. An exploration on mobile interfaces with adaptive mapping strategies in pure data. In Proceedings of the 4th International Pure Data Convention. Bauhaus Universität, Weimar, 2011.• M. Wright. Open sound control-a new protocol for communicating with sound synthesizers. In Proceedings of the 1997 International Computer Music Conference, pages 101–104, 1997.
Leistungsnachweis	<ol style="list-style-type: none">1. Projektarbeit (Konzeption des Instruments, Entwicklung eines Prototyps, Dokumentation auf ca. 5 Seiten)2. Präsentation des Instruments als musikalische Performance (Dauer ca. 5 Minuten)

Gewichtung der Teile 2:1.

Hilfsmittel

Zulassungsvoraussetzung

E-Commerce-Recht

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	E-Commerce-Recht
Modulverantwortliche	Alexander Kübler-Kreß
Dozent(en)	Alexander Kübler-Kreß
Vorkenntnisse	Schulwissen in Wirtschaft und Recht
Arbeitsaufwand	75 Stunden, davon: 30 Stunden Präsenzzeit, 45 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Vorlesung mit integrierten Fallbearbeitungen
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Befähigung zu logisch-analytischem Denken und strukturierter Problemlösung.
Lernziel	Die Studierenden haben ein sicheres Judiz hinsichtlich rechtlicher Probleme beim Internet-Handel und kennen rechtswissenschaftliche Arbeitsmethoden.
Schlüsselqualifikation	Es wird das Erörtern von Rechtsproblemen in freier Rede geübt.
Lehrinhalte	<ul style="list-style-type: none">• Vertragsschluss im Internet• elektronische Willenserklärungen• Schriftform und digitale Signatur• Beweiswert digitaler Dokumente• Allgemeine Geschäftsbedingungen• Zahlungsmittel im elektronischen Geschäftsverkehr• Verbraucherschutz im Internet: Haustürwiderrufsrecht, Fernabsatzrecht, Bestellkorrektur, Verbraucher kreditrecht, Ratenlieferungsverträge, Mehrwertdienste, Pflichten des Unternehmers beim E-Contracting
Literatur	http://www.uni-muenster.de/Jura.itm/hoeren/materialien/Skript/Skript_Internetrecht_Februar2010.pdf zum kostenlosen Download
Leistungsnachweis	Klausur (90 min)

Einführung in die Funktionale Programmierung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Einführung in die Funktionale Programmierung
Modulverantwortliche	Peter Trommler
Dozent(en)	Peter Trommler
Vorkenntnisse	Programmieren I und II, Mathematik I, Algorithmen und Datenstrukturen
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 45 Stunden selbständige Übungen am Rechner, 40 Stunden zur Nachbereitung und Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit Praktikum
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Spezifikation komplexer Systeme und deren Implementierung mit funktionalen Programmen
Lernziel	Programmieren im funktionalen Programmierparadigma, Verbesserung der Analysefähigkeit und Programmierkompetenz auch in objektorientierten Sprachen
Schlüsselqualifikation	Abstraktionsvermögen, Durchhaltevermögen, logisches Schließen und komplexes Denken
Lehrinhalte	<p>Am Beispiel der Programmiersprache Haskell werden die Prinzipien des Funktionalen Programmierens betrachtet und in praktischen Übungen im Labor eingeübt.</p> <p>Folgende Themenbereiche werden behandelt:</p> <ul style="list-style-type: none">• Typen, Klassen, Parametrische Polymorphie• Funktionen, Definition, Komposition• Rekursion, Funktionen höherer Ordnung, Kombinatoren• Reine Funktionen und Nebeneffekte, Monaden• Auswertungsstrategien: strikte und faule Auswertung, unendliche Listen
Literatur	<ul style="list-style-type: none">• Graham Hutton, Programming in Haskell, Cambridge University Press, Cambridge, England, 2. Auflage, 2016.• Brian O'Sullivan, John Goerzen, and Don Stewart, Real World Haskell, O'Reilly, Sebastopol, California, U.S.A., 2008.• Richard Bird, Pearls of Functional Algorithm Design, Cambridge University Press, Cambridge, England, 2010.• Richard Bird, Introduction to Functional Programming using Haskell, Prentice Hall Europe, Harlow, Essex, England, 2. Auflage, 1998.• Miran Lipovaca, Learn You a Haskell for Great Good, no starch press, San Francisco, California, U.S.A., 2011.
Leistungsnachweis	Mündliche Prüfung (Dauer: 20 Minuten)

Hilfsmittel

Zulassungsvoraussetzung

Einführung in Operations-Research

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Einführung in Operations-Research
Modulverantwortliche	Claudia Hirschmann
Dozent(en)	Claudia Hirschmann
Vorkenntnisse	Grundlagen der Mathematik, Mathematik-Vorlesungen.
Arbeitsaufwand	Umfang 2 SWS, und häusliche Arbeiten
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Die Veranstaltung wird auf vier Tagesblöcke im Semester verteilt. Zwischen den Tagesblöcken ist Arbeit in Übungsgruppen einzuplanen und auszuführen. Die Tagesblöcke sind interaktiv gestaltet (Mitarbeiten der Teilnehmenden).
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Grundlagenwissen, Modelle, Methoden, Herangehensweisen zum Operations-Research in Theorie und Praxis, vertieft durch praxisorientierte und Transfer-Übungen. Mathematische und Operations-Research Grundlagen z.B. der linearen Programmierung und anderer Methoden zur Entscheidungsunterstützung. Analytische Fähigkeiten bzgl. Problemstellungen im Hinblick auf Operations- Research. Befähigung zum Transfer von Modellen aus dem Operations- Research auf verschiedene Problemstellungen aus der Informatik, der Geschäftsplanung und dem Prozessmanagement. Selbständige Erarbeitung von (kleineren) Teilgebieten des FWPF, Problemlösung in Arbeitsgruppen, Präsentation der Arbeitsergebnisse.
Lernziel	Einführung, Kenntnis ausgewählter Begriffe und Vorgehensweise aus Teildisziplinen des Operations- Research, und Anwendungsbeispiele aus dem Operations-Research. Unter „Operations-Research“ wird die Modell-gestützte Vorbereitung von Entscheidungen zur Gestaltung und Lenkung sozio-technischer Systeme verstanden. Es ist geprägt durch die Zusammenarbeit von Angewandter Mathematik, Wirtschaftswissenschaften, Informatik und einigen Ingenieurwissenschaften.
Schlüsselqualifikation	Fähigkeit zur grundlegenden mathematischen Modellbildung, zur Analyse und zur Klassifikation von Fragestellungen aus der Praxis u.a. aus der Geschäftsplanung, betriebswirtschaftlichen Fragestellungen und dem Prozessmanagement als Grundlage für die weiteren Lösungsschritte. Fähigkeit zum Verständnis komplexer Probleme und deren Lösungsansätze aus dem Operations- Research, Transfer von Mathematischen Modellen auf verwandte Fragestellungen, Interdisziplinarität, Bearbeitung interdisziplinärer Fragestellungen, Teamarbeit, Kommunikations- und Teamfähigkeiten, Präsentationsfähigkeit, selbständiges Problemlösen und Problemlösen in Arbeitsgruppen.
Lehrinhalte	Die Veranstaltung führt in die Arbeitsweise und Teilgebiete des „Operations-Research“, d.h. der Optimierung von technisch-wirtschaftlichen Problemstellungen unter Beachtung von Nebenbedingungen (wie z.B. Ressourcenbeschränkungen), ein. Ein Einstieg in Operations- Research ist die Betrachtung von Problemstellungen aus

der Praxis, die sich mittels linearer Gleichungen und Ungleichungen beschreiben lassen. Ein in der Vorlesung behandeltes Teilgebiet ist die lineare Optimierung. Neben den reinen Lehrinhalten zu den Teilgebieten des Operations-Research nehmen Anwendung und praktische Beispiele und Übungen einen breiten Rahmen ein.

Literatur

- R. E. Burkard, U. T. Zimmermann: „Einführung in die Mathematische Optimierung“, Springer Verlag; - W. Domschke: Einführung in Operations Research, Springer Verlag; - B. Werners: Grundlagen des Operations Research, Mit Aufgaben und Lösungen, Springer Verlag

Leistungsnachweis

Referat mit Handout (30 Min.) (50%), schriftliche Befragung (60 Min.) (50%); Aufgrund des Veranstaltungsdesigns herrscht Anwesenheitspflicht.

Einstieg in C++ für C#-Programmierer

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Einstieg in C++ für C#-Programmierer
Modulverantwortliche	Bartosz von Rymon Lipinski
Dozent(en)	Bartosz von Rymon Lipinski
Vorkenntnisse	Vorkenntnisse in objektorientierter Programmierung mit C# (erfolgreich abgeschlossenes Programmieren II)
Arbeitsaufwand	75 Stunden, davon: 30 Stunden Präsenzzeit, 45 Stunden für Vor- und Nachbereitung des Lehrstoffs, Selbständige Lösung von Programmieraufgaben.
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Seminar, Projektarbeit
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung von Kenntnissen aus der praktischen und angewandten Informatik. Befähigung implementatorische Herausforderungen einzuordnen und praxisnahe Lösungen umzusetzen.
Lernziel	Kompetenz die konzeptionellen und technischen Rahmenbedingungen in der modernen Programmierung zu erläutern und zu bewerten. Fähigkeit zur Umsetzung eines objektorientierten Systems (mit der Programmiersprache C++). Fähigkeit hinsichtlich des Transfers zwischen verschiedenen objektorientierten Programmiersprachen (C++ und C#).
Schlüsselqualifikation	Praktisches Problemlösen, selbständiges Arbeiten, Kommunikations- und Teamfähigkeit.
Lehrinhalte	<ul style="list-style-type: none">• Modernes C++: Einführung und Abgrenzung zu C#• Einrichtung der Entwicklungsumgebung• Prozedurale Programmierung (Variablen, Kontrollstrukturen, Funktionen, Ein-/Ausgabe, Zeiger/Referenzen, Präprozessor)• Objektorientierung (Klassen/Structs, Vererbung und Polymorphie, inkl. virtuelle und abstrakte Methoden)• Weiterführendes (Namensräume, Überladen von Operatoren, Ausnahmebehandlung, Grundlagen von Templates)• Überblick über die C++ Standardbibliothek (String- und Container-Klassen, Smart pointers, mathematische Berechnungen, Algorithmen)• Eventuell erste Schritte mit dem Qt Framework
Literatur	<ul style="list-style-type: none">• Wolf, J.: Grundkurs C++ - C++-Programmierung verständlich erklärt, Galileo Computing, 2013• Kalista, H.: C++ für Spieleprogrammierer, Carl Hanser Verlag GmbH & Co. KG, 2016• Stroustrup, B.: Die C++-Programmiersprache - Aktuell zu C++11, Carl Hanser Verlag

Leistungsnachweis

Praktische Studienarbeit (Kommentiertes Software-Produkt, entsprechend 35 Stunden Entwicklungsaufwand).

Hilfsmittel

Zulassungsvoraussetzung

Entwicklung eines autonom fahrenden Modellfahrzeugs

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Entwicklung eines autonom fahrenden Modellfahrzeugs
Modulverantwortliche	Ramin Tavakoli Kolagari
Vorkenntnisse	gute Programmierkenntnisse
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	2 SWS Vorlesung, 2 SWS Übung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Bei der Entwicklung des autonomen Modellfahrzeugs lernen die Studierenden: - vertiefte Programmierfertigkeiten durch die Anwendung der Programmierung in einem größeren, zielorientierten Projekt, - Grundlagen der Verifikation (sowie Debuggen) von softwareintensiven Systemen und - Grundlagen des Softwareprojektmanagements.
Lernziel	Bei Abschluss des Lernprozesses wird der erfolgreiche Studierende in der Lage sein, Konzepte und Mechanismen zur Streckenfindung autonom fahrender Modellfahrzeuge zu kennen und diese für konkrete Szenarien anwenden zu können, die unterschiedlichen Methoden für die Entwicklung einfacher eingebetteter Systeme auf Mikrocontrollerebene auswählen und eine Anwendung auf der Basis von einem vorgegebenen Regelwerk prototypisch programmieren zu können.
Schlüsselqualifikation	Fähigkeit zum Verständnis der Herausforderung bei der Entwicklung autonomer Systeme durch die vorgegebenen Rahmenbedingungen.
Lehrinhalte	Im Rahmen dieser Lehrveranstaltung geht es um die Entwicklung eines autonomen Modellrennautos, das sich in einem (internationalen) Wettkampf mit anderen studentischen Teams messen können soll. Die Hardware wird von NXP gesponsert. Mehr Informationen zum Cup finden Sie hier . Die Teilnahme am NXP Cup ist gegebenenfalls Teil des Kurses. Der Wettbewerbsgedanke ist ausdrücklich erwünscht. Konkrete Lehrinhalte sind: <ul style="list-style-type: none">• Programmierung in C• Enge Arbeit im studentischen Team mit Rollenverteilung• Weiterführende Einblicke in die Prozesse der Automotive Softwareentwicklung und Systemmodellierung• Echtzeitsysteme• Robotik
Literatur	<ul style="list-style-type: none">• Ulrich Kaiser, Martin Guddat: C/C++ Das umfassende Lehrbuch, Galileo Computing -

ISBN 978-3-8362-2757-5

- Siciliano, Khatib: Springer Handbook of Robotics, Springer, 2016.
- Robotic-Conferences IROS and ICRA:
 - <http://ieeexplore.ieee.org/xpl/conhome.jsp?punumber=1000393>
 - <http://ieeexplore.ieee.org/xpl/conhome.jsp?punumber=1000639>

Leistungsnachweis

Vortrag inklusive Übungsaufgabe für Kursteilnehmer (70 Min., 50%) und schriftliche Dokumentation (20 Seiten) sowie mündliche Ergebnispräsentation der Projektarbeit (20 Min., 50%).

