

Staatlich anerkannte Fachhochschule
PTL Wedel, Prof. Dr. D. Harms, Prof. Dr. H. Harms
Gemeinnützige Schulgesellschaft mbH

MODULHANDBUCH
Bachelor-Studiengang
Informatik

B_Inf14.0

Wedel, den 30. Juni 2016

Inhaltsverzeichnis

Modulverzeichnis nach Modulkürzel	1
Modulverzeichnis nach Modulbezeichnung	2
1 Erläuterungen zu den Modulbeschreibungen	5
2 Erläuterung des Dualen Studienmodells	9
3 Studienplan	11
4 Modulbeschreibungen	13
4.1 Programmstrukturen 1	13
4.1.1 Programmstrukturen 1	15
4.1.2 Übg. Programmstrukturen 1	16
4.2 Grundlagen der Funktionalen Programmierung	18
4.2.1 Grundlagen der Funktionalen Programmierung	19
4.2.2 Übg. Grundlagen der Funktionalen Programmierung	19
4.3 Informationstechnik	21
4.3.1 Informationstechnik	22
4.4 Einführung in Digitaltechnik	23
4.4.1 Digitaltechnik 1	24
4.4.2 Prakt. Digitaltechnik	25
4.5 Grundlagen der Mathematik 1	26
4.5.1 Analysis	28
4.5.2 Übg. Analysis	29
4.6 Mathematische Konzepte und Diskrete Mathematik	31
4.6.1 Diskrete Mathematik	33
4.7 Programmstrukturen 2	35
4.7.1 Programmstrukturen 2	37
4.7.2 Übg. Programmstrukturen 2	38
4.8 Programmierpraktikum	40
4.8.1 Programmierpraktikum	42
4.9 Grundlagen der Theoretischen Informatik	44
4.9.1 Grundlagen der Theoretischen Informatik	45
4.10 Rechnerstrukturen und Digitaltechnik	48
4.10.1 Digitaltechnik 2	50
4.10.2 Rechnerstrukturen	51
4.11 Grundlagen der Mathematik 2	53
4.11.1 Grundlagen der Linearen Algebra	55
4.11.2 Grundlagen der Statistik	56
4.12 Datenschutz und Wirtschaftsprivatrecht	58
4.12.1 Datenschutz	59
4.12.2 Wirtschaftsprivatrecht	59
4.13 Algorithmen und Datenstrukturen	61
4.13.1 Algorithmen und Datenstrukturen	62
4.13.2 Übg. Algorithmen & Datenstrukturen	62
4.14 Systemnahe Programmierung	64
4.14.1 Systemnahe Programmierung	66
4.14.2 Übg. Systemnahe Programmierung	66

4.15	Datenbanken 1	68
4.15.1	Einführung in Datenbanken	69
4.15.2	Übg. Einführung in Datenbanken	70
4.16	Rechnernetze	71
4.16.1	Rechnernetze	73
4.16.2	Prakt. Rechnernetze	75
4.17	Lineare Algebra	79
4.17.1	Lineare Algebra	80
4.18	Einführung in die Betriebswirtschaft	83
4.18.1	Einführung in die Betriebswirtschaft	84
4.19	Fortgeschrittene Objektorientierte Programmierung	86
4.19.1	Fortgeschrittene Objektorientierte Programmierung	87
4.19.2	Übg. Fortgeschrittene Objektorientierte Programmierung	88
4.20	Datenbanken 2	89
4.20.1	Datenbanktheorie und -implementierung	90
4.21	Software-Design	92
4.21.1	Software-Design	93
4.22	Web-Anwendungen	95
4.22.1	Web-Anwendungen	97
4.22.2	Übg. Web-Anwendungen	98
4.23	UNIX & Shell-Programmierung	100
4.23.1	UNIX & Shell Programmierung	101
4.23.2	Übg. UNIX & Shell-Programmierung	102
4.24	Implementierung von Geschäftsprozessen in ERP-Systemen	103
4.24.1	Implementierung von Geschäftsprozessen in ERP-Systemen	104
4.24.2	Übg. Implementierung von Geschäftsprozessen in ERP-Systemen	104
4.25	Systemsoftware	106
4.25.1	Konzepte der Betriebssysteme	108
4.25.2	Compilerbau	108
4.26	Anwendungen der Künstlichen Intelligenz	111
4.26.1	Anwendungen der Künstlichen Intelligenz	112
4.27	Software-Qualität	114
4.27.1	Softwarequalität	116
4.28	Systemmodellierung	118
4.28.1	Systemanalyse	120
4.28.2	Prozessmodellierung	121
4.29	Seminar Informatik	124
4.29.1	Seminar Informatik	125
4.30	Anwendungsentwicklung in ERP-Systemen	126
4.30.1	Anwendungsentwicklung in ERP-Systemen	127
4.30.2	Übg. Anwendungsentwicklung in ERP-Systemen	127
4.31	IT-Sicherheit	129
4.31.1	IT-Sicherheit	130
4.32	Soft Skills	132
4.32.1	Assistenz	133
4.32.2	Communication Skills	133
4.33	Software-Projekt	136
4.33.1	Projektmanagement	138
4.33.2	Softwareprojekt	139
4.34	Operations Research	141
4.34.1	Operations Research	142

4.34.2 Übg. Operations Research	143
4.35 Grundlagen der Computergrafik	145
4.35.1 Grundlagen der Computergrafik	146
4.35.2 Prakt. Grundlagen der Computergrafik	146
4.36 Echtzeitsysteme	148
4.36.1 Echtzeitsysteme	150
4.36.2 Interface-Technologie	151
4.36.3 Prakt. Echtzeitsysteme	152
4.37 Prozessmodellimplementation	154
4.37.1 Prozessmodellimplementation	156
4.38 Auslandssemester	158
4.38.1 Auslandssemester	159
4.39 Praxissemester (dual)	160
4.39.1 Praxissemester (dual)	162
4.40 Wissenschaftliche Ausarbeitung (dual)	164
4.40.1 Wissenschaftliche Ausarbeitung (dual)	165
4.41 Betriebspraktikum	166
4.41.1 Betriebspraktikum	167
4.42 Bachelor-Thesis	168
4.42.1 Bachelor-Thesis	169
4.43 Bachelor-Kolloquium	170
4.43.1 Kolloquium	171

Modulverzeichnis nach Modulkürzel

B001 Grundlagen der Mathematik 1	26
B002 Mathematische Konzepte und Diskrete Mathematik	31
B003 Programmstrukturen 1	13
B004 Informationstechnik	21
B006 Einführung in Digitaltechnik	23
B010 Grundlagen der Funktionalen Programmierung	18
B019 Grundlagen der Mathematik 2	53
B020 Programmstrukturen 2	35
B022 Grundlagen der Theoretischen Informatik	44
B023 Rechnerstrukturen und Digitaltechnik	48
B034 Einführung in die Betriebswirtschaft	83
B036 Programmierpraktikum	40
B037 Rechnernetze	71
B040 Algorithmen und Datenstrukturen	61
B042 Datenschutz und Wirtschaftsprivatrecht	58
B043 Systemnahe Programmierung	64
B044 UNIX & Shell-Programmierung	100
B045 Lineare Algebra	79
B052 Datenbanken 1	68
B057 Fortgeschrittene Objektorientierte Programmierung	86
B058 Software-Design	92
B059 Web-Anwendungen	95
B080 Implementierung von Geschäftsprozessen in ERP-Systemen	103
B082 Operations Research	141
B085 Grundlagen der Computergrafik	145
B087 Systemmodellierung	118
B093 Software-Qualität	114
B095 Anwendungen der Künstlichen Intelligenz	111
B096 Systemssoftware	106
B098 Anwendungsentwicklung in ERP-Systemen	126
B099 Auslandssemester	158
B101 Echtzeitsysteme	148
B117 Datenbanken 2	89
B118 Soft Skills	132
B121 Software-Projekt	136
B122 IT-Sicherheit	129
B123 Prozessmodellimplementation	154
B146 Seminar Informatik	124
B150 Bachelor-Thesis	168
B159 Betriebspraktikum	166
B160 Bachelor-Kolloquium	170
B176 Praxissemester (dual)	160
B179 Wissenschaftliche Ausarbeitung (dual)	164

Modulverzeichnis nach Modulbezeichnung

Algorithmen und Datenstrukturen.....	61
Anwendungen der Künstlichen Intelligenz	111
Anwendungsentwicklung in ERP-Systemen	126
Auslandssemester	158
Bachelor-Kolloquium	170
Bachelor-Thesis	168
Betriebspraktikum	166
Datenbanken 1	68
Datenbanken 2	89
Datenschutz und Wirtschaftsprivatrecht	58
Echtzeitsysteme	148
Einführung in die Betriebswirtschaft	83
Einführung in Digitaltechnik	23
Fortgeschrittene Objektorientierte Programmierung	86
Grundlagen der Computergrafik	145
Grundlagen der Funktionalen Programmierung	18
Grundlagen der Mathematik 1	26
Grundlagen der Mathematik 2	53
Grundlagen der Theoretischen Informatik	44
Implementierung von Geschäftsprozessen in ERP-Systemen.....	103
Informationstechnik	21
IT-Sicherheit.....	129
Lineare Algebra	79
Mathematische Konzepte und Diskrete Mathematik	31
Operations Research	141
Praxissemester (dual)	160
Programmierpraktikum	40
Programmstrukturen 1	13
Programmstrukturen 2	35
Prozessmodellimplementation	154
Rechnernetze	71
Rechnerstrukturen und Digitaltechnik	48
Seminar Informatik	124
Soft Skills	132
Software-Design	92

Software-Projekt	136
Software-Qualität	114
Systemmodellierung	118
Systemnahe Programmierung.....	64
Systemsoftware	106
UNIX & Shell-Programmierung	100
Web-Anwendungen.....	95
Wissenschaftliche Ausarbeitung (dual).....	164

1 Erläuterungen zu den Modulbeschreibungen

Im Folgenden wird jedes Modul in tabellarischer Form beschrieben. Die Reihenfolge der Beschreibungen richtet sich nach den Modulkürzeln.