Graphical User Interface Design and Information Visualization

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Graphical User Interface Design and Information Visualization
Modulverantwortliche	Axel Platz
Dozent(en)	Axel Platz
Vorkenntnisse	Grundsätzliche Kenntnisse von interaktiven Systemen und Software-Ergonomie
Arbeitsaufwand	150 Stunden, davon: 60 Stunden Präsenz und 90 Stunden Vor- und Nachbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung und praxisorientierte Übungen
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Erstellung von fortgeschrittenen Benutzeroberflächen erfordert informationstechnische, gestalterische und ergonomie-relevante Kenntnisse und Fähigkeiten. Diese Lehrveranstaltung ergänzt die gestalterischen Fähigkeiten.
Lernziel	In der Lehrveranstaltung soll die Fähigkeit zur Konzeption und Gestaltung graphischer Bedienoberflächen in Theorie und Praxis vermittelt und durch praxisorientierte Übungen vertieft werden.
Schlüsselqualifikation	Kreatives Problemlösen, praktische Erfahrung, Interdisziplinarität
Lehrinhalte	Usability / Ästhetik / User Experience, Mensch-Maschine Interaktion, Kommunikationstheorie, Nutzungskontexterhebung (Analyse), Konzeption und visuelle Gestaltung (Synthese), Wahrnehmungsphysiologie und -psychologie (u.a. Gestaltgesetze, optische Täuschungen), Theorie des Bildes / Bildtypen und Bildfunktionen (insbesondere Strukturprojektionen), Visuelle Sprache (Expressivität von Formen und Farben), Anschauliches Denken (Theorie anschaulichen Denkens bei Rudolf Arnheim)
Literatur	Florin, Alexander: User Interface Design Ware, Colin: Information Visualization Arnheim, Rudolf: Visual Thinking, University of California Press, 1969
Leistungsnachweis	1/3 Referat (20 Minuten) und 2/3 Studienarbeit (Konzeption eines User Interface)

Grundlagen Content-Management-Systeme

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Grundlagen Content-Management-Systeme
Modulverantwortliche	Alexander Kröner
Dozent(en)	Alexander Kröner
Vorkenntnisse	Erforderlich <ul style="list-style-type: none">• Fach "Programmieren II"• Fach "Web-Programmierung" (oder äquivalente Fähigkeiten in HTML, CSS, JavaScript, PHP) Empfohlen <ul style="list-style-type: none">• Grundlagen SQL (z.B. Fach "Datenbanken")
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit 85 Stunden Vor- und Nachbereitung, Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung, Projektarbeit
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die Entwicklung von Erweiterungen für Web-Content-Management-Systeme stärkt einerseits praktische Fähigkeiten im Bereich der Web-Programmierung, und eröffnet zugleich einen Zugang zu den diesen populären Anwendungen zugrundeliegenden Softwarearchitekturen.
Lernziel	Kenntnis charakteristischer Merkmale und der Struktur von Content-Management-Systemen (CMS). Anwendung dieses Wissens, um ein Anwendungsszenario hinsichtlich funktionaler Anforderungen an ein CMS zu analysieren. Fähigkeit, Software-Komponenten zur Umsetzung der gewünschten Funktionalität zu konzipieren, implementieren und in eine bestehende Softwarearchitektur zu integrieren.
Schlüsselqualifikation	Implementierung von Content Management Systemen, Softwarearchitekturen Web-basierter Systeme, Softwareentwicklung in einer heterogenen Technologielandschaft.
Lehrinhalte	Content-Management-Systeme (CMS) sind in einer zunehmend durch digitale Daten bestimmten Medienwelt ein zentrales Werkzeug zur Steuerung von Erstellung, Organisation, und Präsentation von Inhalten. Schwerpunkt der Vorlesung bilden Web-basierte CMS. Behandelt u.a.:

- Grundlagen: Features, Klassifizierung, Auswahlkriterien, Architekturkonzepte, Content-Lifecycle, Asset-Management
- CMS erweitern: Frontend und Backend erweitern, Plugins entwickeln und einbinden, Personalisierung, Internationalisierung und Lokalisierung, Integration strukturierter Daten
- Schnittstellen: Import und Export von Content, Content Syndication, Client-side Scripting

Praktische Arbeiten erfolgen mehrheitlich auf Basis von PHP sowie dem CMS WordPress.

Literatur

- B. Boiko (2004). Content Management Bible. 2te Auflage. John Wiley & Sons.

Leistungsnachweis

Seminarleistung bestehend aus:

- 10% Referat (10 Min.)
- 90% Studienarbeit (Softwareanwendung entsprechend 75 Stunden Entwicklungsaufwand)

Grundlagen der Computerspielentwicklung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Grundlagen der Computerspielentwicklung
Modulverantwortliche	Bartosz von Rymon Lipinski
Dozent(en)	Bartosz von Rymon Lipinski
Vorkenntnisse	Programmieren I und II, Algorithmen und Datenstrukturen, Computergraphik. Vorteilhaft: Gestaltungs- und Medienlehre I und II, Softwarearchitektur, Software Engineering.
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbearbeitung, Übungsaufgaben und Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung, Seminar, Projektarbeit
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung von Kenntnissen aus der praktischen und angewandten Informatik im Anwendungsfeld Computerspiele. Zugang zum interdisziplinären Gebiet, in dem Softwareentwicklung und Gestaltung interaktiver Systeme zusammenwirken. Befähigung komplexe computergraphische Anwendungen zu verstehen, konzipieren und herzustellen.
Lernziel	Kompetenz die technischen sowie kreativen elementaren Rahmenbedingungen bei der Entwicklung einer interaktiven virtuellen Welt zu erläutern und zu bewerten. Grundlegende Kenntnisse hinsichtlich der Planung und Konzeption eines Computerspiels. Fähigkeiten zur Auswahl und Beurteilung geeigneter Softwarewerkzeuge und darauf basierend zur prototypischen Umsetzung.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, selbständiges Arbeiten, Kommunikations- und Teamfähigkeit, Interdisziplinarität.
Lehrinhalte	1. Einführung und Geschichte der Computerspiele 2. Planung einer Computerspielproduktion 3. Grundlagen des (technischen) „Game Designs“ 4. Technische Architektur von Computerspielen - Einordnung 5. Überblick und Einordnung bzgl. „Game Engines“ 6. Planung, Konzeption und Umsetzung eines Spielprototypen
Literatur	<ul style="list-style-type: none">• Fullerton, T. J.: Game Design Workshop, Apple Academic Press Inc., 2013• Schell, J.: The Art of Game Design - A Book of Lenses, Morgan Kaufmann, 2014• Rabin, S.: Introduction to Game Development, Cengage Delmar, 2009• Gregory, J.: Game Engine Architecture, Taylor & Francis Ltd., 2014• Nystrom, R.: Game Programming Patterns, Genever Benning, 2014

- McShaffry, M., Graham, D.: Game Coding Complete, Cengage Learning, 2012
- Novak, J.: Game Development Essentials, Delmar Cengage Learning, 2011

Leistungsnachweis

Praktische Studienarbeit (Kommentiertes Software-Produkt, entsprechend 85 Stunden Entwicklungsaufwand).

Grundlagen des maschinellen Lernens

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Grundlagen des maschinellen Lernens
Modulverantwortliche	Tobias Bocklet
Dozent(en)	Tobias Bocklet
Vorkenntnisse	Keine
Arbeitsaufwand	150h (60h Präsenzzeit, 90h Eigenstudium)
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS seminaristischer Unterricht mit Vorlesung und Übungen
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	
Lernziel	Die Studierenden kennen und verstehen die theoretischen und algorithmischen Grundlagen des maschinellen Lernens. Sie können Probleme aus dem Bereich des Maschinellen Lernens erkennen und selbständig lösen.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Fähigkeit zu selbständigem Arbeiten, Analyse und Klassifikation von Problemen, kreatives Problemlösen
Lehrinhalte	<ul style="list-style-type: none">• Grundlegende Prinzipien und Theorien des Maschinellen Lernens• Einführung der wichtigsten algorithmischen Konzepte• Lernen aus Daten• Praktische Anwendungen
Literatur	<ul style="list-style-type: none">• Goodfellow, I and Bengio, Y and Courville, A: <i>Deep Learning</i>. 2016• Niemann, H: <i>Klassifikation von Mustern</i>. 2. Überarbeitete Auflage, 2003
Leistungsnachweis	Schriftliche Prüfung (90min) am Veranstaltungsende
Hilfsmittel	
Zulassungsvoraussetzung	

Grundlagen des Softwaretests

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Grundlagen des Softwaretests
Modulverantwortliche	Ricarda Dormeyer
Dozent(en)	Ricarda Dormeyer
Vorkenntnisse	Zu Test & Qualitätssicherung: keine. Programmierkenntnisse sind wünschenswert, ebenso Grundkenntnisse zu Vorgehensmodellen und Software Engineering.
Arbeitsaufwand	150h, davon: 65h Präsenz, 85h Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Blockveranstaltung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Weil Softwarefehler schwerwiegende Folgen haben können, muss jede Software vor ihrem Einsatz getestet werden. Deshalb sind Grundkenntnisse über systematisches Testen von Software für Informatiker wichtig.
Lernziel	Erlernen grundlegender Konzepte, Begriffe und Methoden zu Test und Qualitätssicherung von Software
Schlüsselqualifikation	Test und Qualitätssicherung von Software (Komponenten, Systemen, Systemlandschaften), Testprozesse und ihre Integration in Softwareentwicklungsprozesse, Testmethoden, Testarten, Testrollen, Testwerkzeuge
Lehrinhalte	<p>Die Veranstaltung deckt die Inhalte des ISTQB Certified Tester Foundation Level ab. Dabei handelt es sich um ein seit vielen Jahren weltweit anerkanntes und etabliertes Aus- und Weiterbildungsschema für Softwaretester.</p> <p>Themen sind insbesondere:</p> <ul style="list-style-type: none">• Motivation des Softwaretests• Grundbegriffe• Fundamentaler Testprozess• Teststufen und Testarten• Statische Qualitätssicherung• Testbarkeit von Requirements• Testmethoden (Black Box, White Box, erfahrungsbasiert)• Konzeption und Planung von Tests• Wirtschaftlichkeit des Testens• Testrollen• Kategorien von Testwerkzeugen.

Literatur

Spillner, Linz: Basiswissen Softwaretest. dpunkt Verlag, 5. Auflage

Ergänzend/vertiefend:

Spillner, Roßner, Winter, Linz: Praxiswissen Softwaretest - Testmanagement. dpunkt Verlag, 3. Auflage

Roßner, Brandes, Götz, Winter: Basiswissen Modellbasierter Test. dpunkt Verlag.

Crispin, Gregory: Agile Testing. Addison-Wesley

Seidl, Baumgartner, Bucsics: Basiswissen Testautomatisierung. dpunkt Verlag

Leistungsnachweis

Mündliche Prüfung im Anschluss an die Blockveranstaltung (20 min)

In-Memory Computing am Beispiel von SAP HANA

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	In-Memory Computing am Beispiel von SAP HANA
Modulverantwortliche	Rainer Weber
Dozent(en)	Rainer Weber
Vorkenntnisse	Kenntnisse über Datenbanksysteme, insbesondere von SQL. Programmierkenntnisse in einer Programmiersprache (nicht notwendigerweise der im Kurs verwendeten). SAP-Erfahrung ist nicht erforderlich.
Arbeitsaufwand	75 Stunden, davon: 30 Präsenzzeit, 45 Stunden Vor- und Nachbereitung des Lehrstoffs
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Lehrveranstaltung mit Vorlesungs- und Übungsanteilen
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Die erworbenen Kenntnisse und Methoden sind nützlich für Aufgaben im Bereich „Business Intelligence“ und Anwendungsentwicklung.
Lernziel	Kenntnis des Potentials von In-Memory Computing für Unternehmenssoftware. Fähigkeit, In-Memory-Anwendungen durch Modellierung und durch Softwareentwicklung zu erstellen.
Schlüsselqualifikation	Fähigkeit zum Verständnis großer komplexer Systeme, Lernen am Beispiel, Transferfähigkeit, algorithmisches Denken, Teamfähigkeit
Lehrinhalte	<p>Der Kurs befasst sich mit einem frischen Thema: In-Memory Computing, sowohl im Allgemeinen wie auch speziell am Beispiel des recht neuen Systems SAP HANA.</p> <p>Bei In-Memory-Computing wird die vollständige Datenbank im Hauptspeicher gehalten statt auf einer Magnetplatte. Es wird ermöglicht durch immer größere und billigere Hauptspeicher und die parallele Bearbeitung durch Multicore-Prozessoren. Dadurch lassen sich Antwortzeiten komplexerer analytischer Anwendungen vom Minutenbereich in den Sekundenbereich bringen („sub-second response time“). Bestimmte Auswertungen sowie neuartige Anwendungen werden dadurch erst möglich.</p> <p>Der Kurs befasst sich vor allem mit der Architektur von Anwendungen, die das Potential von In-Memory Computing nutzen. Ist traditionell die Datenbank der Performanzengpass eines betrieblichen Anwendungssystems, so läuft die Anwendungslogik in einem In-Memory System umgekehrt möglichst nahe an der Datenbank.</p> <p>In dem Kurs werden die Konzepte und die Architektur von In-Memory Systemen geschildert (z.B. die spaltenorientierte Speicherung). Der wesentliche Teil ist allerdings</p>

die Nutzung von In-Memory Systemen, insbesondere an praktischen Übungen mit SAP HANA. Zum einen werden analytische Anwendungen durch Modellierung (ohne Programmierung) erstellt, die Auswertungen geschehen mit Business-Intelligence-Werkzeugen. Hierbei erfolgt auch eine Einführung in die Begriffe und Methoden von Business Intelligence und Data Warehousing. Zum anderen werden Anwendungen programmiert. In der Lehrveranstaltung wird dies im wesentlichen die Programmierung von Stored Procedures mit der Programmiersprache SQLScript sein. Ergänzende Programmiertechnologien werden kürzer angesprochen: REST-basierte Web-Services und serverseitiges JavaScript für die schmale Applikationsserverschicht und clientseitiges JavaScript und SAPUI5-Technologie für die breitere Client-seitige Präsentationsschicht.

Geplant ist auch, auf Beispiele für neuartige Anwendungen einzugehen.

Literatur

H. Plattner, G. Zeier: In-Memory Data Management: Technology and Applications. 2. Auflage. Springer, Berlin Heidelberg New York, 2012
B. Berg, P. Silvia: Einführung in SAP HANA. 2. Auflage. Galileo Press, Bonn, 2013
T. Schneider, E. Westenberg, H. Gahm: ABAP Entwicklung für SAP HANA. Galileo Press, Bonn, 2013

Leistungsnachweis

Mündliche Prüfung (Gruppenprüfung, bei zwei Personen 30 Minuten)

Lean Startup

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Lean Startup
Modulverantwortliche	Korbinian Riedhammer
Dozent(en)	Korbinian Riedhammer
Vorkenntnisse	Keine
Arbeitsaufwand	150h (60h Präsenzzeit, 90h Eigenstudium)
Leistungspunkte	5
Semesterwochenstunden	4
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Systematische Erarbeitung von technischen wie wirtschaftlichen Anforderungen
Lernziel	Die Studierenden erlernen die Lean Innovation und Lean Startup Methoden und können diese auf Problemstellungen im IT Umfeld anwenden. Sie können Innovationen aus technischer und unternehmerischer Sicht bewerten und mit geeigneten Werkzeugen prototypische Lösungen realisieren.
Schlüsselqualifikation	Problemanalyse, unternehmerisches Denken, agile Methoden
Lehrinhalte	<p>Lean Innovation ist eine Methode zur Steigerung der Effizienz durch das frühe und konsequente Einbeziehen von Benutzer- bzw. Kundenfeedback. Eine Vorgehensweise zur strukturierten Generierung und Entwicklung von Ideen ist hierbei Design Thinking, bei dem Lösungen iterativ und in enger Abstimmung mit den Benutzern erfolgt. Lean Startup ist eine wissenschaftliche Herangehensweise zur Ausarbeitung von Geschäftsideen welche auf der strukturierten Definition und experimentellen Validierung oder Widerlegung von Hypothesen beruht. Dazu wird Schritt für Schritt ein Business Model Canvas (BMC) erstellt, bei dem das Wertversprechen (Value Proposition) im Vordergrund steht. Die folgenden Methoden und Aktivitäten werden erarbeitet und angewendet:</p> <ul style="list-style-type: none">• Lean Innovation und Design Thinking• Lean Startup• Business Model Canvas• Customer Development• Zielgerichtete Interviews: Gesprächsführung und -beobachtung• Pitchen: Ideen erfolgreich präsentieren
Literatur	<ul style="list-style-type: none">• Steve Blank. The Startup Owner's Manual: The Step-By-Step Guide for Building a Great Company• Alexander Osterwalder, Yves Pigneur. Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers• Eric Ries. The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses• Frank Rimalovski, Giff Constable, Tom Fishburne. Talking to Humans: Success starts with understanding your customers

Leistungsnachweis	Semesterbegleitende Projektarbeit mit 12 wöchentlichen Milestones (je max. 10 Punkte) und abschließender Präsentation (max. 20 Punkte). Die Note wird aus der Gesamtpunktzahl (max. 140) bestimmt.
Hilfsmittel	2
Zulassungsvoraussetzung	Keine

Logik

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Logik
Modulverantwortliche	Yvonne Stry, Peter Trommler
Dozent(en)	Yvonne Stry
Vorkenntnisse	Mathematik, Theoretische Informatik
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit 30 Stunden Vor- und Nachbereiten des Vorlesungsstoffes 55 Stunden Vorbereiten der Präsentation
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung und Seminar
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Theoretischer Hintergrund
Lernziel	<ul style="list-style-type: none">• Einführung in die Aussagenlogik und in die Prädikatenlogik• Studieren der Entwicklung der Logik und ihrer zahlreichen Zweige
Schlüsselqualifikation	Präsentationskompetenz
Lehrinhalte	Vorlesung: Aussagenlogik, Prädikatenlogik Vortragsthemen: Geschichte der Logik, spezielle Logiken, Methoden des logischen Schließens, formale Methoden der Softwareentwicklung, Künstliche Intelligenz, automatisches Beweisen etc.
Literatur	Uwe Schöning, Logik für Informatiker, Spektrum Akademischer Verlag
Leistungsnachweis	Seminarvortrag 25 min

NoSQL-Datenbanken

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	NoSQL-Datenbanken
Modulverantwortliche	Johannes Schildgen
Dozent(en)	Johannes Schildgen
Vorkenntnisse	Datenbanken, SQL
Arbeitsaufwand	
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung + Übung
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	
Lernziel	Die Studierenden sind in der Lage, die neuartigen Datenmodelle und Anfragesprachen von NoSQL-Datenbanken mit klassischen relationen Ansätzen zu vergleichen und sie in der Praxis einzusetzen.
Schlüsselqualifikation	
Lehrinhalte	Diese Vorlesung vermittelt die Grundlagen von NoSQL-Datenbanken sowie der Verarbeitung von großen Datenmengen (Big Data). Neben fundamentalen Technologien werden die verschiedenen Kategorien von NoSQL-Systemen anhand von bekannten Implementierungen präsentiert und in praktischen Übungen eingesetzt. In letzteren geht es um die Benutzung der NoSQL-Systeme sowie um die Entwicklung von Java-Anwendungen unter Verwendung der jeweiligen DBMS-APIs. - Big Data - Key-Value-Datenbanken (Redis,...) - Wide-Column-Stores (HBase,...) - Dokumentendatenbanken (MongoDB,...) - Graphdatenbanken (Neo4J,...) - MapReduce (Hadoop, HDFS,...) - PageRank und andere Algorithmen - Höhere Sprachen (Pig, Hive, ...) - CAP-Theorem - Konsistenzmodelle
Literatur	<ul style="list-style-type: none">• Stefan Edlich: NoSQL• Lena Wiese: Advanced Data Management• Johannes Schildgen: MongoDB kompakt• neo4j.com/books (kostenlose Bücher in Deutsch und Englisch)
Leistungsnachweis	Mündliche Prüfung (30min).
Hilfsmittel	1 keine Hilfsmittel
Zulassungsvoraussetzung	

Programmierung und Technologie betrieblicher Standardsoftware (SAP-Praktikum)

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Programmierung und Technologie betrieblicher Standardsoftware (SAP-Praktikum)
Modulverantwortliche	Rainer Weber
Vorkenntnisse	Programmierung (mindestens eine Programmiersprache), Datenbanksysteme, insbesondere SQL
Arbeitsaufwand	150 Stunden, davon: 65 Präsenz, 85 Stunden Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Üblicherweise Blockkurs mit intensivem Anteil von praktischen Übungen. Im Wintersemester 2017/2018 nicht als Blockkurs, sondern während der regulären Vorlesungszeit.
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Die erworbenen Kenntnisse und Methoden befähigen zur Übernahme von Aufgaben in der Anwendungsentwicklung und im Informationsmanagement (Entwicklung, Erweiterung und Anpassung von betrieblicher Standardsoftware).
Lernziel	Die spezifischen Methoden und Techniken zur Erstellung betrieblicher Standardsoftware kennenlernen Fähigkeit, kleinere Anwendungsprogramme in der Programmiersprache ABAP erstellen zu können
Schlüsselqualifikation	Fähigkeit zum Verständnis großer komplexer Systeme, Lernen am Beispiel, Transferfähigkeit, algorithmisches Denken, Teamfähigkeit
Lehrinhalte	<ul style="list-style-type: none">• Programmierung betrieblicher Standardsoftware am Beispiel von SAP-Software in der Programmiersprache ABAP<ul style="list-style-type: none">◦ Grundlegende ABAP-Sprachelemente◦ Datenbankzugriffe, Data Dictionary◦ ABAP Objects◦ Web Dynpro für ABAP◦ Business Add-ins• Ausgewählte Aspekte der Technologie von Unternehmenssoftware
Literatur	H. Keller, S. Krüger: ABAP Objects. Galileo Press, 3. Auflage, 2006 R. Weber: Technologie von Unternehmenssoftware. Berlin – Heidelberg (Springer), 2012
Leistungsnachweis	Mündliche Prüfung (Gruppenprüfung, bei zwei Personen 30 Minuten)

Selbstlokalisierung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Selbstlokalisierung
Modulverantwortliche	Jörg Roth
Dozent(en)	Jörg Roth
Vorkenntnisse	Programmieren, Grundlagen der Informatik
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 65 Stunden praktische Studienarbeit, 20 Stunden für Prüfung und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit Praktikum
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Kenntnisse über Lösungsstrategien bei der Auswertung von fehlerbehafteten oder unsicheren Informationen, Kenntnisse der Programmierung auf Plattformen mit reduzierten Ressourcen, Fachbezogene Englisch-Kenntnisse.
Lernziel	Bei Abschluss des Lernprozesses wird der erfolgreiche Studierende in der Lage sein, Mechanismen zur Positionsbestimmung auf der Basis von Sensorinformationen zu kennen, Lösungsstrategien zur Fehlerbehebung und Interpretation von Lokationssensoren anwenden zu können und eine Anwendung aus dem Bereich der Selbstlokalisierung prototypisch entwickeln zu können.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Analyse und Klassifikation von Problemen und Identifikation von Lösungen.
Lehrinhalte	<p>Für viele neue Anwendungen ist es notwendig, die Position eines Benutzers oder eines mobilen Objektes zu ermitteln. Neben Verfahren zur absoluten Positionsbestimmung erlauben einige Sensoren (z. B. Odometer, Beschleunigungssensor, Gyroskopsensor) die relative Positionsbestimmung oder messen bestimmte Lageinformationen wie die Ausrichtung. Kann sich das Objekt aus eigenem Antrieb kontrolliert fortbewegen (z. B. im Fall von fahrerlosen Transportsystemen), erlaubt die Bewegungsansteuerung gewünschte räumliche Positionen aktiv anzusteuern.</p> <p>In dieser Veranstaltung werden folgende Themen behandelt:</p> <ul style="list-style-type: none">• Grundlagen der Positionsbestimmung• Positions- und Lagesensoren• Bilderkennung• Sensordatenfusion• Karten und Umweltmodelle• Navigation und Bahnplanung• Antriebsarten <p>Bestandteil dieser Veranstaltung ist eine Projektarbeit. Hierzu werden Aufgaben aus</p>

den oben genannten Themen in Kleingruppen bearbeitet.