Vor den Modulbeschreibungen sind zwei Verzeichnisse aufgeführt, die den direkten Zugriff auf einzelne Modulbeschreibungen unterstützen sollen. Ein Verzeichnis listet die Modulbeschreibungen nach Kürzel sortiert auf, das zweite Verzeichnis ist nach Modulbezeichnung alphabetisch sortiert. Die folgenden Erläuterungen sollen die Interpretation der Angaben in einzelnen Tabellenfeldern erleichtern, indem sie die Annahmen darstellen, die beim Ausfüllen der Felder zugrunde gelegt wurden.

Angaben zum Modul

Modulkürzel:	FH-internes, bezogen auf den Studiengang eindeutiges Kürzel des Moduls
Modulbezeichnung:	Textuelle Kennzeichnung des Moduls
Lehrveranstaltungen:	Lehrveranstaltungen, die im Modul zusammen gefasst sind, mit dem FH-internen Kürzel der jeweiligen Leistung und ihrer Bezeichnung
Prüfung im Semester:	Auflistung der Semester, in denen nach Studienordnung erstmals Modulleistungen erbracht werden können
Modulverantwortliche(r):	Die strategischen Aufgaben des Modulverantwortlichen umfassen insbesondere: <ul style="list-style-type: none">• Synergetische Verwendung des Moduls auch in weiteren Studiengängen• Entwicklung von Anstößen zur Weiterentwicklung der Moduls und seiner Bestandteile• Qualitätsmanagement im Rahmen des Moduls (z. B. Relevanz, ECTS-Angemessenheit)• Inhaltsübergreifende Prüfungstechnik. Die operativen Aufgaben des Modulverantwortlichen umfassen insbesondere: <ul style="list-style-type: none">• Koordination von Terminen in Vorlesungs- und Klausurplan• Aufbau und Aktualisierung der Modul- und Vorlesungsbeschreibungen• Zusammenführung der Klausurbestandteile, die Abwicklung der Klausur (inkl. Korrekturüberwachung bis hin zum Noteneintrag) in enger Zusammenarbeit mit den Lehrenden der Modulbestandteile• Funktion als Ansprechpartner für Studierende des Moduls bei sämtlichen modulbezogenen Fragestellungen.
Zuordnung zum Curriculum:	Auflistung aller Studiengänge, in denen das Modul auftritt

Querweise:	Angabe, in welchem Zusammenhang das Modul zu anderen Modulen steht
SWS des Moduls:	Summe der SWS, die in allen Lehrveranstaltungen des Moduls anfallen
ECTS des Moduls:	Summe der ECTS-Punkte, die in allen Lehrveranstaltungen des Moduls erzielt werden können
Arbeitsaufwand:	Der Gesamtarbeitsaufwand in Stunden ergibt sich aus den ECTS-Punkten multipliziert mit 30 (Stunden). Der Zeitaufwand für das Eigenstudium ergibt sich, wenn vom Gesamtaufwand die Präsenzzeiten abgezogen werden. Diese ergeben sich wiederum aus den Semesterwochenstunden (SWS), die multipliziert mit 45 (Minuten) geteilt durch 60 die Präsenzzeit ergeben.
Voraussetzungen:	Module und Lehrveranstaltungen, die eine inhaltliche Grundlage für das jeweilige Modul darstellen. Bei Lehrveranstaltungen ist der Hinweis auf das jeweilige Modul enthalten, in dem die Lehrveranstaltung als Bestandteil auftritt.
Dauer:	Anzahl der Semester die benötigt werden, um das Modul abzuschließen
Häufigkeit:	Angabe, wie häufig ein Modul pro Studienjahr angeboten wird (jedes Semester bzw. jährlich)
Studien-/Prüfungsleistungen:	Auflistung aller Formen von Leistungsermittlung, die in den Veranstaltungen des Moduls auftreten
Prozentualer Anteil an der Gesamtnote:	Prozentualer Anteil des Moduls an der Gesamtnote
Sprache:	In der Regel werden die Lehrveranstaltungen aller Module auf Deutsch angeboten. Um Gaststudierenden unserer Partnerhochschulen, die nicht der deutschen Sprache mächtig sind, die Teilnahme an ausgewählten Lehrveranstaltungen zu ermöglichen, ist die Sprache in einigen Modulen als „deutsch/englisch“ deklariert. Dieses wird den Partnerhochschulen mitgeteilt, damit sich die Interessenten für ihr Gastsemester entsprechende Veranstaltungen herausuchen können.
Lernziele des Moduls:	Übergeordnete Zielsetzungen hinsichtlich der durch das Modul zu vermittelnden Kompetenzen und Fähigkeiten aggregierter Form

Angaben zu den Lehrveranstaltungen

Lehrveranstaltung:	Bezeichnung der Lehrveranstaltung, die im Modul enthalten ist
Dozent(en):	Namen der Dozenten, die die Lehrveranstaltung durchführen
Hörtermin:	Angabe des Semesters, in dem die Veranstaltung nach Studienordnung gehört werden sollte
Art der Lehrveranstaltung:	Angabe, ob es sich um eine Pflicht- oder Wahlveranstaltung handelt
Lehrform / SWS:	Die SWS der im Modul zusammen gefassten Lehrveranstaltungen werden nach Lehrform summiert angegeben
ECTS:	Angabe der ECTS-Punkte, die in dieser Lehrveranstaltung des Moduls erzielt werden können
Medienformen:	Auflistung der Medienform(en), die in der Veranstaltung eingesetzt werden
Lernziele/Kompetenzen:	Stichwortartige Nennung die zentralen Lernziele der Lehrveranstaltung
Inhalt:	Gliederungsartige Auflistung der wesentlichen Inhalte der Lehrveranstaltung
Literatur:	Auflistung der wesentlichen Quellen, die den Studierenden zur Vertiefung zu den Veranstaltungsinhalten empfohlen werden. Es wird keine vollständige Auflistung aller Quellen gegeben, die als Grundlage für die Veranstaltung dienen.

2 Erläuterung des Dualen Studienmodells

Ein Praxissemester verändert das Kompetenzprofil der Absolventinnen und Absolventen und bereitet zielgenau auf die spätere Berufstätigkeit vor.

In den Studiengängen mit einem verpflichtenden Auslandssemester (5. Semester bei Betriebswirtschaftslehre und Wirtschaftsingenieurwesen) wird das Auslandssemester durch das Praxissemester ersetzt.

In alle anderen Studiengängen entfallen Lehrveranstaltungen des 6. Semesters (Studienbeginn Wintersemester) im Umfang von 15 - 20 ECTS.

Für Studienanfänger mit Beginn zum Sommersemester entfallen dieselben Veranstaltungen. Im Rahmen wiederholter Studienberatungen erarbeiten die Studierenden zusammen mit den Studiengangsleitern einen individuell angepassten Studienverlaufsplan.

Für einige Kooperationspartner und Studieninteressierte ist das Prädikat "Vollstudium" entscheidungsrelevant. Um diese Klientel zu adressieren, soll das duale Studienmodell auch in einer Vollvariante angeboten werden. Hier ersetzt das Praxissemester kein Hochschulsemester sondern wird zusätzlich erbracht, d.h. das Hochschulsemester kann hinzu gewählt werden. Es handelt sich somit um einen Studiengang mit zwei Regelstudienzeiten. Da dieses Studienmodell noch stärker auf den Bachelorabschluss als höchsten akademischen Abschluss fokussiert, ist ein achtsemestriger Bachelor mit 240 Kreditpunkten gerechtfertigt. In Konsequenz wird kein konsekutiver Masterstudiengang angeboten.

Bei der Bildung der Abschlussnote wird der prozentuale Anteil eines Moduls daran neu ermittelt.

Die folgende Grafik stellt die Studienmodelle exemplarisch für die Informatik-Studiengänge gegenüber.

	Vollzeitstudium	Praxisbegleitendes duales Studium	Praxisbegleitendes Vollstudium
1			
2			
3			
4			
5			
6	Theoriesemester (Mobilitätsfenster)	Praxissemester	Theoriesemester
7	Betriebspraktikum Abschlussarbeit	Betriebspraktikum Abschlussarbeit	Praxissemester
8			Betriebspraktikum Abschlussarbeit
Σ	210 CP	210 CP	240 CP

3 Studienplan

Semester 1	Semester 2	Semester 3	Semester 4	Semester 5	Semester 6	Semester 7
Programmstrukturen 1 C 5 ECTS	Programmstrukturen 2 C 5 ECTS	Algorithmen und Datenstrukturen C 5 ECTS	Fortgeschrittene Objektorientierte Programmierung C 5 ECTS	Systemsoftware C 5 ECTS	IT-Sicherheit C 5 ECTS	Praktikum und Thesis E 30 ECTS
Funktionale Programmierung C 5 ECTS	Programmierpraktikum C 5 ECTS	Systemnahe Programmierung C 5 ECTS	Software-Design C 5 ECTS	Systemmodellierung C 5 ECTS	Soft Skills D 5 ECTS	
Mathematik 1 A 5 ECTS	Theoretische Informatik C 5 ECTS	Rechnernetze C 5 ECTS	UNIX und Shell-Programmierung C 5 ECTS	Software-Qualität C 5 ECTS	Software-Projekt C 10 ECTS	
Diskrete Mathematik A 5 ECTS	Mathematik 2 A 5 ECTS	Datenbanken 1 C 5 ECTS	Datenbanken 2 C 5 ECTS	Künstliche Intelligenz C 5 ECTS	Wahlblock (2 aus 4) F 10 ECTS Prozessimplementierung Echtzeitsysteme Operations Research Computergrafik 1	
Informationstechnik G 5 ECTS	Rechnerstrukturen und Digitaltechnik G 5 ECTS	Lineare Algebra A 5 ECTS	Web-Anwendungen C 5 ECTS	Seminar Informatik C 5 ECTS		
Digitaltechnik G 5 ECTS	Datenschutz und Wirtschaftsprivatrecht B 5 ECTS	Betriebswirtschaftslehre B 5 ECTS	Implementierung von Prozessen in ERP-Systemen B 5 ECTS	Entwicklung in ERP-Systemen B 5 ECTS		