Literatur

Kursunterlagen

Leistungsnachweis

Projektarbeit im Umfang von 65 Stunden, mündliche Befragung (20 min), jeweils gewichtet mit 50%

Software-Zuverlässigkeit: Theorie und Praxis

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Software-Zuverlässigkeit: Theorie und Praxis
Modulverantwortliche	Harald Stieber
Dozent(en)	Harald Stieber
Vorkenntnisse	Mathematik III (Statistik)
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden Vor- und Nachbearbeitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Testen ist ein wesentlicher Teil des Software-Lebenszyklus. Software-Zuverlässigkeit ist ein wichtiger Aspekt der Software-Qualität.
Lernziel	Kenntnis der Grundprinzipien des Software-Tests. Fähigkeit Fehlerdaten zu sammeln und mit geeigneten mathematischen Modellen die Software-Zuverlässigkeit abzuschätzen und zu verbessern.
Schlüsselqualifikation	Logisches Denkvermögen
Lehrinhalte	<ul style="list-style-type: none">• Testmethodik• Benutzungsprofile• Testen nach Benutzungsprofil• statistische Modelle zur Vorhersage der Software-Zuverlässigkeit• Methoden zur Überprüfung der Vorhersagegenauigkeit• Optimierung des Testprozesses
Literatur	<ul style="list-style-type: none">• M. Lyu (Herausgeber): Handbook of Software Reliability Engineering, McGraw-Hill• A. Mathur: Foundations of Software Testing, Addison-Wesley• Musa, Iannino, Okumoto: Software Reliability - Measurement, Prediction, Application, McGraw-Hill (1987)
Leistungsnachweis	Schriftlich, 60 Minuten
Hilfsmittel	
Zulassungsvoraussetzung	

Technical Computing

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Technical Computing
Modulverantwortliche	Oliver Kluge
Dozent(en)	Oliver Kluge
Vorkenntnisse	Programmiererfahrung
Arbeitsaufwand	150 Stunden, davon 65 Stunden Präsenzzeit + 85 Stunden Vor- und Nachbereitung, Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung und Übung
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vertiefung der Theorie des Technical Computing und Anwendung in der Praxis
Lernziel	Ob bei der statistischen Analyse von Messwerten, der Berechnung des Amplitudenspektrums oder der Filterung im Zeit- und Frequenzbereich, Werkzeuge wie MatLab & Co bieten dem Entwickler jede Menge Unterstützung an. Aber wie funktionieren sie? Der Kurs befähigt die Studierenden die zugrundeliegenden Algorithmen zu analysieren, deren Leistungsfähigkeit zu beurteilen und eigene Lösungen zu entwickeln. Weitere Informationen: https://www.technical-computing.de
Schlüsselqualifikation	Analytisches Denken, Abstraktionsvermögen, Fähigkeit zum Verständnis komplexer Systeme
Lehrinhalte	Algorithmen der digitalen Signalverarbeitung (DFT, IDFT, FFT, STFT, Filter im Zeit- und Frequenzbereich), Algorithmen der statistischen Analyse (Korrelation, Regression, grafische Verfahren)
Literatur	https://www.technical-computing.de/references.html
Leistungsnachweis	Mündliche Prüfung (30 Minuten)
Hilfsmittel	
Zulassungsvoraussetzung	

Text Analytics

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Text Analytics
Modulverantwortliche	Jens Albrecht
Dozent(en)	Jens Albrecht
Vorkenntnisse	Wünschenswert: Datenbanken, Programmieren 1 und 2
Arbeitsaufwand	150 Stunden, davon: 50 Stunden Präsenzzeit 100 Stunden Vor- und Nachbereitung der Vorlesung, Abschlussprojekt und Abschlussprüfung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminaristischer Unterricht mit praktischen Übungen. Die notwendigen theoretischen Konzepte werden in der Vorlesung gemeinsam erarbeitet und im Übungsteil exemplarisch in umgesetzt bzw. unter Verwendung existierender Frameworks angewandt. Es wird mit Python gearbeitet.
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums Lernziel	Nach der Teilnahme an diesem Modul sind die Studierenden in der Lage <ul style="list-style-type: none">• wichtige Begriffe im Bereich des Natural Language Processing zu definieren• überwachte und unüberwachte maschinelle Lernverfahren für die Analyse unstrukturierter, natürlichsprachlicher Texte zu erklären, zu bewerten und einzusetzen• Textdaten mit Python systematisch aufzubereiten, zu visualisieren und zu analysieren• eigenständig Lösungsansätze für analytische Fragestellungen zu natürlichsprachlichen Texten zu entwickeln• Machine-Learning-Prozesse in Python umzusetzen•
Schlüsselqualifikation	Fähigkeit zu selbständigem Lernen, komplexem Denken, Abstraktions- und Transferfähigkeit, analytische Kompetenz, Teamfähigkeit
Lehrinhalte	Ein Großteil der digital verfügbaren Information liegt in Form unstrukturierter Texte vor, und diese Datenmenge wächst täglich. Aufgrund der Eigenheiten natürlicher Sprache erfordert die automatisierte Erschließung von Web-Inhalten, Nutzer-Kommentaren, Emails oder digitalisierten Gesprächsprotokollen besondere Verfahren und Algorithmen. Diese kommen beispielsweise in Suchmaschinen, Spam-Filtern oder bei

der Verbrechensbekämpfung zum Einsatz. Ziel der Veranstaltung ist es, die Studierenden mit grundlegenden Methoden zum Auffinden von Information (Information Retrieval), zur automatisierten Analyse von Textinhalten (Text Mining) und zur Extraktion von Wissen (Knowledge Acquisition) vertraut zu machen. Im Einzelnen wird auf folgende Aspekte eingegangen:

- Grundlagen des Natural Language Processing (NLP)
 - Vagheit in der Sprache
 - Aufbereitung von Texten
 - Part-of-Speech-Tagging
 - Named Entity Recognition
- Information Retrieval
 - Vektorraum-Modell
 - Ähnlichkeit von Dokumenten
 - Arbeitsweise von Suchmaschinen
- Text Mining und Machine Learning auf Texten
 - Topic-Analyse und Clustering
 - Klassifikation
 - Meinungs- und Stimmungsanalyse
 - Kontextuelle Ähnlichkeitsanalyse mit Word Embeddings

Literatur

Bengfort, Benjamin ; Bilbro, Rebecca ; Ojeda, Tony: Applied Text Analysis with Python. O'Reilly, 2018.

Bird, Steven ; Klein, Ewan ; Loper, Edward: Natural Language Processing with Python. O'Reilly, 2009. Online frei verfügbar unter <http://www.nltk.org/book/>

Henrich, Andreas: Information Retrieval 1. Universität Bamberg, Lehrstuhl für Medieninformatik, 2008. Online frei verfügbar unter <https://www.uni-bamberg.de/minf/ir1-buch/>

Manning, Christopher D. ; Raghavan, Prabhakar ; Schütze, Hinrich: An Introduction to Information Retrieval. Cambridge University Press, 2009. Online frei verfügbar unter <https://nlp.stanford.edu/IR-book/>

Raschka, Sebastian: Python Machine Learning. Packt Publishing, 2015. Online im Campusnetz verfügbar unter <http://proquest.tech.safaribooksonline.de/9781783555130> (auf Deutsch bei MITP Verlag erschienen, aber nicht online verfügbar)

VanderPlas, Jake: Python Data Science Handbook. O'Reilly, 2016. Komplett als Jupyter-Notebooks verfügbar unter <https://jakevdp.github.io/PythonDataScienceHandbook/>

Zhai, ChengXiang ; Massung, Sean: Text Data Management and Analysis. ACM Books, 2016. (in Bib. und IN-Präsenzbib.)

Leistungsnachweis

Studienarbeit in Form eines Abschlussprojekts mit Vortrag (Aufwand ca. 40h) und mündliche Befragung (15 min), Gewichtung jeweils 50%.

Hilfsmittel

Zulassungsvoraussetzung

Verteilte Systeme - Grundlagen und Algorithmen

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Verteilte Systeme - Grundlagen und Algorithmen
Modulverantwortliche	Michael Zapf
Dozent(en)	Michael Zapf
Vorkenntnisse	Rechnerkommunikation
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz 30 Stunden Vor- und Nachbereitung des Lehrstoffs 40 Stunden praktische Übungen 5 Stunden für Prüfung und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	4 SWS seminaristischer Unterricht mit Vorlesung und Übungen
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	<p>Die Vernetzung von Rechnersystemen ist nur ein erster Schritt, um Ressourcen standortübergreifend nutzbar zu machen. Jenseits des einfachen Nachrichtenaustauschs lassen sich diese Rechner koordinieren, um verteilte Dienste zu realisieren; Operationen werden über das Netz ausgelöst, Ergebnisse zurückgeliefert. Dies kann so weit gehen, dass Anwendungen über mehrere Rechnersysteme verteilt zur Ausführung kommen, wobei die Verteilung vollständig transparent wird. Dies zeigt nicht nur Vorteile in der Performanz, sondern auch in der Ausfallsicherheit und in der effizienten Nutzung der verfügbaren Ressourcen.</p> <p>Das Konzept der Verteilung äußert sich nicht nur im Großen, wenn komplette Computersysteme betrachtet werden, sondern auch im Kleinen, insbesondere im Entwurf paralleler Architekturen, wenn Rechenkerne in großer Zahl zu Hochleistungsrechnern zusammengefügt werden. Die Effekte der weiträumigen verteilten Systeme, betreffend die Konsistenz von Datenspeichern, konkurrente Zugriffe, Verklemmungen und Abstimmungsprozesse, sind in sehr ähnlicher Form in diesen Multiprozessor-Architekturen wiederzufinden.</p>
Lernziel	Analyse typischer Strukturen bei verteilten Systemen; Verstehen der besonderen Herausforderungen im Vergleich zu einfachen lokalen Systemen insbesondere in Bezug auf Synchronisation, Konsistenz, Effekte durch uneinheitliche Nachrichtenlaufzeiten; Verstehen, Anwenden und Bewerten diverser Verfahren zur Synchronisierung, Auswahl, Ausschluss und Terminierungserkennung in verteilten Systemen; Untersuchen des Laufzeitverhaltens von verteilten Systemen anhand von Kommunikationsdiagrammen; Erkennung von Systemverklemmungen; Anwenden von Techniken zu deren Vermeidung insbesondere im verteilten Falle, Bewerten der Robustheit verteilter Anwendungen; Verständnis der Erfordernis und Wirkung von Konsistenzgarantien; Auswahl einer für gegebene Szenarien angemessenen Garantie; Vergleichen der Techniken und typischer Szenarien für Peer-to-Peer-Netze.

Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Analyse und Klassifikation von Problemen. Fähigkeit zur Modellierung von komplexen, verteilten Computersystemen mithilfe typischer Metamodelle und Analyse des Verhaltens eines Systems
Lehrinhalte	<ul style="list-style-type: none"> • Was sind verteilte Systeme? • Multicast-Konsistenz • Logische Uhren • Physikalische Uhren • Wechselseitiger Ausschluss • Auswahlverfahren • Deadlocks • Terminierung verteilter Anwendungen • Fehlertolerante verteilte Systeme • Verteilter Speicher und Konsistenzmodelle • Optimistische Replikation • Peer-to-Peer-Architekturen
Literatur	<p>Tanenbaum, Andrew S. ; van Steen, Maarten: Verteilte Systeme. 2., Aufl. : Pearson Studium, 2007. - ISBN 3827372933</p> <p>Friedemann Mattern: Verteilte Basisalgorithmen. Springer-Verlag, ISBN 3-540-51835-5, 1989</p>
Leistungsnachweis	Schriftliche Prüfung (Klausur, 90 min) am Veranstaltungsende
Hilfsmittel	5(10), 7

Virtuelle Maschinen

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Virtuelle Maschinen
Modulverantwortliche	Ralf-Ulrich Kern
Dozent(en)	Ralf-Ulrich Kern
Vorkenntnisse	Rechnersysteme; unbedingt erforderlich: bestandene Prüfung in "Betriebssysteme"
Arbeitsaufwand	75 Stunden, davon: 30 Stunden Präsenzzeit, 45 Stunden für Vor- und Nachbereitung des Lehrstoffs, Vorbereitung des Seminarvortrags und schriftliche Ausarbeitung.
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Seminar
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung von Spezialkenntnissen im Fachgebiet Betriebssysteme, die für vielfältige Gebiete der Informatik nützlich sind.
Lernziel	Analyse der architektonischen Voraussetzungen, Vergleich verschiedener Typen und Implementierungsformen von virtuellen Maschinen und Fähigkeit, das Einsatzpotential von virtuellen Maschinen zu beurteilen.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, Präsentationskompetenz
Lehrinhalte	<p>Mit Virtualisierung im hier verwendeten Sinne ist die Replizierung (Vervielfältigung) realer Hardware gemeint. D. h., aus einem Rechner werden mittels einer Software-Schicht virtuell (d. h. scheinbar) mehrere Rechner ("virtuelle Maschinen") gemacht, und in jeder dieser virtuellen Maschinen kann zur selben Zeit ein eigenes Betriebssystem ablaufen, isoliert von allen anderen.</p> <p>Diese Technik ist seit ca. 1965 bekannt, als sie zunächst für IBM-Großrechner entwickelt wurde, und sie ist aus verschiedenen Gründen zu neuer Blüte gekommen, da es dafür viele Anwendungsmöglichkeiten sowohl auf dem Desktop als auch im Server-Bereich sowie in eingebetteten Systemen gibt.</p> <p>In diesem Seminar sollen abseits des Marketingrummels die technischen Grundlagen aus den Bereichen Prozessorarchitektur und Betriebssysteme erarbeitet werden. Ausgangspunkt ist die immer noch aktuelle Arbeit von Popek und Goldberg, in der Bedingungen für die effiziente Virtualisierbarkeit einer Prozessorarchitektur angegeben wurden. Da diese Bedingungen in der klassischen x86-Architektur von Intel und AMD nicht erfüllt waren, mussten für die Implementierung virtueller Maschinen Umgehungsmöglichkeiten gefunden werden, die auch behandelt werden. Inzwischen haben Intel und AMD jeweils eigene Virtualisierungserweiterungen der x86-Architektur entwickelt, die den Popek/Goldberg-Bedingungen genügen und die die</p>

Implementierung effizienter virtueller Maschinen vereinfachen.

Nach einer kurzen Vorlesungseinführung in das Thema werden in Vorträgen die historische Entwicklung, die technischen Grundlagen und auch eine Auswahl aus der laufend anwachsenden Menge von aktuellen Virtualisierungssystemen beleuchtet, wobei bei letzteren der Schwerpunkt auf die Varianten der methodischen Vorgehensweise im Zusammenhang mit dem jeweiligen Einsatzgebiet gelegt wird.

Stichwörter zum Inhalt:

Virtuelle Maschine, Virtualisierungssystem, Popek/Goldberg-Bedingungen

Literatur

- POPEK, G. J. ; GOLDBERG, R. P.: Formal Requirements for Virtualizable Third Generation Architectures.
In: Communications of the ACM (1974).
- ROBIN, J. S. ; IRVINE, C. E.: Analysis of the Intel Pentium's ability to Support a Secure Virtual Machine Monitor.
In: Proceedings of the 9th Usenix Symposium (2000).
- ADAMS, K. ; AGESEN, O. : A comparison of software and hardware techniques for x86 virtualization.
In: ASPLOS-XII: Proceedings of the 12th international conference on Architectural support for programming languages and operating systems. New York, NY, USA : ACM Press, 2006.

Leistungsnachweis

Referat (max. 45 min) und Studienarbeit (10 - 15 Seiten, Ausarbeitung des Referats); Gewichtung je 50 %. Anwesenheitspflicht während der Seminarvorträge

Visualisierung

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Visualisierung
Modulverantwortliche	Matthias Teßmann
Dozent(en)	Matthias Teßmann
Vorkenntnisse	Grundlagen der Informatik, Algorithmen und Datenstrukturen, Computergraphik, Programmierkenntnisse (C++, Python), Mathematik (insb. Lineare Algebra, Differential- & Integralrechnung)
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz 85 Stunden Vor- und Nachbereitung des Lehrstoffes sowie Übungsaufgaben
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung und Seminar
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Die Visualisierung von Daten aus technisch-wissenschaftlichen Experimenten oder Simulationen zum Zwecke des Erkenntnisgewinnes ist ein wichtiges Anwendungsgebiet der Informatik. Kenntnis der Methoden zur Verarbeitung und Darstellung dieser Daten können auf vielfältige Weise eingesetzt werden.
Lernziel	Analyse und Vergleich der grundlegenden Algorithmen zur Visualisierung von Daten aus technisch-wissenschaftlichen Experimenten. Bewertung der Anwendbarkeit verschiedener Verfahren in Bezug auf spezifische Fragestellungen. Entwicklung von Strategien zur Bearbeitung neuer Problemstellungen.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, selbständiges Arbeiten, Fähigkeit zu komplexem Denken, Fähigkeit zum Verständnis komplexer Systeme und Algorithmen, Lernen am Beispiel, Transferfähigkeit
Lehrinhalte	Die Visualisierung beschäftigt sich mit der visuellen Aufbereitung der (oft großen) Datenmengen aus technisch-wissenschaftlichen Experimenten oder Simulationen zum Zweck des tieferen Verständnisses und der einfacheren Präsentation komplexer Phänomene. Die Vorlesung gibt eine Einführung in die grundlegenden Algorithmen und Datenstrukturen, sowie einen Überblick über die verfügbaren Softwarewerkzeuge und verbreiteten Dateiformate. Behandelt werden u.a. folgende Themen: <ul style="list-style-type: none">• Visualisierungsszenarien• Gitterstrukturen und Interpolation• Verfahren für 2D Skalar- und Vektorfelder• Verfahren für 3D Skalar- und Vektorfelder

- Verfahren für multivariate Daten
- Volumenvisualisierung mit Isoflächen
- Direktes Volume-Rendering

Literatur

- W. Schroeder and K. Martin, "The Visualization Toolkit", Kitware Inc. 2004
- M. Ward, G.G. Grinstein and D. Keim, "Interactive Data Visualization: Foundations, Techniques, and Applications", Taylor & Francis, 2010
- C.D. Hansen and C.R. Johnson, "Visualization Handbook", Academic Press, 2004

Leistungsnachweis

Schriftliche Prüfung (90 Minuten, 100%), Seminarvortrag (20 Minuten) für die Zulassung zur Prüfung erforderlich

Hilfsmittel

Zulassungsvoraussetzung

Voice-over-IP

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer Schwerpunkt-FWPF
Modul	Voice-over-IP
Modulverantwortliche	Matthias Teßmann
Dozent(en)	Matthias Teßmann
Vorkenntnisse	Grundlagen der Informatik, Algorithmen und Datenstrukturen, Programmieren I & II (zusätzliche Kenntnisse in C und C++ wären von Vorteil), Rechnerkommunikation
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbereiten des Lehrstoffs sowie Übungsaufgaben
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminar mit Praktikum
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Voice-over-IP ist ein wichtiges Anwendungsgebiet der Informatik und mittlerweile die wichtigste Technologie für globale Sprachkommunikation
Lernziel	Identifikation der wesentlichen Einzelkomponenten eines komplexen Softwareprodukts am konkreten Beispiel einer Voice-over-IP Anwendung. Gegenüberstellung verschiedener Implementierungsansätze für die einzelnen Komponenten (z.B. Audio-Codecs, Jitter-Buffer, ...), sowie die Analyse der daraus resultierenden Auswirkungen auf das Gesamtsystem. Aufzeigen der Schwierigkeiten und Herausforderungen, die sich für den Betrieb eines Voice-over-IP-Systems in einem offenen Netzwerk ergeben. Fähigkeit zur Bewertung der Rolle freier Standards und Protokolle, die eine weitreichende Interoperabilität verschiedener Implementierungen erst ermöglichen. Sammeln praktischer Erfahrung durch die Implementierung einer konkreten Voice-over-IP Anwendung.
Schlüsselqualifikation	Analyse und Klassifikation von Problemen, kreatives Problemlösen, selbständiges Arbeiten, Fähigkeit zu komplexem Denken, Fähigkeit zum Verständnis großer komplexer Systeme, Lernen am Beispiel, Transferfähigkeit
Lehrinhalte	<ul style="list-style-type: none">• Einführung in Telekommunikationssysteme und -netze• Audiosignalverarbeitung: Sprache, Töne & Musik• Sprach- und Audiokodierung<ul style="list-style-type: none">◦ POTS, HD-Voice, Full-HD Voice• Standards und Standardisierungsprozesse• IP-Protokollfamilie: IP, TCP, UDP, SCTP• Das Real-Time Transport Protokoll (RTP)• IP-basierte Echtzeitkommunikation<ul style="list-style-type: none">◦ Herausforderungen: Delay, Jitter, Packet Loss, QoS

Literatur

- RFCs 3261, 3550, 3640, 4040, 4961 u.a. (<http://www.ietf.org/rfc.html>)
- R. Swale, D. Collins, "Carrier Grade Voice Over IP", McGraw-Hill Professional, 3rd Ed., 2013
- C. Perkins, "RTP: Audio and Video for the Internet", Adison Wesley, 2012
- D. Minoli, E. Minoli, "Delivering Voice over IP Networks", John Wiley & Sons, 2nd Ed., 2002

Leistungsnachweis

Referat (45 Minuten, 60%) und praktische Studienarbeit (kommentiertes Software-Produkt entsprechend 25 Stunden Entwicklungsaufwand, 40%). Während der Präsentationen besteht Anwesenheitspflicht.

Hilfsmittel

Zulassungsvoraussetzung

Betriebssysteme für einfache Mikrocontroller

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Betriebssysteme für einfache Mikrocontroller
Modulverantwortliche	Thomas Fischer
Dozent(en)	Thomas Fischer
Vorkenntnisse	Grundkenntnisse im Bereich der Rechnerarchitektur und Programmiersprachen
Arbeitsaufwand	75 Stunden, davon: 21 Stunden Präsenz, 14 Stunden zum Lösen der Übungsaufgaben 20 Stunden zum Lesen von Artikeln und Fachbüchern, 20 Stunden zur Nachbereitung und Prüfungsvorbereitung.
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Seminaristischer Unterricht mit Übungen
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Vertiefung von Kenntnissen und praktischen Fertigkeiten im Bereich angewandter Softwaretechnologie und Betriebssysteme speziell für eingebettete Systeme (Stichwort IoT)
Lernziel	<ul style="list-style-type: none">• Die AbsolventInnen sind in der Lage Echtzeit-Betriebssysteme für Mikrocontroller zu vergleichen und für eine gegebene Aufgabenstellung auszuwählen, da Sie mit der internen Funktionsweise vertraut sind.• Insbesondere erkennen Sie die Wechselwirkungen zwischen Hardware (speziell Interrupts) und dem eingesetzten Betriebssystem. Aus den vom Betriebssystem angebotenen Mechanismen zur Synchronisation und Koordination können Sie die geeigneten Funktionen auswählen.• Die AbsolventInnen dieser Lehrveranstaltung können unter Einbindung eines einfachen Betriebssystems Anwendungen für Mikrocontroller planen und implementieren.
Schlüsselqualifikation	Fähigkeit zu selbständigem Arbeiten, Fähigkeit zum Verständnis komplexer Systeme Analyse, Interdisziplinarität Fremdsprachenkenntnisse, insbesondere in Englisch -- die LV kann auf Wunsch/ bei Bedarf jederzeit in englischer Sprache angeboten werden! Unterlagen sind fast ausschließlich in englischer Sprache.
Lehrinhalte	Dieser Kurs ist eine Einführung in die Programmierung von Mikrocontrollern mit einfachen (Echtzeit-)Betriebssystemen. Das Thema wird als Überleitung von der Interrupt gesteuerten Programmierung eines Vorder-/Hintergrundsystems zu einer, durch ein Betriebssystem unterstützten, Anwendungssoftware erschlossen: <ul style="list-style-type: none">• Wie laufen Programme in Vorder-/Hintergrundsystemen ab?• Was bestimmt das Zeitverhalten?• Welchen Einfluss haben Interrupts?• Welchen Vorteil bietet die Verwendung eines Betriebssystems?

- Wie unterscheidet sich das Software Design?

Zu diesen Fragen werden folgende Themen vorgetragen bzw. diskutiert:

- Grundlagen und Aufgaben von Betriebssystemen, Geräteverwaltung, Scheduling, Parallelisierung und Prozesssynchronisation.

Im zweiten Teil wird nun der Begriff der Echtzeitsysteme erläutert.

- Welchen Zweck erfüllen Echtzeit-Betriebssysteme?
- Welche Probleme können auftreten, welche Mechanismen sollen diese Probleme verhindern?

Zu diesen Fragen werden folgende Themen vorgetragen bzw. diskutiert:

- Grundlagen der Echtzeitsysteme, Konzepte zu parallelen/ nebenläufigen Prozessen (Prozesskommunikation, Synchronisation, Concurrency).
- Fallstudie des Betriebssystems FreeRTOS und/oder C/OS-III.