* Im dualen Studienmodell werden die Module durch ein Praxissemester ersetzt.
 Im Vollzeitstudium kann ein freiwilliges Auslandssemester absolviert werden. Es entfallen folgende Module im Umfang von 20 Leistungspunkten:
 IT-Sicherheit, Software-Projekt, Soft Skills

■ **MATHEMATIK**
 ■ **INFORMATIK**
 ■ **KERNFACH**
 ■ **NATURWISSENSCHAFT & TECHNIK**
■ **BWL & RECHT**
 ■ **VERTIEFUNG/ WAHL**

Alle Angaben ohne Gewähr
 Stand 22.06.2016



BSc Informatik

Start zum Sommersemester

Semester 1	Semester 2	Semester 3	Semester 4	Semester 5	Semester 6	Semester 7
Programmstrukturen 1 C 5 ECTS	Programmstrukturen 2 C 5 ECTS	Theoretische Informatik C 5 ECTS	Algorithmen und Datenstrukturen C 5 ECTS	Fortgeschrittene Objektorientierte Programmierung C 5 ECTS	Systemsoftware C 5 ECTS	Praktikum und Thesis E 30 ECTS
Mathematik 1 C 5 ECTS	Programmierpraktikum C 5 ECTS	UNIX und Shell-Programmierung A 5 ECTS	Systemnahe Programmierung C 5 ECTS	Software-Design C 5 ECTS	Seminar Informatik C 5 ECTS	
Mathematik 2 A 5 ECTS	Funktionale Programmierung C 5 ECTS	IT-Sicherheit C 5 ECTS	Datenbanken 1 C 5 ECTS	Datenbanken 2 C 5 ECTS	Software-Qualität C 5 ECTS	
Diskrete Mathematik A 5 ECTS	Rechnernetze C 5 ECTS	Web-Anwendungen C 5 ECTS	Systemmodellierung C 5 ECTS	Wahlblock (2 aus 4) Prozessimplementierung Echtzeitsysteme Operations Research Computergrafik 1 F 10 ECTS	Künstliche Intelligenz C 5 ECTS	
Digitaltechnik G 5 ECTS	Informationstechnik G 5 ECTS	Rechnerstrukturen und Digitaltechnik A 5 ECTS	Lineare Algebra C 5 ECTS	Soft Skills D 5 ECTS	Software-Projekt (Die Lehrveranstaltung „Projektmanagement“ liegt im Sommersemester) C 10 ECTS	
Datenschutz und Wirtschaftsprivatrecht B 5 ECTS	Betriebswirtschaftslehre B 5 ECTS	Implementierung von Prozessen in ERP-Systemen B 5 ECTS	Entwicklung in ERP-Systemen B 5 ECTS			

Bei einem Studienstart im Sommersemester ist für die Nutzung eines Mobilitätsfensters eine Beratung erforderlich.

- **MATHEMATIK**
- **INFORMATIK**
- **BWL & RECHT**
- **KERNFACH**
- **SOFT SKILLS**
- **NATURWISSENSCHAFT & TECHNIK**
- **VERTIEFUNG / WAHL**

Alle Angaben ohne Gewähr
Stand 22.06.2016

4 Modulbeschreibungen

4.1 Programmstrukturen 1

B003 Programmstrukturen 1

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B003
Modulbezeichnung	Programmstrukturen 1
Lehrveranstaltung(en)	B003a Programmstrukturen 1 B003b Übg. Programmstrukturen 1
Modulverantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist ein Einführungsmodul in den Themenbereich Programmierung für Studiengänge mit Informatikbezug. Die erworbenen Kompetenzen sind insbesondere die Grundlage für das Modul „Programmstrukturen 2“, aber auch für die Module „Systemnahe Programmierung“ und „Unix & Shell-Programmierung“.
SWS des Moduls	10
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 92 Stunden Eigenstudium: 58 Stunden
Voraussetzungen	Grundkenntnisse der Schulmathematik, Basisfähigkeit zum abstrakten Denken.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B003a), Abnahme (Teil B003b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls verfügen die Studierenden über erste Kenntnisse hinsichtlich der Entwicklung von Programmen. Studierende mit Vorkenntnissen im Bereich der Programmierung sind in der Lage, diese fachlich fundiert einzuordnen.

Nach Abschluss des Moduls sind die unterschiedlichen Vorkenntnisse angeglichen und es ist eine gemeinsame Basis für die weiteren Veranstaltungen im thematischen Umfeld der Programmierung gelegt.

Die Studierenden beherrschen sowohl die grundlegenden theoretischen Aspekte der Programmierung als auch die Basiskonzepte von imperativen Programmiersprachen: Sie kennen alle wesentlichen Anweisungen zur Umsetzung algorithmischer Strukturen ebenso wie die einfachen und strukturierten Datentypen. Die Studierenden können auf Basis dieser Kenntnis die programmiersprachlichen Mittel adäquat bei der Formulierung von Programmtexten nutzen.

Sie sind in der Lage, vollständige Programme begrenzter Komplexität eigenständig zu entwickeln und dabei die funktionale Korrektheit der Software sicherzustellen.

4.1.1 Programmstrukturen 1

Lehrveranstaltung	Programmstrukturen 1
Dozent(en)	Andreas Häuslein
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration, interaktive Entwicklung und Diskussion von Modellen

Lernziele

Die Studierenden ...

- kennen die grundlegenden Konzepte imperativer Programmiersprachen und ihre Umsetzung in der Programmiersprache Pascal und können diese benennen.
- kennen der Syntax, Semantik und Pragmatik als wesentliche Aspekte von Programmiersprachen und können diese unterscheiden.
- kennen wichtigsten Sprachbestandteile der Programmiersprache Pascal und beschreiben diese.
- setzen die Konzepte und Sprachbestandteile angemessen zur Lösung von Problemstellungen begrenzter Komplexität ein und bauen vollständige Programme für diese Problemstellungen auf.
- kennen die wesentlichen Datenstrukturen imperativer Programmiersprachen und wählen zwischen diesen in Abhängigkeit von der Aufgabenstellung sicher aus.
- kennen wesentliche Qualitätskriterien und können diese bei der Software-Entwicklung berücksichtigen.
- führen eine Fehlersuche und -beseitigung (Debugging) bei ihren Programmtexten durch.

Inhalt

- Grundkonzepte der Datenverarbeitung
- Entwurf und Darstellung von Algorithmen
- Allgemeine Aspekte von Programmiersprachen
- Daten in Programmen
 - Grundlegende Datentypen
 - Variablen, Zuweisungen, Konstanten
- Grundsätzlicher Aufbau von Programmen
- Operatoren und Ausdrücke
- Einfache und strukturierte Anweisungen
- Statische strukturierte Datentypen und ihre Nutzung
 - Strings

- Arrays
- Records
- Sets
- Zeigertypen
 - Besonderheiten und Probleme bei der Nutzung von Zeigertypen
 - Aufbau dynamischer Datenstrukturen mit Hilfe von Zeigertypen
- Strukturierung von Programmen
 - Prozeduren und Funktionen
 - Units

Literatur

- COOPER, Doug; CLANCEY, Michael:
PASCAL, Lehrbuch für das strukturierte Programmieren.
6. Aufl. Wiesbaden: Vieweg+Teubner, 2003.
- OTTMANN, Thomas; WIDMAYER, Peter:
Programmierung mit PASCAL. 8. Aufl., Vieweg+Teubner, 2011.
- HENNING, Peter A.; VOGELSANG, Henning:
Taschenbuch Programmiersprachen.
München: Carl Hanser Fachbuchverlag, 2007.
- GUMM, Heinz-Peter; SOMMER, Manfred: Einführung in die Informatik.
11. Aufl. München: Oldenbourg Wissenschaftsverlag, 2013.
- VAN CANNEYT, Michael:
Free Pascal 2: Handbuch und Referenz.
Böblingen: C& L Computer- und Literaturverlag, 2009.
- FREE PASCAL TEAM:
Free Pascal <http://www.freepascal.org>. Aktualisierungsdatum März 2014

4.1.2 Übg. Programmstrukturen 1

Lehrveranstaltung	Übg. Programmstrukturen 1
Dozent(en)	Lars Neumann
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Software demonstration

Lernziele

Die Studierenden ...

- festigen und vertiefen ihr Wissen zu den in der zugehörigen Vorlesung „Programmstrukturen 1“ vorgestellten Konzepten
- beherrschen die Arbeit mit einer modernen Entwicklungsumgebung (Embarcadero Rad Studio XE2)
- erweitern ihre Teamfähigkeit durch die eigenständige praktische Anwendung des erlernten Wissens in Zweiergruppen

Inhalt

Ausgehend von Struktogrammen, Syntaxdiagrammen und grundlegenden Datentypen werden in der Übung Programmstrukturen 1 in den einzelnen Aufgaben Ein- und Ausgabe, Operatoren, Bedingungen, Schleifen, Strings (sowohl über Stringfunktionen als auch über indizierten Zugriff), Arrays, Records, Mengen, Prozeduren und Funktionen sowie anfänglich Zeiger behandelt.

Die Inhalte höherer Aufgaben schließen dabei ggf. die Inhalte der vorherigen mit ein.