In den Übungen werden Rechen- und Programmierbeispiele zum Thema Scheduling und Prozesssynchronisation und Inter Task Kommunikation (Mutex, Queues, Semaphore) an Hand des Betriebssystems FreeRTOS und/oder C/OS-III durchgeführt. Zum Einsatz kommen Entwicklungskits mit dem STM32 Mikrocontroller.

Literatur

Dieser Kurs basiert auf dem Textbuch von

- Barry, Richard: Using the FreeRTOS Real Time Kernel – a Practical Guide, eBook unter www.freertos.org

Weitere Empfohlene Bücher und Artikel sind in Auszügen:

- Kienzle, Friedrich: Programmierung von Echtzeitsystemen, Hanser Verlag 2009
- Labrosse, Jean J.: uC/OS-III – The Real Time Kernel, Micrium Press 2009

Leistungsnachweis

Schriftliche Prüfung (60%) und Übungsaufgaben (40%)

Dauer der schriftlichen Prüfung: 60 Minuten

Die Übungsaufgaben sind eine Projektaufgabe in der die Studierenden in einer Gruppe von 2-4 Personen eine Aufgabenstellung, durch Anwendung der in der Lehrveranstaltung erlernten Fähigkeiten/Kompetenzen, lösen müssen.

Für die Lösung dieser Projektaufgabe steht ein Zeitraum von 2-3 Wochen zur Verfügung.

Der Zeitaufwand pro Studierendem als Beitrag zur Lösung beträgt etwa 20 Stunden.

Die Präsentation und Abgabe erfolgt im Anschluß an die schriftliche Prüfung.

Datenbanken II in der Wirtschaftsinformatik

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Datenbanken II in der Wirtschaftsinformatik
Modulverantwortliche	Alfred Holl
Dozent(en)	Alfred Holl
Vorkenntnisse	Datenbanken
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden für Vorbereitung von Referat und Ausarbeitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminar (Bachelor)
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs in Deutsch und Englisch
Beitrag zu den Zielen des Studiums	Vertiefte Kenntnisse über Datenbanken spielen in vielen Berufsfeldern von Absolventen informatikorientierter Studiengänge eine wichtige Rolle.
Lernziel	Kenntnisse über Ziele und Methodiken von Datenbanken Fähigkeit, unterschiedliche Typen von Datenbanken in verschiedenen Anwendungsbereichen in der Wirtschaftsinformatik einzusetzen
Schlüsselqualifikation	Fähigkeit zur Analyse komplexer Probleme Umgang mit wissenschaftlicher Literatur technisch-wissenschaftliches Schreiben Präsentationsfähigkeiten
Lehrinhalte	Eine Auswahl der folgenden Themen: <ul style="list-style-type: none">• Datenbanken im Internet• temporale Datenbanken• Multimedia-Datenbanken• sqlite• Datenbanken und XML• NoSQL-Datenbanken• Graph-Datenbanken• Datenbanken und Datenschutz• Ontologien• Data Warehouses und Data Marts• Föderierte DB• objektrelationale DB• objektorientierte DB• ODBC-Schnittstelle• unformatierte DB (Dokumente) und automatische Indexierung• kommerzielle Datenbankanbieter• spezielle betriebswirtschaftliche Datenbankanwendungen

- Datenbanken in Medizin, Genetik, Biologie
- Datenbanken in Mathematik, Physik, Chemie
- Datenbanken in den Geisteswissenschaften, Digital Humanities
- Datenbanken in den Rechtswissenschaften
- SQL-Dialekte und SQL-Erweiterungen von DBMS-Produkten
- Open Source DB
- Recoverymaßnahmen
- Data-Dictionary-Systeme
- verteilte DB und Client-Server-Architekturen

Literatur

- Date, Christopher: Introduction to database systems. Reading, MA: Addison Wesley, 2003
- Kemper, A.; Eickler, A.: Datenbanksysteme. München: Oldenbourg, 2001
- Date, C.J.; Darwen, H.: A guide to the SQL standard. Reading, MA: Addison Wesley, 1997
- Pernul, Günter; Unland, Rainer: Datenbanken im Unternehmen. München: Oldenbourg, 2003

Leistungsnachweis

Referat (75 min, 50 %) und schriftliche Ausarbeitung (15 Seiten, 50 %) je Zweiergruppe.

Digitalisierung und Nachhaltigkeit

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Digitalisierung und Nachhaltigkeit
Modulverantwortliche	Dina Barbian
Dozent(en)	Dina Barbian
Vorkenntnisse	Grundlagen der Wirtschaftswissenschaften, Supply Chain Management
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz, 85 Stunden Vor- und Nachbereiten des Lehrstoffs, insb. Vorbereitung des Referats und Erstellen der Studienarbeit
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminaristischer Unterrichtsstil mit vielen Fallstudien
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	<ul style="list-style-type: none">• Identifikation der Möglichkeiten und Grenzen einer zunehmenden Digitalisierung für eine nachhaltige Entwicklung• Erlernen von informationstechnischen Lösungen zur Steigerung der Ressourcen- und Energieeffizienz• Kennenlernen von innovativen Anwendungssystemen zur Unterstützung des Menschen bei der Verrichtung von Arbeit
Lernziel	Die Studierenden können: <ul style="list-style-type: none">• die Begriffe "Digitalisierung" und "Nachhaltigkeit" erklären,• eine zunehmende Digitalisierung im Kontext von Nachhaltigkeit kritisch einordnen,• zu den gesellschaftlichen Herausforderungen informationstechnische Lösungen benennen,• wichtige Anwendungsfelder für eine Digitalisierung und zur Erreichung von Nachhaltigkeit identifizieren, und• die Risiken durch eine zunehmende Digitalisierung erläutern.
Schlüsselqualifikation	<ul style="list-style-type: none">• 40% Fachkompetenz• 30% Methodenkompetenz• 30% Sozialkompetenz
Lehrinhalte	<ol style="list-style-type: none">1. Grundlegende Begrifflichkeiten, Abgrenzung und Historie<ol style="list-style-type: none">1.1 Begriffe von Digitalisierung1.2 Entwicklung von Industrie 1.0 zu Industrie 4.01.3 Was ist Nachhaltigkeit?1.4 Zusammenhänge Ökonomie – Ökologie – Gesellschaft – Technologie1.5 Historie zu Nachhaltigkeit1.6 UN Sustainable Development Goals (SDGs) / UN-Entwicklungsziele1.7 Digitalisierung und Nachhaltigkeit: nachhaltig-digitale Organisationen

und Forderungen

2. Gesellschaftliche Herausforderungen
 - 2.1 Zunehmende Alterung in der Bevölkerung
 - 2.2 Fachkräftemangel
 - 2.3 Zunehmende Umweltverschmutzung, Klimawandel
 - 2.4 Schutz der kritischen Infrastrukturen
 - 2.5 Zunehmende Vernetzung: vernetzte Konsumgüter und Maschinen
3. Lösungsansätze
 - 3.1 Zunehmende Automatisierung als Lösung des Altersproblems?
 - 3.2 Einsatz von Robotik (z. B. als Hilfe in Altenheimen oder Hotels)
 - 3.3 Power-to-X-Technologien: Power-to-Gas (s. Audi e-Gas-Projekt)
 - 3.4 Leichtbauweise durch 3D-Druck zur Einsparung von Ressourcen
 - 3.5 Serviceroboter in Kaufhäusern und Banken; Sprachassistenten und Gesichtserkennung
 - 3.6 Abfallmanagementsysteme
 - 3.7 Modularer Aufbau elektronischer Produkte (siehe Fair- und ShiftPhone)
 - 3.8 Substitution nicht-erneuerbarer durch erneuerbare Ressourcen
 - 3.9 Sharing Economy: Car-sharing etc.
4. Ausgewählte Anwendungsfelder
 - 4.1 Zunehmender Einsatz von Drohnen, Sensorik, Robotik
 - 4.2 Smart Grid in der Energiewirtschaft: erneuerbare Energien
 - 4.3 Logistik und Transportsektor: eTrucks, Echtzeit-Nachverfolgung
 - 4.4 Industrielle Produktion: Predictive Maintenance, Smart Factory
5. Nutzeffekte
 - 5.1 Erhöhung der Ressourcen- und Energieeffizienz
 - 5.2 Assistenzen bei extremen Arbeitsbedingungen (Monotonie, Gefahren)
 - 5.3 Einsparung von Arbeitskräften, Rohstoffen und Zeit
6. Risiken durch eine zunehmende Digitalisierung
 - 6.1 Cyber-Kriminalität und Datensicherheit
 - 6.2 Schwierigkeiten bei der Normung und Rechtsrahmen
 - 6.3 Haftung für Fehler und Unfälle
 - 6.4 Zunehmende Arbeitslosigkeit unter Gering-Qualifizierten?

Literatur

Barbian, D., Ökonomie und Sustainable Development – Entwicklung eines Ansatzes zur Umsetzung von Nachhaltigkeit, Aachen 2001.

Mertens, P., Barbian, D. und Baier, S., Digitalisierung und Industrie 4.0 – Eine Relativierung, Berlin-Heidelberg 2017 (im Druck).

Barbian, D., Industrie 4.0 in der Lagerlogistik – Überblick, aktuelle Trends und Folgen für eine nachhaltige Entwicklung, in: Glock, C. und Grosse, E. (Hrsg.), Warehousing 4.0 – Technische Lösungen und Managementkonzepte für die Lagerlogistik der Zukunft, Lauda-Königshofen 2017, S. 17-36.

Barbian, D., Our common WASTE – solutions for a sustainable society, in: Plöhn, J. und Chobanov, G. (eds.), Sustainability and Welfare Policy in European Market Economies, Frankfurt am Main 2017, S. 127-145.

Barbian, D., Umweltmanagement – wozu?, Technik in Bayern 06/2016, S. 10-11.

Barbian, D., Cyber-Physical Systems - Can They Contribute to More Sustainability? in: Herzog, M. (ed.), Economics of communication: ICT driven fairness and sustainability for local and global marketplaces, Berlin 2015, S. 29-44.

Mertens, P. und Barbian, D., Beherrschung systemischer Risiken in weltweiten Netzen,

Informatik Spektrum 38 (2015) 4, S. 283-289.

Mertens, P. und Barbian, D., Die Wirtschaftsinformatik der Zukunft – auch eine Wissenschaft der Netze?, HMD – Praxis der Wirtschaftsinformatik, Heft 300 „Paradigmenwechsel“, Band 51, Heft 6, Dezember 2014, S. 729-743.

Leistungsnachweis

Die Seminarleistung besteht aus einem Referat (20 min., Gewichtung 50%) und einer Studienarbeit (20-30 Seiten, Gewichtung 50%).

Einsatz von Simulations-Software

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Einsatz von Simulations-Software
Modulverantwortliche	Wolfgang Bremer
Dozent(en)	Wolfgang Bremer
Vorkenntnisse	Grundlagen der Betriebswirtschaftslehre, insbesondere Interesse an Supply Chain Management, Grundkenntnisse in Statistik
Arbeitsaufwand	150 Stunden, davon 60 Stunden Präsenz sowie 90 Stunden Vor- und Nachbereiten des Lehrstoffes sowie Übungsaufgaben. Die Übungsaufgaben sind im Labor teilweise mit der Software zu lösen.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung + Übung im Labor (PC, Fallstudien)
Semesterturnus	Winter- und Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Stärkung des Abstraktionsvermögens. Ausbilden analytischer Fähigkeiten.
Lernziel	Kenntnis erlangen in den Instrumenten: Simulation und Modellierung. Befähigt sein, Supply-Chain und Serviceprozesse im Computermodell abzubilden, deren Verhalten quantitativ im Vorfeld zu ermitteln und die Ergebnisse zu interpretieren.
Schlüsselqualifikation	Fähigkeit zu selbständigem Arbeiten. Abstraktion und Modellbildung. Analyse und Modellierung von Systemen. Kreatives Problemlösen
Lehrinhalte	1. Grundlagen betrieblicher Simulation 2. Einführung in die Software Arena 3. Aufbau einfacher Simulationsmodelle 4. Simulationslauf und Interpretation der Ergebnisse
Literatur	Altiock Tayfur et al.: Simulation Modeling and Analysis with Arena, Academic Press, 2007 Banks, Jerry et al.: Discrete Event Simulation, 5th edition, Pearson, 2010 Kelton, W. David: Simulation with Arena (5th Edition), Mc Graw Hill Robinson, Stewart: Simulation, 2nd edition, Palgrave Macmillian, 2014

Electronic Commerce

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Electronic Commerce
Modulverantwortliche	Patricia Brockmann
Dozent(en)	Patricia Brockmann
Vorkenntnisse	Inhalte der (Teil-)Module Grundlagen der Betriebswirtschaftslehre
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenz 85 Stunden Vor- und Nachbereiten des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminar
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Englisch
Beitrag zu den Zielen des Studiums	Integration von Konzepten des Geschäftsprozessmanagements mit technischen Fähigkeiten, um IT-unterstützt betriebliche Informationssysteme zu analysieren und entwickeln.
Lernziel	Die Studierenden erhalten in dem Kurs einen vertiefenden Einblick und untersuchen sinnvoller Anwendungen des Electronic Commerce in Unternehmen
Schlüsselqualifikation	Analytische Fähigkeiten im Bereich Anforderungsspezifikationen Entwicklung von technischen Konzepten um Problemlösungen zu generieren Präsentationsfähigkeiten Praktische Anwendung englischer Sprache in einer technischen Umgebung Teamfähigkeiten
Lehrinhalte	Electronic Commerce Applications <ul style="list-style-type: none">• Eye-Tracking in E-Commerce• M-Commerce: mobile und ortsbezogene Anwendungen• Internationale Aspekte des E-Commerce• Virales Marketing, Affiliate Marketing• Soziale Netzwerke in E-Marketing• Künstliche Intelligenz in E-Commerce: Intelligente Agenten, Mustererkennung• E-Payment Systems• Gamification in E-Commerce Weitere Themen können von Studierenden vorgeschlagen werden.
Literatur	"E-Commerce: Business, Technology, Society", Laudon, K., Traver, G. Prentice Hall 2016.
Leistungsnachweis	Seminarleistung: Erstellung, Präsentation und schriftliche Dokumentation eines

Konzepts für eine E-Commerce-Anwendung, auf Englisch oder auf Deutsch (10 Seiten) mit Referat (45 Min). Gewichtung 50% Hausarbeit 50% Referat.

Embedded Systems

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Embedded Systems
Modulverantwortliche	Friedhelm Stappert
Dozent(en)	Friedhelm Stappert
Vorkenntnisse	Grundlagen Betriebssysteme, Grundlagen Rechnerarchitektur, Programmierkenntnisse
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden zur Vor- und Nachbereitung des Lehrstoffs und Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit Praktikum
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Vermittlung von Kenntnissen in einem wesentlichen Bereich der angewandten Informatik
Lernziel	<ul style="list-style-type: none">• Kenntniss der speziellen Eigenschaften und Anforderungen von eingebetteten Systemen• Hardware-Architekturen für eingebettete Systeme verstehen und bewerten• Einfache, hardwarenahe Programme für Mikrocontroller erstellen können• Software-Architekturen für eingebettete Systeme kennen und beurteilen• Modellierungsparadigmen für eingebettete Systeme kennen und anwenden• Fähigkeit zum Entwurf eingebetteter Systeme
Schlüsselqualifikation	Fähigkeit zu komplexem, analytischem Denken; Fähigkeit zu Analyse und Verständnis komplexer Systeme; Ausdauer bei Problemlösungen
Lehrinhalte	<p>Ein Embedded System ist eine HW/SW-Einheit, die über Sensoren und Aktoren in ein Gesamtsystem "eingebettet" ist und spezielle Überwachungs-, und Regelungsaufgaben übernimmt.</p> <p>Embedded Systems finden sich in vielen Bereichen des täglichen Lebens, von der Waschmaschinensteuerung bis zum Airbag-Steuergerät im Automobil. Täglich kommen wir mit Dutzenden solcher Systeme -- mehr oder weniger unbewusst -- in Berührung.</p> <p>Dieser Kurs vermittelt grundlegende Kenntnisse zum Entwurf von eingebetteten Systemen:</p> <ul style="list-style-type: none">• Besondere Eigenschaften, Anwendungsbereiche und Anforderungen von eingebetteten Systemen• Hardware-Architekturen, Microcontroller• Software für eingebettete Systeme• Design / Modellierung eingebetteter Systeme• Echtzeitsysteme

Literatur

Peter Marwedel. *Eingebettete Systeme*. Springer 2007.
Berns, Schürmann, Trapp. *Eingebettete Systeme; Systemgrundlagen und Entwicklung eingebetteter Software*. Vieweg + Teubner 2010.
Jean J. Labrosse. *Embedded Software*. Newnes 2008.
Peter Scholz. *Softwareentwicklung eingebetteter Systeme*. Springer 2005.

Leistungsnachweis

Schriftliche Prüfung (90 Minuten). Aktive Teilnahme am Praktikum ist Voraussetzung für die Klausur

Finanzmathematik

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Finanzmathematik
Modulverantwortliche	Yvonne Stry
Dozent(en)	Yvonne Stry
Vorkenntnisse	Mathematik I und II
Arbeitsaufwand	75 Stunden, davon: 30 Präsenz 45 Stunden Vor- und Nachbereitung
Leistungspunkte	2.5
Semesterwochenstunden	2
Veranstaltungstyp	Seminaristischer Unterricht mit zahlreichen Beispielen und Übungen
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Finanzmathematik als Anwendung der Mathematik in der Wirtschaft
Lernziel	Einführung in die Methoden der Finanzmathematik
Schlüsselqualifikation	Anwendung einfacher mathematischer Formeln bei Geldgeschäften
Lehrinhalte	<ul style="list-style-type: none">• Arithmetische und geometrische Folgen und Reihen• Zinsrechnung (einfache Verzinsung, Zinseszins, gemischte Zinsrechnung, unterjährliche Verzinsung, Effektivzins)• Rentenrechnung (vorschüssige und nachschüssige Rente, Rentenbarwerte, Rentenendwerte)• Tilgungsrechnung (Ratentilgung, Annuitätentilgung)• Ausblick
Literatur	Schwenkert/Stry, Finanzmathematik kompakt, Springer, 2. Auflage 2016
Leistungsnachweis	Klausur 90 min

Integrierte Warenwirtschaft mit einem ERP-System

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Integrierte Warenwirtschaft mit einem ERP-System
Modulverantwortliche	Joachim Scheja
Vorkenntnisse	Inhalte der (Teil-)Module Grundlagen der Betriebswirtschaftslehre, ERP-Praktikum
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 60 Stunden für die Vor- und Nachbereitung des Lehrstoffes und der Übungsaufgaben, 25 Stunden für die Prüfungsvorbereitung
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung mit integrierten Übungen
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die erworbenen Kenntnisse und Methoden bilden eine Grundlage zur Übernahme von Aufgaben in der Geschäftsprozess-Organisation und der Anwendungsentwicklung.
Lernziel	<ul style="list-style-type: none">• Funktionen und Prozesse integrierter Warenwirtschaftssysteme des Handels identifizieren und verstehen• Formen, Technologie, Architektur und Programmierung von ERP-Systemen kennen und analysieren• ERP-Systeme anwenden, vergleichen, hinsichtlich Stärken und Schwächen beurteilen, auswählen und weiterentwickeln• Algorithmen der Bestandssteuerung und der manuellen Kommissionierung analysieren, verbessern und programmieren
Schlüsselqualifikation	Fähigkeit zum Verständnis komplexer Unternehmenssoftware, Lernen am Beispiel, Transferfähigkeit, Problemlösen, algorithmisches Denken
Lehrinhalte	Anhand des ERP-Systems v.soft der Firma Vepos werden u.a. folgende Themengebiete behandelt: <ul style="list-style-type: none">• Aufbau und betriebswirtschaftliche Funktionen von Warenwirtschaftssystemen• Abbildung von Geschäftsprozessen in ERP-Systemen• Usability (Schwachstellen identifizieren, systematisieren und verbessern)• Funktionsweise von E-Commerce: ERP-Anbindung, Multi-Channel-Management• Algorithmen der Bestandssteuerung und (manueller) Kommissionierung
Literatur	<ul style="list-style-type: none">• Becker, J., Uhr, W., Vering, O., Integrierte Informationssysteme in Handelsunternehmen auf der Basis von SAP-Systemen, Heidelberg 2000• Hertel, J., Zentes, J., Schramm-Klein, H.: Supply Chain Management und Warenwirtschaftssysteme im Handel, 2. Aufl., Heidelberg 2011• Schütte, R., Vering, O., Wiese, J., Erfolgreiche Geschäftsprozesse durch standardisierte Warenwirtschaftssysteme – Marktanalyse, Produktübersicht, Auswahlprozess, 3. Aufl., Berlin et al. 2011

- Weber, R., Technologie von Unternehmenssoftware, Heidelberg 2012

Leistungsnachweis

Mündliche Prüfung (20 Min.)

Praxis des Software-Tests eingebetteter Systeme

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Praxis des Software-Tests eingebetteter Systeme
Modulverantwortliche	Kristian Trenkel
Dozent(en)	Kristian Trenkel
Vorkenntnisse	Basiswissen zu Eingebetteten Systemen
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Blockveranstaltung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Die fortschreitende Vernetzung und die Integration von Systemen, vor allem von eingebetteten Systemen, stellt nicht nur Anforderungen an die Entwicklung dieser Systeme. Der Test dieser Systeme ist ein immer größerer Bestandteil der Arbeit von Ingenieuren. Das in dieser Veranstaltung vermittelte Wissen zum Software-Tests eingebetteter Systeme erhöht die Problemlösungskompetenz der Studierenden und verbesserte Ihre Möglichkeiten zur interdisziplinären Zusammenarbeit.
Lernziel	Die Studierenden können den Software-Test für eingebettete Systeme praktisch anwenden. Dabei liegt der Fokus auf den verstehen Funktionaler Tests und dem anwenden des Hardware In The Loop (HIL)-Testverfahrens. Dabei sollen die Studierenden die Abläufe von Testaktivitäten und deren Aufwände innerhalb des Entwicklungsprozesses eingebetteter Systeme erkennen und bewerten können. Den Studierenden sollen die Möglichkeiten und Vorteile des HIL-Testverfahrens bewerten können. Des Weiteren sollen sie auch die Grenzen des Testverfahrens beurteilen und bewerten. Anhand von Beispielen wird den Studierenden das Anwender der Testfahren und Testsysteme praktisch verdeutlicht. Es werden die Begrifflichkeiten entsprechend des ISTQB Glossars wie in der Lehrveranstaltung „Grundlagen des Softwaretests“ verwendet.
Lehrinhalte	<ul style="list-style-type: none">• Motivation für den Software-Test mit Fokus auf Eingebettete Systeme• Einordnung des Tests in den Entwicklungsprozess• Wiederholung fundamentaler Testprozesse• Wiederholung Teststufen und Testarten• Einführung des HIL-Testverfahrens (Technische Entwicklung, Einsatzbereiche, Vorgehen, ...)• Einordnung des HIL-Testverfahrens in die Teststufen• Erläuterung des Aufbaues von typischem HIL-Testsystemen (inklusive der Echtzeitsysteme verschiedener Hersteller)• Erläuterung des Aufgabenverteilung innerhalb des Testsystems (z.B. Echtzeit-Modell, Ablaufsteuerung des Tests)• Darstellung der Tool-Umgebungen für HIL-Testsysteme• Erläuterungen der Testautomatisierung• Darstellung verschiedener Testfälle und ihrer Umsetzung am Beispiel eines

Automobil-Steuergerätes

- Umsetzung eines beispielhaften Testfalles von den Anforderungen über die Testspezifikation und die Testimplementierung bis hin zu den Testergebnissen
- Erläuterung der Schnittstellen zu anderen Testverfahren
- Erläuterung der Grenzen des HIL-Testverfahrens
- Durchführung praktischer Beispiele

Literatur

Grünfelder, Stephan.: "Software-Test für Embedded Systems - Ein Praxishandbuch für Entwickler, Tester und technische Projektleiter", dpunkt.verlag, 2013

Leistungsnachweis

Schriftliche Prüfung - Dauer: 90 min

Programmieren von Mikrocontrollern

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Programmieren von Mikrocontrollern
Modulverantwortliche	Thomas Fischer
Dozent(en)	Thomas Fischer
Vorkenntnisse	Grundlagen der C-Programmierung Rechnerarchitektur Grundlagen
Arbeitsaufwand	150 Stunden, davon: 64 Stunden Präsenz, 46 Stunden zum Lösen der Übungsaufgaben 20 Stunden zum Lesen von Artikeln und Fachbüchern, 20 Stunden zur Nachbereitung und Prüfungsvorbereitung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Seminaristischer Unterricht mit Übungen
Semesterturnus	Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Vertiefung von Kenntnissen und praktischen Fertigkeiten im Bereich Funktionsweise von Computer-Systemen speziell für eingebettete Systeme (Stichwort IoT)
Lernziel	<ul style="list-style-type: none">• Die Studierenden können einfache kombinatorische und sequentielle Systeme entwerfen und anwenden. Sie sind imstande einfache Probleme durch Auswahl eines Mikrocontrollers und Erstellung des Programms in C zu lösen. Sie können im Entwurfsprozess verschiedene Lösungswege vergleichen und die geeigneten Peripheriekomponenten auswählen.• Für den Entwurf von Anwenderprogrammen für eingebettete Systeme (Internet of Things) sind die Studierenden in der Lage die Möglichkeiten der verwendeten Hardware zu analysieren und die dafür notwendigen Konfigurationsparameter abzuleiten.
Schlüsselqualifikation	Fähigkeit zu selbständigem Arbeiten, Fähigkeit zum Verständnis komplexer Systeme Analyse und Klassifikation von Problemen und Identifikation von Lösungen, Teamfähigkeit, Interdisziplinarität Fremdsprachenkenntnisse, insbesondere in Englisch -- die LV kann auf Wunsch/ bei Bedarf jederzeit in englischer Sprache angeboten werden! Unterlagen sind fast ausschließlich in englischer Sprache.
Lehrinhalte	Grundlagen des Aufbaus der Hardware und der internen Baugruppen eines Datenverarbeitungssystems (MC8). Interne Abläufe bei der Ausführung von Befehlen eines Programmes. Unterschiede Mikroprozessor - Mikrocontroller - System on Chip. Grundlagen der Schnittstellen und Peripheriekomponenten (interne und externe Kommunikation). Timer, Interrupts, USART, PS/2 Tastatur, LCD Display. Lösen einer Problemstellung (STM32 - ARM Cortex M3). Erstellen von C-Programmen zur Lösung verschiedener Aufgabenstellungen

angepasst an die verwendete Hardware.
Überleitung von der Interrupt gesteuerten Programmierung eines
Vorder-/Hintergrundsystems zu einer, durch ein Betriebssystem unterstützten
modularen Programmierung.
Zum Einsatz kommen Entwicklungskits mit dem STM32 Mikrocontroller.