Literatur

Skript:

- Vorlesungsskript unter <https://stud.fh-wedel.de/handout/Haeuslein/Programmstrukturen%201/>
- Weiteres Material unter <http://www.fh-wedel.de/mitarbeiter/ne/programmstrukturen-1/>

4.2 Grundlagen der Funktionalen Programmierung

B010 Grundlagen der Funktionalen Programmierung

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B010
Modulbezeichnung	Grundlagen der Funktionalen Programmierung
Lehrveranstaltung(en)	B010a Grundlagen der Funktionalen Programmierung B010b Übg. Grundlagen der Funktionalen Programmierung
Modulverantwortliche(r)	Prof. Dr. Uwe Schmidt
Zuordnung zum Curriculum	Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist eine sinnvolle Grundlage für alle Module über Programmiersprachen und Softwaretechnik.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Außer elementarer Schulmathematik sind keine Vorkenntnisse für dieses Modul erforderlich.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B010a), Abnahme (Teil B010b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls kennen die Studierenden das Paradigma des funktionalen Programmierens und können damit praktisch arbeiten.

Sie kennen die Einfachheit und Eleganz der Konstruktion von Algorithmen durch das Weglassen von Programmvariablen, Zuweisungen und Schleifen. Sie können diese durch praktische Übungen auch konkret nachvollziehen. Sie beherrschen die Konstruktion rekursiver Algorithmen und kennen die Unterschiede zwischen Endrekursion, linearer Rekursion und allgemeineren Rekursionsformen.

Neben Korrektheitsargumentationen verfügen sie ferner über ein erstes Grundverständnis über das Führen solcher Argumentationen, wie Terminierung und Ableitung von Gesetzen und Programmeigenschaften. Sie kennen die Vorteile beim Entwickeln in der Programmiersprache Haskell bezüglich einer streng getypten und auf mathematischen Konzepten basierenden Programmiersprache.

Sie verfügen über die Fähigkeiten zur Abstraktion und zur Modellbildung. Ferner verfügen sie über eine Basis für ein sorgfältiges, systematisches und methodisches Vorgehen beim Programmieren im Kleinen.

4.2.1 Grundlagen der Funktionalen Programmierung

Lehrveranstaltung	Grundlagen der Funktionalen Programmierung
Dozent(en)	Uwe Schmidt
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration

Lernziele

Die Studierenden ...

- kennen die Programmierparadigmen der funktionalen Programmierung und der Algorithmenkonstruktion ohne Programmvariablen, Zuweisungen und Schleifen.
- besitzen das Verständnis und den Umgang mit Rekursion beim Entwickeln von Algorithmen.
- kennen die Vorteile einer streng getypten Programmiersprache bei der Programmentwicklung und beim Erlernen einer neuen Programmiersprache.
- erwerben das erste Grundverständnis über das Führen von Korrektheitsargumentationen wie Terminierung und Ableitung von Gesetzen und Programmeigenschaften.
- sind geschult bezüglich der Fähigkeiten zur Abstraktion und zur Modellbildung.
- trainieren das sorgfältige, methodische und systematische Programmieren im Kleinen.

Inhalt

- Einleitung
- Erste Schritte
- Typen und Klassen
- Funktionsdefinitionen
- Rekursive Funktionen
- Funktionen höherer Ordnung
- Korrektheitsargumentationen
- Selbstdefinierte Datenstrukturen und Bäume

Literatur

- Uwe Schmidt:
Funktionale Programmierung,
Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~si/vorlesungen/fp/fp.html>
- Graham Hutton:
Programming in Haskell
Cambridge University Press, 2007, ISBN-13: 9780511292187

4.2.2 Übg. Grundlagen der Funktionalen Programmierung

Lehrveranstaltung	Übg. Grundlagen der Funktionalen Programmierung
Dozent(en)	nicht benannt
Hörtermin	1

Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- beherrschen die grundlegenden Sprachelemente der Programmiersprache Haskell und die grundlegenden Konzepte der funktionalen Programmierung.
- erlangen die Fähigkeit zur selbständigen Erstellung kleiner, aber nicht trivialer, Algorithmen einschließlich systematischer Tests dieser Programmteile.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache Haskell behandelt, die nicht Bestandteil der Vorlesung sind wie z.B. Entwicklungsumgebungen für Haskell.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Grundlagen der Funktionalen Programmierung

4.3 Informationstechnik

B004 Informationstechnik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B004
Modulbezeichnung	Informationstechnik
Lehrveranstaltung(en)	B004a Informationstechnik
Modulverantwortliche(r)	PD Dr. Dennis Säring
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul “Informationstechnik” ist ein Einführungsmodul und soll ein breites Grundverständnis für die Funktionsweise von Rechnern vermitteln. Die erworbenen Kompetenzen stellen damit die Grundlagen für zum Beispiel die Module “Rechnerstrukturen und Digitaltechnik”, “Systemsoftware” und “Großintegrierte Systeme” dar.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Keine
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Grundlegendes Verständnis für die Funktionsweise von Rechnern, sowohl aus Sicht der technischen Vorgänge und technischen Funktionselemente als auch aus informationstheoretischer Sicht.

Kenntnisse der rechnerinternen Abläufe auf allen technischen Beschreibungsebenen: vom Transistor, über Logikgatter und Schaltnetzen, hin zu Prozessorstrukturen, der Maschinenbefehlsebene und der Hochsprachenbefehlsebene.

Verständnis des quantitativen Informationsbegriffs und unterschiedlichen Kodierungsmöglichkeiten von Informationen, sowohl verlustfrei als auch verlustbehaftet.

Wissen um alternative Informationsverarbeitende Ansätze, die sich stark von der von-Neumann-Architektur unterscheiden.

4.3.1 Informationstechnik

Lehrveranstaltung	Informationstechnik
Dozent(en)	Dennis Säring
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	5.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, E-Learning

Lernziele

Die Studierenden ...

- besitzen grundlegende Kompetenzen zum Verständnis der Funktionalität von Rechnern in Bezug auf ihre informationstheoretischen Grundlagen und deren praktische Implementierung
- können Vorgänge der Informationsverarbeitung auf der Maschinenebene theoretisch sowie praktisch umsetzen
- sind in der Lage die Umsetzung von Befehlen höherer Sprachebenen in Maschinenbefehle und in deren rechnerinternen Interpretation nachzuvollziehen
- kennen die Ansätze aktueller Rechnerstrukturen und Kommunikationsschnittstellen mit der Peripherie
- sind vertraut mit informationstheoretischen Ansätzen und unterschiedlichen Kodierungsverfahren.

Inhalt

- Grundlagen der Halbleitertechnik
- Logikgatter und Schaltnetze
- Zahlendarstellung und Berechnung
- FlipFlop und weitere Speicherstrukturen
- Moderne Rechnerarchitekturen
- Programmcode zu Assembler
- Computerperipherie
- Informationstheorie und Kodierung

Literatur

- Gumm, Hans-Peter; Sommer, Manfred: Einführung in die Informatik, Oldenbourg, 8. Auflage 2009.
- Müller, Käser, et., al. :Technische Informatik 1, vdf-Hochschulverlag Zürich, 2003
- Schiffmann, Schmitz: Technische Informatik 2, Grundlagen der Computertechnik, Springer-Verlag 1998
- Martin: Einführung in die Rechnerarchitektur, Fachbuchverlag Leipzig, 2003

4.4 Einführung in Digitaltechnik

B006 Einführung in Digitaltechnik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B006
Modulbezeichnung	Einführung in Digitaltechnik
Lehrveranstaltung(en)	B006a Digitaltechnik 1 B006b Prakt. Digitaltechnik
Modulverantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Informatik (Bachelor) IT-Engineering (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Einführung in Digitaltechnik“ ist ein Einführungsmodul. Die erworbenen Kompetenzen stellen Grundlagen für zum Beispiel die Module „Rechnerstrukturen und Digitaltechnik“, „Diskrete Systeme“ und „Systementwurf mit VHDL“ dar. Grundsätzlich kann das Modul sinnvoll mit den Modulen kombiniert werden, die ein Rechnersystem auf höheren Abstraktionsebenen (über dem Gatterniveau) behandeln.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Es werden mathematische Grundlagenkenntnisse entsprechend der Abitur-Stufe (überwiegend diskrete Mathematik) sowie das Verständnis einfacher technisch-naturwissenschaftlicher Zusammenhänge vorausgesetzt.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B006a), Praktikumsbericht / Protokoll (Teil B006b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Das Ziel des Moduls besteht in Vermittlung der allgemeinen Konzepte und Prinzipien des Aufbaus, der Funktionsweise sowie der Analyse und des Entwurfs digitaler Systeme. Angefangen bei Grundbegriffen (analoge und digitale Signale und Systeme, Schaltvariablen, Schaltalgebra usw.) lernen die Studierende Schaltnetze als technische Umsetzung von Schaltfunktionen kennen. Als inhaltliche Voraussetzung für das Modul „Rechnerstrukturen und Digitaltechnik“ werden schließlich die Speicherelemente diskutiert. Somit sind Studierende nach dem Abschluss des Moduls auf die Betrachtung komplexer, zustandsbehafteter Systeme vorbereitet, mit der das Modul „Rechnerstrukturen und Digitaltechnik“ beginnt.

4.4.1 Digitaltechnik 1

Lehrveranstaltung	Digitaltechnik 1
Dozent(en)	Sergei Sawitzki
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout

Lernziele

Die Studierenden ...