Literatur

Herstellerdokumentation von ARM und ST (www.arm.com ;www.st.com)

Leistungsnachweis

Schriftliche Prüfung (60 min, 50%) und Projektaufgabe (20 Std., 50%)

Spezielle Rechnerarchitekturen - vom Embedded System zum Supercomputer

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Spezielle Rechnerarchitekturen - vom Embedded System zum Supercomputer
Modulverantwortliche	Axel Hein
Dozent(en)	Axel Hein
Vorkenntnisse	Rechnersysteme, Betriebssysteme, Programmieren.
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Anwesenheit, 85 Stunden Bearbeitung der Vortrags- und Projektthemen.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung und Seminar.
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Kennenlernen, Verstehen und Bewertung spezieller exemplarischer Rechnersysteme mit Anwendung der konzeptionellen Grundlagen aus den Lehrveranstaltungen Rechnersysteme und Betriebssysteme.
Lernziel	Bei Abschluss des Lernprozesses wird der erfolgreiche Studierende in der Lage sein, Spezielle Computersysteme zu verstehen, zu analysieren, zu bewerten und zu entwerfen.
Schlüsselqualifikation	Fähigkeit zu komplexem Denken, Ausdauer bei Problemlösungen.
Lehrinhalte	Vorstellung und Diskussion konkreter Rechnersysteme und ihrer Komponenten mit Anwendung der konzeptionellen Grundlagen aus den Lehrveranstaltungen Rechnersysteme und Betriebssysteme (Beispiele: Top500 Supercomputers [http://www.top500.org], Prozessoren [Intel Core, Atom, ..., IBM PowerPC, Cell Processor, ..., ARM Processors for mobile devices, ...], DSP Prozessoren zur Digitalen Signalverarbeitung, Graphikprozessoren, Speichersysteme, Bussysteme, etc.)
Literatur	<ul style="list-style-type: none">• Andrew Tanenbaum: Modern Operating Systems, 2014, 4th Global Edition, Pearson (Prentice Hall)• Andrew S. Tanenbaum, Todd Austin: Structured Computer Organization, 2012, 6th International Edition, Pearson (Prentice Hall)• John L. Hennessy, David A. Patterson: Computer Organization and Design – The Hardware/Software Interface, 2013, Morgan Kaufmann Publishers Inc., 5th edition• Aktuelle Veröffentlichungen in Fachpublikationen und Internet
Leistungsnachweis	Anwesenheitspflicht Seminar (Vortrag 45 Minuten, Ausarbeitung 25 Seiten)

Spieltheorie

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Fachwissenschaftliche Wahlpflichtfächer wählbar außerhalb Schwerpunkt
Modul	Spieltheorie
Modulverantwortliche	Ralf-Ulrich Kern
Dozent(en)	Ralf-Ulrich Kern
Vorkenntnisse	Mathematik-Vorlesungen; Kenntnisse in OR vorteilhaft.
Arbeitsaufwand	150 Stunden, davon: 65 Stunden Präsenzzeit, 85 Stunden für Vor- und Nachbereitung des Lehrstoffs, Vorbereitung auf die Prüfung.
Leistungspunkte	5
Semesterwochenstunden	4
Veranstaltungstyp	Vorlesung
Semesterturnus	Wintersemester
Unterrichtssprache	Kurs nur in Deutsch
Beitrag zu den Zielen des Studiums	Mathematische Grundlagen für weiterführende Veranstaltungen über Methoden zur Entscheidungsunterstützung.
Lernziel	Kenntnis wichtiger Begriffe der mathematischen Spieltheorie: Einführung in strategisches Denken; Einführung in die Theorie der Verhandlungslösungen. Fähigkeit, Verhandlungspositionen einzuschätzen.
Schlüsselqualifikation	Grundlagen des strategischen Denkens und von Verhandlungsstrategien.
Lehrinhalte	<p>Die mathematische Spieltheorie befasst sich mit optimalen Entscheidungen bei Vorhandensein mehrerer Entscheidungsträger, anders ausgedrückt, mit optimalem strategischem Handeln. Die Bezeichnung "Spieltheorie" kommt daher, dass die ersten Studien anhand von Gesellschaftsspielen betrieben wurden; inzwischen findet die Spieltheorie Anwendung in Ökonomie, Politik, Bio- und Sozialwissenschaften.</p> <p>In Teil I der Vorlesung (kompetitive Spieltheorie) soll es um Verhalten bei Konkurrenz gehen, wenn also die Interessen der Entscheidungsträger (Spieler) entgegengesetzt gerichtet sind. Man versucht, Handlungsalternativen (Strategien) zu bestimmen, die unter Berücksichtigung des Verhaltens der Mitspieler den eigenen Gewinn maximieren.</p> <p>Stichwörter zum Inhalt: Spiele in extensiver Form, Spiele in Normalform, Matrixspiele mit (im)perfekter Information, mit/ohne Zufallseinfluss, Nullsummenspiele. Gleichgewichtspunkte, gemischte Strategien. Satz von Nash, Lösung von Matrixspielen.</p> <p>In Teil II, der kooperativen Spieltheorie, werden Gewinne betrachtet, die im Gegensatz zur Konkurrenztheorie durch Verfolgung gleichgerichteter Interessen erzielt werden. Das Augenmerk liegt daher nicht auf der Wahl der richtigen Strategie, um einen</p>

bestimmten Gewinn zu realisieren, sondern auf der Aufteilung des gemeinsam erzielten Gewinns unter den (typischerweise mehr als zwei) kooperierenden Spielern. Dabei spielen Gesichtspunkte wie Fairness, Gerechtigkeit, Durchsetzbarkeit, Stabilität eine Rolle.

Man betrachtet Spiele nicht mehr in der Normalform, sondern in der "charakteristischen Form" und versucht, daraus Verhandlungslösungen, also Vorschläge zur Gewinnaufteilung, herzuleiten, entweder in Form von eindeutigen Funktionen oder von mengenwertigen Abbildungen.

Stichwörter zum Inhalt:

Verhandlungslösungen (z. B. Core, Shapley-Wert, Nash-Wert).
Individuelle Rationalität, Verhandlungsstärke, Koalitionsbildung.

Dazu zahlreiche (auch unterhaltsame) Beispiele.

Literatur

- Güth, Werner: Spieltheorie und ökonomische (Bei)Spiele. Springer-Verlag Berlin usw. (1999). ISBN 3-540-54921-8.
- Holler, Manfred J., Illing, Gerhard: Einführung in die Spieltheorie. Springer-Verlag Berlin usw. (7. Aufl. 2009). ISBN 978-3-540-69372-7.

Leistungsnachweis

Klausur (90 min), leider kein Spielturnier :-)

Bachelorarbeit

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Abschlussarbeit
Modul	Bachelorarbeit
Vorkenntnisse	Alle Fähigkeiten, die das Studium vermittelt
Arbeitsaufwand	2 Monate
Leistungspunkte	12
Semesterwochenstunden	0
Veranstaltungstyp	Selbständiges Arbeiten
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Die weitgehend selbständige Arbeit auf wissenschaftlicher Grundlage ist eines der wesentlichen Ziele eines jeden Bachelor-Studiums.
Lernziel	Die Fähigkeit, ein praxisbezogenes Problem aus den Gebieten der Pflicht- und fachwissenschaftlichen Wahlpflichtmodule, auch fachübergreifend, selbständig und auf wissenschaftlichen Grundlagen methodisch zu bearbeiten.
Schlüsselqualifikation	Fähigkeit zu selbständigem Arbeiten auf wissenschaftlicher Grundlage, technisches Schreiben, Analyse und Klassifikation von Problemen, kreatives Problemlösen
Lehrinhalte	Anleitung zu selbständiger Arbeit auf wissenschaftlichen Grundlagen
Literatur	Themenbezogen
Leistungsnachweis	Zur Bachelorarbeit kann sich anmelden, wer den praktischen Teil des praktischen Studiensemesters mit Erfolg abgelegt und insgesamt 160 Leistungspunkte erreicht hat. Die Bachelorarbeit kann mit Genehmigung der Prüfungskommission auch in Englisch oder in einer anderen Fremdsprache verfasst werden. Leistungsnachweis ist die termingerecht abgegebene Arbeit.

Seminar zur Bachelorarbeit

Studiengang	Bachelor Medieninformatik 2. Studienabschnitt Abschlussarbeit
Modul	Seminar zur Bachelorarbeit
Vorkenntnisse	Praktisches Studiensemester
Arbeitsaufwand	60 Stunden, davon: 30 Stunden Präsenzzeit
Leistungspunkte	2
Semesterwochenstunden	2
Veranstaltungstyp	2 SWS Seminar
Semesterturnus	Winter- bzw. Sommersemester
Unterrichtssprache	Kurs in Deutsch und/oder Englisch (bitte nachfragen)
Beitrag zu den Zielen des Studiums	Die Lehrveranstaltung fördert die Kompetenz, die Ergebnisse der eigenen Arbeit zu präsentieren und in Diskussionen zu erläutern. Zuhörer erhalten einen umfassenden Überblick über Arbeitsgebiete an der Hochschule und berufliche Perspektiven in Unternehmen.
Lernziel	Fähigkeit, eine eigene fachlich-wissenschaftliche Arbeit zu präsentieren und zu verteidigen. Fähigkeit, spezielle fachliche Inhalte zu verstehen und zu hinterfragen.
Schlüsselqualifikation	Fähigkeit zu selbständigem Arbeiten, Präsentationskompetenz, Zeitmanagement, praktische Erfahrung
Lehrinhalte	Präsentation, Diskussion und Disputation. Fachliche Inhalte entsprechend den vorgestellten Themen der Bachelor-Arbeiten.
Leistungsnachweis	Jeder Bachelor-Kandidat hält im Seminar einen Vortrag von ca. 45 min Dauer (einschließlich Diskussion) zum Thema seiner Bachelorarbeit. Er muss vor Anmeldung seiner Bachelorarbeit an zwei je 90-minütigen Kursen zu den Themen Wissenschaftliches Schreiben und Literaturrecherche teilnehmen und bis zum Abschluss seiner Bachelorarbeit mindestens 8 Vorträge im Seminar besuchen. Der Studierende führt den Nachweis hierzu selbst und muss diesen vor der Anmeldung bzw. vor der Bewertung seiner Bachelorarbeit dem Erstprüfer seiner Bachelorarbeit vorlegen. Einzelheiten zum organisatorischen Ablauf werden auf www.in.th-nuernberg.de unter -> Studierende -> Infos zum Studium -> Abschlussarbeiten bekanntgegeben. Das Seminar zur Bachelorarbeit wird benotet, die Note wird im Zeugnis ausgewiesen und geht in die Gesamtnote ein.

Hilfsmittel

- 1) keine Hilfsmittel
- 2) unbeschränkt
- 3) Vorlesungsmitschrift
- 4) Vorlesungsumdrucke
- 5) selbstgefertigte Arbeitsunterlagen (DinA4 Seitenzahl)
- 6) mathematische Formelsammlung
- 7) Taschenrechner, nicht programmierbar (Taschenrechner mit vollständiger alphanumerischer Tastatur und/oder Graphikdisplay sind nicht erlaubt)
- 8) Gesetzestexte, z.B. BGB, UrhG, PatG, UWG, Betr.VG, BDSG, StGB, Stopp, TKG
- 9) Lehrbuch