- gewinnen eine Einsicht in die mathematischen Grundlagen der digitalen Schaltungstechnik
- erlernen die Methoden zur Analyse, Darstellung und Vereinfachung von Schaltfunktionen
- begreifen ein Schaltnetz als technische Umsetzung einer Schaltfunktion
- lernen die wichtigsten Grundelemente digitaler Systeme kennen
- erwerben die Fähigkeit, einfache digitale Systeme zu begreifen, zu spezifizieren, zu entwerfen und zu optimieren

Inhalt

- Einleitung: Digitale Systeme
- Mathematische Grundlagen
 - Entstehungsgeschichte
 - Aussagenlogik und Boolesche Algebra
 - Schaltalgebra, Schaltfunktionen und Schaltfunktionssysteme
 - Operatorensysteme
 - Normalformen und Dualitätsprinzip
- Schaltnetze
 - Darstellung
 - Vereinfachung (KV-Diagramme, QMCV, BDDs)
 - Analyse (Funktion, Komplexität, Zeitverhalten)
 - Synthese und Realisierung
 - Beispiele
- Speicherelemente

Literatur

- Hoffmann, Dirk: Grundlagen der technischen Informatik, Carl Hanser Verlag 2007
- Schiffmann, Wolram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Beuth, Klaus: Elektronik 4. Digitaltechnik, 13. Auflage Vogel Verlag und Druck 2003

4.4.2 Prakt. Digitaltechnik

Lehrveranstaltung	Prakt. Digitaltechnik
Dozent(en)	Dieter Opitz
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	Handout

Lernziele

Fähigkeit zur Übertragung des theoretischen Wissens in eine erlebbare, reale technische Umgebung.

Inhalt

- Schaltnetzentwurf Schaltnetz wird aus einer gegebenen Spezifikation formal entworfen. Der Entwurf wird auf einem IC-Trainer realisiert. Die Schaltung wird auf Funktion und Einhaltung der Spezifikation überprüft. Die Ergebnisse werden dokumentiert.
- Schaltwerkentwurf Schaltwerk (z. B. ein Zähler) wird aus einer gegebenen Spezifikation formal entworfen. Der Entwurf wird auf einem IC-Trainer realisiert. Die Schaltung wird auf Funktion und Einhaltung der Spezifikation überprüft. Die Ergebnisse werden dokumentiert.

Literatur

Vorlesungsskript

4.5 Grundlagen der Mathematik 1

B001 Grundlagen der Mathematik 1

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B001
Modulbezeichnung	Grundlagen der Mathematik 1
Lehrveranstaltung(en)	B001a Analysis B001b Übg. Analysis
Modulverantwortliche(r)	Prof. Dr. Eike Harms
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll mit anderen Modulen der Mathematik zu kombinieren und zur Bildung mathematischer Grundlagenkompetenzen in allen naturwissenschaftlichen, ingenieurtechnischen und wirtschaftswissenschaftlichen Studiengängen verwendbar.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Voraussetzung zur Teilnahme am Modul sind schulische Grundlagen der Mathematik. Insbesondere gehören hierzu die grundlegenden Begriffe über Mengen, das Rechnen mit reellen Zahlen, Gleichungen mit einer Unbekannten, Basiswissen zur elementaren Geometrie sowie zu Funktionen und Kurven.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B001a), Teilnahme empf. oder Pflicht (Teil B001b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nachdem erfolgreichen Abschluss des Moduls besitzen die Studierenden Rechenfertigkeiten, anschauliche Vorstellungen und theoretisches Verständnis von Funktionen. Sie können dieses

auf Funktionen einer reellen Veränderlichen anwenden, Problemstellungen und Lösungswege klassifizieren und bewerten sowie Problemlösungen prüfen und beurteilen. Sie beherrschen die Grundbegriffe der Analysis einer Veränderlichen, können dieses auf Funktionen mehrere Veränderlicher übertragen und als Fundament für die weiteren fachwissenschaftlichen Studien nutzen. Sie verfügen über formalisierte mathematische Denk- und Arbeitsweisen und sind befähigt mathematische Kausalzusammenhänge aufzustellen und sich in neue formale Systeme einzuarbeiten.

Durch die Übungen erarbeiten sie sich einen sicheren, präzisen und selbständigen Umgang mit den in den Vorlesungen behandelten Begriffen, Aussagen und Methoden. Praxisorientierte Problemstellungen können sie in mathematische Beziehungen und Modelle umsetzen und anhand dieser Modelle bearbeiten. Sie können die Praxisrelevanz der Analysis für verschiedene Fachgebieten bewerten und die Analysis auf Problemstellungen aus Informatik, Technik und Ökonomie anwenden.

4.5.1 Analysis

Lehrveranstaltung	Analysis
Dozent(en)	Eike Harms
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tutorien

Lernziele

Die Studierenden ...

- kennen und verstehen die grundlegenden Begriffe, Aussagen und Methoden der Analysis,
- können mathematische Regeln korrekt anwenden,
- verstehen Beweistechniken,
- erkennen die fundamentale Bedeutung des Grenzwertbegriffes für die Analysis,
- beherrschen die Methoden des Differenzierens und Integrierens,
- können die eindimensionale Differentialrechnung bei praxisorientierten Fragestellungen flexibel einsetzen und dabei beurteilen, welche analytischen Hilfsmittel für welche Problemstellungen zielführend sind,
- können praxisorientierte Problemstellungen in mathematische Beziehungen bzw. Modelle umzusetzen und anhand analytischer Modelle weiter bearbeiten
- können neue, unklare und ungewöhnliche Aufgabenstellungen als solche erkennen und zur Bearbeitung weiterführende Hilfestellung in Anspruch nehmen,
- verfügen über gesteigerte Kompetenzen sich Fähigkeit durch Selbststudium anzueignen und sich in neue formale Systeme einzuarbeiten

Inhalt

- Zahlentypen
- Folgen
 - Bildungsgesetze
 - Grenzwerte
- Funktionen, Relationen
 - Funktionstypen
 - Umkehrfunktion
- Differentialrechnung
 - Differentiationsregeln
 - Anwendungen der Differentialrechnung (Kurvendiskussionen und Extremwerte)
- Integralrechnung

- Integrationsmethoden
- Anwendungen der Integralrechnung (Bestimmte Integrale)
- Funktionen mit zwei Variablen
 - Partielle Differentiation
 - Extremwertaufgaben mit Nebenbedingungen

Literatur

- BÖHME, Gert:
Analysis 1.
6. Aufl. Berlin: Springer-Verlag, 1990
- FETZER, Albert; FRÄNKEL, Heiner:
Mathematik 1.
10. bearbeitete Aufl. Berlin: Springer-Verlag, 2008
- FETZER, Albert; FRÄNKEL, Heiner:
Mathematik 2.
6. korrigierte Aufl.. Berlin: Springer-Verlag, 2009
- HENZE, Norbert; Last, Günter:
Mathematik für Wirtschaftsingenieure 1.
2. Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2005
- KUSCH, Lothar:
Mathematik. Aufgabensammlung mit Lösungen. Bd. 3
9. Aufl. Berlin: Cornelsen Verlag, 1995
- OHSE, Dietrich: Mathematik für Wirtschaftswissenschaftler 1. Analysis.
6. Aufl. München: Verlag Vahlen, 2004
- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler 1: Ein Lehr- und Arbeitsbuch für das Grundstudium.
12. überarbeitete und erweiterte Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2009
- PREUSS, Wolfgang; WENISCH, Günter:
Lehr- und Übungsbuch Mathematik 1: Grundlagen - Funktionen - Trigonometrie.
2. neu bearbeitete Aufl. München: Carl Hanser Verlag, 2003
- PREUSS, Wolfgang; WENISCH, Günter:
Lehr- und Übungsbuch Mathematik 2: Analysis.
3. Aufl. München: Carl Hanser Verlag, 2003

4.5.2 Übg. Analysis

Lehrveranstaltung	Übg. Analysis
Dozent(en)	Fikret Koyuncu
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden können ...

- praktische Problemstellungen mathematisch formulieren

- beurteilen, welche analytischen Hilfsmittel zielführend sind
- neue, unklare und ungewöhnliche Aufgabenstellungen als solche erkennen und mit weiterführender Hilfestellung bearbeiten
- Lösungsansätze präsentieren und begründen

Inhalt

- Bearbeitung von Übungsaufgaben aus dem Themenspektrum der zugehörigen Lehrveranstaltung
- Vorstellung und Diskussion möglicher Lösungswege
- Heranführung an mathematische Softwaretools

Literatur

PAPULA, Lothar: Mathematik für Ingenieure und Naturwissenschaftler: Klausur- und Übungsaufgaben 4. überarbeitete und erweiterte Aufl. Wiesbaden: Vieweg + Teubner Verlag, 2010

4.6 Mathematische Konzepte und Diskrete Mathematik

B002 Mathematische Konzepte und Diskrete Mathematik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B002
Modulbezeichnung	Mathematische Konzepte und Diskrete Mathematik
Lehrveranstaltung(en)	B002a Diskrete Mathematik
Modulverantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist ein Einführungsmodul. Es liefert die Konzepte für ein tieferes Verständnis der anderen Mathematikmodule wie „Grundlagen der Mathematik 1“ und „Lineare Algebra“. Die vermittelten Konzepte und Inhalte werden gebraucht in den Modulen „Informationstechnik“, „Einführung in Digitaltechnik“, „Programmstrukturen 1 und 2“, „Grundlagen der Theoretischen Informatik“, „Algorithmen und Datenstrukturen“, „Datenbanken 1“ und „Anwendungen der Künstlichen Intelligenz“.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Die Studierenden müssen auf dem Kenntnisstand der Schulmathematik der 9. Klasse (Gymnasium) sein. Sie sollten insbesondere mit den Mengen der natürlichen, ganzen, rationalen und reellen Zahlen sowie mit den dafür geltenden Rechengesetzen vertraut sein. Außerdem wird ein gutes logisches Denkvermögen vorausgesetzt.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss de Moduls verstehen und beherrschen die Studierenden allgemeine formalisierte mathematische Denk- und Arbeitsweisen. Sie kennen grundlegende Beweistechniken

und haben Einsicht in die Notwendigkeit mathematischen Beweisens. Ferner verfügen sie über die Fähigkeit, Kausalzusammenhänge nachzuvollziehen und zu erarbeiten. Sie können mathematische Regeln korrekt anwenden. Die Studierenden sind in der Lage, die Verwendbarkeit der vermittelten mathematischen Hilfsmittel auf praktische Problemstellungen kompetent zu beurteilen. Sie können praxisorientierte Problemstellungen in mathematische Beziehungen bzw. Modelle umsetzen und anhand dieser Modelle bearbeiten und lösen. Ferner können sie sich in neue formale Systeme einarbeiten und dessen Regelwerke richtig anwenden. Schließlich besitzen sie die Fähigkeit, neue, unklare und ungewöhnliche Aufgabenstellungen als solche zu erkennen und zu ihrer Bearbeitung weiterführende Hilfestellung in Anspruch zu nehmen. Im Speziellen beherrschen sie die wesentlichen Konzepte der Diskreten Mathematik und können diese auf anwendungsbezogene Problemstellungen in den Gebieten der Informatik, Technik und Wirtschaft anwenden.

4.6.1 Diskrete Mathematik

Lehrveranstaltung	Diskrete Mathematik
Dozent(en)	Sebastian Iwanowski
Hörtermin	1
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	5.0
Lehr- und Medienform(en)	Tutorien

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Beherrschen der grundlegenden mathematischen Begriffe und Konzepte (Definition, Satz, Beweis) und Fähigkeit zur Unterscheidung derselben.
- Beherrschen der Grundlagen und der Formalisierung logischen Denkens.
- Verständnis elementarer Logik und Mengenlehre und des inneren Zusammenhangs dieser Gebiete.
- Darauf aufbauendes Verständnis von Relationen und Funktionen.
- Fähigkeit, elementare Beweisprinzipien wie vollständige Induktion in verschiedenen Kontexten anzuwenden.
- Beherrschen der grundlegenden Sätze der elementaren Zahlentheorie, Gruppen- und Körpertheorie, Kombinatorik und Graphentheorie und selbständige Anwendung an Beispielen.

Inhalt

- Grundlagen der Mathematik
 - Einführung
 - Aussagenlogik
 - Prädikatenlogik
- Mengenlehre
 - Grundlegende Begriffe und Konzepte
 - Relationen
 - Funktionen
 - Boolesche Algebren
- Beweisführung
 - Strukturen der mathematischen Beweisführung
 - Vollständige Induktion
 - Beweisstrategien
- Zahlentheorie
 - Teilbarkeit

- Teilen mit Rest
- Primzahlen
- Modulare Arithmetik
- Algebraische Strukturen
 - Gruppen
 - Körper
- Kombinatorik
 - Zählformeln für Mengen
 - Permutationen
- Graphentheorie
 - Terminologie und Repräsentation
 - Wege in Graphen
 - Bäume
 - Planare Graphen
 - Färbungen

Literatur

- Sebastian Iwanowski / Rainer Lang:
Diskrete Mathematik mit Grundlagen, Springer 2014, ISBN 978-3-658-07130-1 (Print),
978-3-658-07131-8 (Online)
- Albrecht Beutelspacher / Marc-Alexander Zschiegner:
Diskrete Mathematik für Einsteiger.
Vieweg 2004 (2. Auflage), ISBN 3-528-16989-3
- Norman L. Biggs:
Discrete Mathematics.
Oxford University Press 2002, ISBN 0-19-850717-8
- Neville Dean: Diskrete Mathematik.
Pearson Studium, Reihe “im Klartext” 2003, ISBN 3-8273-7069-8
- Christoph Meinel / Martin Mundhenk:
Mathematische Grundlagen der Informatik.
Teubner 2002 (2. Auflage), ISBN 3-519-12949-3

4.7 Programmstrukturen 2

B020 Programmstrukturen 2

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B020
Modulbezeichnung	Programmstrukturen 2
Lehrveranstaltung(en)	B020a Programmstrukturen 2 B020b Übg. Programmstrukturen 2
Modulverantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul basiert auf den im Modul „Programmstrukturen 1“ erworbenen Kompetenzen. Es schafft die Grundlagen für Module der fortgeschrittenen Programmierung in Informatik-Studiengängen, zum Beispiel die Module „Algorithmen und Datenstrukturen“, „Fortgeschrittene Objektorientierte Programmierung“ und „Web-Anwendungen“.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Kenntnis der Grundkonzepte imperativer Programmiersprachen hinsichtlich der wesentlichen statischen Datenstrukturen und Anweisungen zur Umsetzung der algorithmischen Grundstrukturen, Fähigkeit zur Erstellung von vollständigen Programmen begrenzter Komplexität. Die erfolgreiche Teilnahme an der Übung B003b Übg. Programmstrukturen 1 ist Voraussetzung, um an der Übung B020b Übg. Programmstrukturen 2 teilzunehmen.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B020a), Abnahme (Teil B020b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls kennen die Studierenden die Basiskonzepte der Objektorientierten Programmierung und können diese Kenntnisse zur Erstellung von objektorientierter Software

begrenztem Umfang einsetzen. Die Studierenden wissen, wie die Programmiersprache Java grundsätzlich aufgebaut ist, sie kennen die grundlegenden Sprachelemente der Programmiersprache Java und können diese sicher zur Realisierung entsprechender algorithmischer Strukturen nutzen. Die Studierenden können Bezüge zwischen der imperativ prozeduralen Sprache Pascal und der Programmiersprache Java herstellen.

Die Studierenden besitzen die Fähigkeit, eine moderne Entwicklungsumgebung zur Software-Erstellung zu nutzen. Sie können mit den Mitteln der objektorientierten Sprache Java einfache rekursive Datenstrukturen (Listen) aufbauen und kennen grundlegende Algorithmen für diese Datenstrukturen.

Die Studierenden sind in der Lage die Modularisierungskonzepte der Sprache Java, soweit sie Gegenstand der Vorlesung sind, zu einer problemadäquaten Strukturierung eines Programms mittleren Umfangs und begrenzter Funktionalität einzusetzen. Sie kennen bezogen auf die Gestaltung einer grafischen Benutzungsoberfläche die wesentlichen Regeln und Richtlinien und sind in der Lage diese für die Gestaltung konkreter Oberflächen einzusetzen.

4.7.1 Programmstrukturen 2

Lehrveranstaltung	Programmstrukturen 2
Dozent(en)	Andreas Häuslein
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration

Lernziele

Die Studierenden ...

- identifizieren die Basiskonzepte der Objektorientierten Programmierung und stellen diese den Konzepten der prozeduralen Programmierung gegenüber.
- entwickeln Software auf der Grundlage der Objektorientierten Programmierung.
- stellen die grundlegenden Sprachelemente (Datentypen, Anweisungen, Realisierung von objektorientierten Konzepten) von Java zusammen und wählen daraus aus, um Java-Programme mittlerer Komplexität zu entwickeln.
- vergleichen die Programmiersprachen Pascal und Java und stellen ihre Gemeinsamkeiten und Unterschiede heraus.
- setzen eine moderne Entwicklungsumgebung zur Unterstützung der Softwareentwicklung ein und stellen die damit verbundenen Funktionalitäten und Vorgehensweisen dar.
- entwerfen einfache dynamische Datenstrukturen im Kontext einer objektorientierten Programmiersprache.
- erläutern grundlegende Algorithmen, die auf den vermittelten Datenstrukturen arbeiten.
- entwerfen für Programme mittlerer Komplexität durch Einsatz geeigneter Elemente der Programmiersprache Java eine angemessene Modularisierung und legen entsprechende Schnittstellen zwischen den Modulen fest.
- benennen die Grundregeln der benutzungsgerechten Gestaltung von Programmen und nutzen diese, um Benutzungsoberflächen von Programmen begrenzter Funktionalität sowohl strukturell als auch funktional angemessen zu gestalten.

Inhalt

- Grundkonzept der Programmiersprache Java
 - Grundlegende Eigenschaften der Sprache
 - Grundlegender Aufbau von Java-Programmen
 - Ausführung von Java-Programmen
- Vorstellung der eingesetzten Entwicklungsumgebung (Eclipse)
- Grundlegende Programmelemente
 - Primitive Datentypen in Java
 - Variablen, Zuweisung, Gültigkeitsbereiche

- Operatoren und Ausdrücke
- Anweisungen
- Referenzdatentypen
 - Arrays
 - Klassen
- Statische Methoden
- Strings
- Grundkonzepte der Objektorientierung
 - Klassen und Instanzen mit Attributen und Methoden
 - Sichtbarkeit, Packages
 - Konstruktoren
 - Vererbung und Überschreiben
 - Dynamisches Binden, Polymorphie
 - Abstrakte Klassen, Interfaces
 - Rekursive dynamische Datenstrukturen (Listen)
 - Dateien
 - Realisierung grafischer Benutzungsoberflächen

Literatur

- HABELITZ, Hans-Peter:
Programmieren lernen mit Java.
2. Aufl. Galileo Computing, 2014
- RATZ, Dietmar:
Grundkurs Programmieren in Java.
Hanser Verlag, 2011
- ULLENBOOM, Christian:
Java ist auch eine Insel: Insel 1: Das umfassende Handbuch.
11. Aufl., Galileo Computing, 2014
- INDEN, Michael:
Der Weg zum Java-Profi: Konzepte und Techniken für die professionelle Java-Entwicklung.
2. Aufl. dpunkt.verlag, 2012
- SAAKE, Gunter; SATTLER, Kai-Uwe:
Algorithmen und Datenstrukturen: Eine Einführung mit Java.
5. Aufl., dpunkt.verlag, 2014
- STEYER, Ralph:
Einführung in JavaFX: Moderne GUIs für RIAs und Java-Applikationen.
Springer-Vieweg, 2014

4.7.2 Übg. Programmstrukturen 2

Lehrveranstaltung	Übg. Programmstrukturen 2
Dozent(en)	Gerit Kaleck
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel

ECTS	2.0
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner

Lernziele

Die Studierenden erwerben in der Übung die ...

- Fähigkeit zum praktischen Einsatz der Basiskonzepte objektorientierter Programmiersprachen sowie ihrer Umsetzung in Java.
- Fähigkeit zum Aufbau einfacher dynamischer Datenstrukturen im Kontext einer objektorientierten Programmiersprache und Fähigkeit der Anwendung grundlegender Algorithmen auf diesen Datenstrukturen.
- Fähigkeit zur Nutzung einer aktuellen, verbreiteten Entwicklungsumgebung.
- Fähigkeit zur Realisierung von vollständigen Software-Systemen kleineren Umfangs ausgehend von einer verbalen Aufgabenstellung.
- Fähigkeit zur Software-Entwicklung im kleinen Team.
- Fähigkeit zur Ermittlung geeigneter Testfälle zur Qualitätssicherung.
- Kenntnis der Grundregeln zur Gestaltung benutzungsgerechter Oberflächen und bedienerfreundlicher Software.

Inhalt

- Einführung in die Programmierung mit Java und die Entwicklungsumgebung.
- Anwendung der in der Vorlesung vorgestellten Grundkonzepte der objektorientierten Programmierung durch das Lösen verbal formulierter Aufgabenstellungen in kleinen Teams.
- Testen und Präsentieren der sauber strukturierten Lösung.

Literatur

- HABELITZ, Hans-Peter: Programmieren lernen mit Java - Keine Vorkenntnisse erforderlich, Galileo Computing, 2014 (2. Auflage) ISBN-13: 978-3836228626
- ULLENBOOM, Christian: Java ist auch eine Insel: Insel 1: Das umfassende Handbuch, Galileo Computing, 2014 (11. Auflage) ISBN-13: 978-3836228732

4.8 Programmierpraktikum

B036 Programmierpraktikum

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B036
Modulbezeichnung	Programmierpraktikum
Lehrveranstaltung(en)	B036a Programmierpraktikum
Modulverantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul baut auf die im Modul „Programmstrukturen 2“ erworbenen Programmierkompetenzen auf. Es bildet die Grundlage für Module von Informatik-Studiengängen, in denen Programmierung von Softwareeinheiten größeren Umfangs und softwaretechnische Aspekte eine Rolle spielen, zum Beispiel die Module „Software-Design“, „Software-Projekt“ und „Software-Qualität“.
SWS des Moduls	0
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 2 Stunden Eigenstudium: 148 Stunden
Voraussetzungen	Kenntnis der zentralen Konzepte der Objektorientierten Programmierung und der Programmiersprache Java, Grundkenntnisse in der Benutzung einer Entwicklungsumgebung.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Praktikumsbericht / Protokoll
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Die Studierenden verfügen nach Abschluss des Moduls über die Fähigkeit, aus einer textuellen, problemorientiert formulierten Aufgabenstellung die wesentlichen funktionalen Anforderungen an eine zu entwickelnde Software abzuleiten und in Form eines Pflichtenheftes zu dokumentieren. Sie sind zur Konzeption, insbesondere zur angemessenen Modularisierung von Softwaresystemen mittleren Umfangs in der Lage. Sie können die objektorientierte Programmiersprache Java einsetzen, um ein solches Softwaresystem eigenständig zu implementieren. Sie sind fähig, dabei eine moderne Entwicklungsumgebung zu nutzen.

Dabei verfügen sie über das Problembewusstsein im Hinblick auf die benutzungsgerechte Softwaregestaltung und verfügen über Kenntnisse hinsichtlich der Grundregeln von benut-

zungsgerechter Oberflächengestaltung. Sie verfügen zudem über Fähigkeiten zum angemessenen Einsatz von Interaktionselementen in grafischen Oberflächen bei Aufgabenstellungen mittleren Schwierigkeitsgrades.

Sie verfügen über Basiskenntnisse hinsichtlich der Qualitätssicherung von Software in Form einfacher Teststrategien und können diese einsetzen, um die funktionale Korrektheit und ein ausreichendes Maß an Zuverlässigkeit der Software zu gewährleisten.

Die Studierenden besitzen die Fähigkeit, die Entwicklung einer Software und die Software selbst in einer geeigneten Form zu dokumentieren.

4.8.1 Programmierpraktikum

Lehrveranstaltung	Programmierpraktikum
Dozent(en)	Gerit Kaleck
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	5.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, studentische Arbeit am Rechner

Lernziele

Die Studierenden erwerben mit diesem Praktikum die ...

- Fähigkeit zum praktischen Einsatz der Basiskonzepte objektorientierter Programmiersprachen sowie ihrer Umsetzung in Java in einem Softwareprojekt mittleren Umfangs.
- Fähigkeit zur Nutzung einer aktuellen, verbreiteten Entwicklungsumgebung.
- Fähigkeit zur eigenständigen Strukturierung und Realisierung von vollständigen Softwaresystemen mittleren Umfangs ausgehend von einer problemorientierten Aufgabenstellung.
- Fähigkeit zur Softwareentwicklung und -dokumentation.
- Fähigkeit zur Anwendung der Grundregeln benutzungsgerechter Oberflächengestaltung.

Inhalt

- Entwicklung eines vollständigen Softwaresystems mittleren Umfangs in Java ausgehend von einer problemorientierten Aufgabenstellung.
- Strukturierung und Modularisierung des Projektes.
- Eigenständiger Entwurf passender Datenmodelle.
- Benutzungsgerechte Gestaltung der Oberfläche.
- Testen der entstandenen Software und Dokumentation der Tests.
- Erstellen eines Pflichtenhefts, Dokumentation des Programms und Erstellen eines Benutzerhandbuchs.

Literatur

- ULLENBOOM, Christian:
Java ist auch eine Insel: Insel 1: Das umfassende Handbuch,
Galileo Computing, 2014 (11. Auflage)
ISBN-13: 978-3836228732
- ZÖRNER, Stefan:
Softwarearchitekturen dokumentieren und kommunizieren: Entwürfe, Entscheidungen
und Lösungen nachvollziehbar und wirkungsvoll festhalten
Carl Hanser Verlag, 2012
ISBN-13: 978-3446429246
- PRESSMAN, Roger S.; MAXIM, Bruce:
Software Engineering: A Practitioners Approach

Mcgraw Hill Book Co, 2014
ISBN-13: 978-0078022128

4.9 Grundlagen der Theoretischen Informatik

B022 Grundlagen der Theoretischen Informatik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B022
Modulbezeichnung	Grundlagen der Theoretischen Informatik
Lehrveranstaltung(en)	B022a Grundlagen der Theoretischen Informatik
Modulverantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul baut auf dem Modul „Mathematische Konzepte und Diskrete Mathematik“ auf. Es liefert die Voraussetzungen für das Modul „Systemsoftware“. Zudem liefert es die Grundlagen für ein tieferes Verständnis des Programmierens und damit der Module „Programmstrukturen 1 und 2“, „Algorithmen und Datenstrukturen“, „Systemnahe Programmierung“, „Fortgeschrittene Objektorientierte Programmierung“.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Die Studierenden müssen elementare Logik und Mengenlehre sowie das darauf aufbauende Verständnis von Relationen und Funktionen beherrschen. Ferner wird eine gute Beherrschung elementarer Beweisprinzipien vorausgesetzt. Ein Grundverständnis von Graphentheorie sollte ebenfalls vorhanden sein.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls können die Studierenden umgangssprachliche Sachverhalte sicher in eine logische Formelsprache übertragen. Sie beherrschen die Begriffe und Modelle von Automaten, formalen Sprachen und Algorithmen.

Sie können diese Begriffe mit Beispielen belegen und vorgegebene Beispiele richtig einordnen. Die Studierenden können den Zusammenhang zwischen formalen Sprachen und Automaten bestimmten Typs benennen und an Beispielen belegen. Sie verstehen die grundlegende Bedeutung der Berechenbarkeit und Komplexität von Problemen für die Praxis und können die wesentlichen Zusammenhänge an Beispielen benennen.

4.9.1 Grundlagen der Theoretischen Informatik

Lehrveranstaltung	Grundlagen der Theoretischen Informatik
Dozent(en)	Sebastian Iwanowski
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assig. .
ECTS	5.0
Lehr- und Medienform(en)	Handout, Tutorien

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Grundkenntnisse von formalen Konzepten der Informatik.
- Anwendungssicherheit beim Umformen von logischen Formeln.
- Fähigkeit, Programme formal zu entwerfen und zu verifizieren, im Detail mit Hoare-Tripeln und vollständiger Induktion.
- Kenntnis der Grenzen dieser Konzepte.
- Kenntnisse über verschiedene Grammatiken und Sprachklassen.
- Kenntnisse über Eigenschaften abstrakter Automaten.
- Verständnis über den Zusammenhang zwischen Automaten und Sprachen.
- Kennen der Grundbegriffe der Programmsyntax und von Compilern.

Inhalt

- Formale Logik und Anwendungen auf Alltagsaufgaben
 - Umwandlung beliebiger aussagenlogischer Formeln in konjunktive Normalform
 - Beschreiben von Alltagsaussagen mit prädikatenlogischen Formeln
- Verifikationstechniken für Programmstrukturen
 - Verifikation von Zuweisungen mit Hoare-Tripeln
 - Verifikation von Verzweigungen mit Hoare-Tripeln
 - Verifikation von Schleifen mit vollständiger Induktion
 - Verifikation von rekursiven Prozeduren mit vollständiger Induktion
 - Rekursionstypen und ihre Bedeutung für Compiler
- Endliche Automaten
 - mit und ohne Ausgabe
 - Deterministische und nichtdeterministische Automaten
 - Reguläre Sprachen
 - Reguläre Ausdrücke
 - Äquivalenz und Minimierung endlicher Automaten
 - Anwendungen endlicher Automaten

- Pumping-Lemma für reguläre Sprachen
- Nicht-reguläre Sprachen
- Formale Sprachen
 - Ersetzungssysteme
 - Grammatiken
 - Chomsky-Hierarchie
 - Kontextfreie Grammatiken
 - Normalformen
 - Pumping-Lemma für kontextfreie Sprachen
 - Abschlusseigenschaften kontextfreier Sprachen
 - Entscheidbarkeit bei kontextfreien Sprachen
 - Eindeutigkeit und Mehrdeutigkeit von Grammatiken
 - Syntaxanalyse von Programmiersprachen
 - Kellerautomaten
 - Kontext-sensitive Sprachen
 - Linear-beschränkte Automaten
 - Turingmaschinen
- Einführung in Berechenbarkeit und Entscheidbarkeit
 - Probleme und Algorithmen
 - Unvollständigkeitssatz von Gödel
 - Halteproblem
 - Bedeutung für die Praxis
- Komplexität von Problemen
 - Komplexitätsmaße
 - Polynomzeit-Probleme
 - die Klassen P und NP
 - NP-Vollständigkeit

Literatur

- Roland Backhouse: Programmkonstruktion und Verifikation, Hanser 1989, ISBN 3-446-15056-0, Englische Neuauflage: Program Construction: Calculating Implementations from Specifications, Wiley 2003, ISBN 0470848820
- David Harel / Yishai Feldman: Algorithmik, Springer 2006, ISBN 3-540-24342-9
- John E. Hopcroft / Rajeev Motwani / Jeffrey D. Ullman: Einführung in die Automaten-theorie, Formale Sprachen und Komplexitätstheorie. 2. überarb. Aufl. München: Addison-Wesley Longman Verlag, 2002, ISBN 978-3827370204
- Michael Huth / Mark Ryan: Logic in Computer Science, Cambridge University Press 2004 (2. Auflage), ISBN 052154310X
- Rainer Lang: Theoretische Informatik. Vorlesungsskript, FH-Wedel, 2008
- Uwe Schöning: Logik für Informatiker, Spektrum 2000 (5. Auflage), ISBN 3-8274-1005-3
- Gottfried Vossen / Kurt-Ulrich Witt: Theoretische Informatik, Verlag Vieweg & Teubner 2004 (3. Auflage), ISBN 978-3528231477

- Wegener, Ingo: Theoretische Informatik - eine algorithmenorientierte Einführung. 2. Aufl. Stuttgart: B., G. Teubner Verlag, 1999, ISBN 978-3519121237

4.10 Rechnerstrukturen und Digitaltechnik

B023 Rechnerstrukturen und Digitaltechnik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B023
Modulbezeichnung	Rechnerstrukturen und Digitaltechnik
Lehrveranstaltung(en)	B023a Digitaltechnik 2 B023a Rechnerstrukturen
Modulverantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Informatik (Bachelor) IT-Engineering (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Rechnerstrukturen und Digitaltechnik“ baut auf den im Modul „Einführung in Digitaltechnik“ erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul „Rechnerstrukturen und Digitaltechnik“ erworbenen Kompetenzen stellen die Grundlagen für zum Beispiel die Module „Diskrete Systeme“ und „Systementwurf mit VHDL“ dar. Das Modul kann sinnvoll mit den Modulen, die einerseits Grundlagen der Digitaltechnik beleuchten und andererseits ein Rechnersystem auf höheren Abstraktionsebenen (über dem Gatterniveau) behandeln, kombiniert werden.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Es werden Kompetenzen aus dem Modul „Einführung in Digitaltechnik“ vorausgesetzt.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Aufbauend auf den im Modul „Einführung in Digitaltechnik“ bekommen die Studierenden das Verständnis für zustandsbehaftete Systeme und deren Implementierung in Form von Schaltwerken vermittelt. Die Abstraktion mathematischer Darstellung von Zustandsautomaten ist nach dem Absolvieren des Moduls mit ihren technischen Abbildern versehen. Dabei lernen die Studierenden zu begreifen, worin die Unterschiede zwischen Modellen und realen Schaltungen und Systemen bestehen, warum Abstraktionen und modellhafte Darstellungen unvermeidlich sind und wo deren Grenzen liegen. Aufbauend auf den einfacheren Schaltungen werden Rechnersysteme als komplexe Vertreter digitaler Systeme betrachtet. Die Studierende lernen den Aufbau und die Funktionsweise moderner Rechner kennen und mit den Begriffen

und Konzepten aus dem Bereich Rechnerarchitektur sicher umzugehen. Sie sollten in die Lage versetzt werden, Abläufe in Hardware eines modernen Rechners zu begreifen und klassische sowie innovative Architekturkonzepte zu erkennen und richtig einzuordnen. Ein wesentliches Lernziel besteht außerdem in Erkennung der Bedeutung der Zeitverhaltens von einfachen logischen Gattern und Schaltungen und deren Einflusses auf die Leistungsfähigkeit digitaler Systeme (Verzögerungs-, Setz und Haltezeiten, Taktfrequenz, Steigerung des Durchsatzes in modernen Rechnerarchitekturen).

4.10.1 Digitaltechnik 2

Lehrveranstaltung	Digitaltechnik 2
Dozent(en)	Sergei Sawitzki
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout

Lernziele

Die Studierenden ...

- ergänzen und erweitern die Grundlagen aus der Lehrveranstaltung Digitaltechnik 1
- erlangen Kenntnisse über Aufbau und Funktionsweise von Speicherelementen und Schaltwerken
- erlernen die Methoden zur Analyse, Darstellung und Vereinfachung von endlichen Zustandsautomaten
- begreifen ein Schaltwerk als technische Umsetzung eines endlichen Zustandsautomaten
- gewinnen eine Einsicht in die Methoden der Zeitverhaltensanalyse und Zeitverhaltensoptimierung von digitalen Systemen, sowie lernen das Zeitverhalten und die Zeitvorgaben beim Entwurf digitaler Systemen zu berücksichtigen;
- erlangen die Fähigkeit, digitale Systeme in der Gesamtheit verschiedener Aspekte zu begreifen, die für ihren praktischen Einsatz eine Rolle spielen (Schnittstellen, Komplexität, Zeitverhalten, Leistungsaufnahme, usw.)
- erlangen die Fähigkeit, digitale Systeme mittlerer Komplexität zu begreifen, zu spezifizieren, zu entwerfen und zu optimieren.

Inhalt

- Einleitung
 - Lernziele
 - Organisatorisches
 - Literatur
 - Bezeichnungen und Konventionen
 - Einordnung und historische Entwicklung
 - Voraussetzungen
- Schaltwerke
 - Einleitung und Grundbegriffe, Definitionen
 - Speicherelemente
 - Analyse
 - Synthese
 - Zusammenschaltung

- Transformationen
- Zustandskodierung
- Zustandsminimierung
- Realisierung, Beispiele
- Zeitverhalten
 - Zeitverhalten von Schaltnetzen
 - Modellierung der Gatter- und Leitungsverzögerungen
 - Statische Timing-Analyse (STA)
 - Zeitverhalten von Schaltwerken
 - Metastabilität

Literatur

- Hoffmann, Dirk: Grundlagen der technischen Informatik, Carl Hanser Verlag 2007
- Schiffmann, Wolfram; Schmitz, Robert: Technische Informatik, in 3 Bänden. 3. Auflage Springer Verlag, 1996
- Veendrick, Harry: Nanometer CMOS ICs, Springer 2008
- Rabaey, Jan; Chandrakasan, Anantha; Nikolic, Borivoje: Digital Integrated Circuits, A Design Perspective, 2nd edition, Prentice Hall 2003
- Beuth, Klaus: Elektronik 4. Digitaltechnik, 13. Auflage, Vogel Verlag und Druck 2003

4.10.2 Rechnerstrukturen

Lehrveranstaltung	Rechnerstrukturen
Dozent(en)	Dennis Säring
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, E-Learning

Lernziele

Die Studierenden ...

- kennen grundlegende Rechnerarchitekturkonzepte, die beschreiben, wie verschiedene Baugruppen von Rechnern zusammenarbeiten und wie sich unterschiedliche Rechner-systeme voneinander unterscheiden
- haben die Funktionselemente von Rechnern mit ihren typischen Systemeigenschaften und deren Abbildung auf ein bestimmtes Architekturmodell gelernt
- können das Zusammenwirken der beteiligten Hardware- und Softwarekonzepte im Rahmen einer Aufgabe zur Informationsverarbeitung einschätzen
- besitzen ein Verständnis für Ansätze zur Steigerung der Systemleistung insbesondere unter Berücksichtigung der Aspekte von Parallelität

Inhalt

- Entwicklung der Rechnerarchitekturen

- Grundlegende Rechnerstrukturen
- Architekturkonzepte nach von Neumann
- Mikroprogrammierung
- Moderne parallele und nicht-sequentielle Architekturkonzepte
- Mehrprozessorsysteme
- Aktuelle und zukünftige Entwicklungen

Literatur

- Märtin: Einführung in die Rechnerarchitektur, Fachbuchverlag Leipzig, 2003
- Oberschelp, Gossen: Rechneraufbau und Rechnerstrukturen, Verlag Oldenbourg 1998
- van de Goor: Computer Architecture and Design, Verlag Addison Wesley, 1989
- Müller-Schloer, Schmitter: RISC-Workstation Architekturen, Verlag Springer 1991
- Ungerer: Datenfluß-Rechner, Verlag Teubner, 1993

4.11 Grundlagen der Mathematik 2

B019 Grundlagen der Mathematik 2

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B019
Modulbezeichnung	Grundlagen der Mathematik 2
Lehrveranstaltung(en)	B019a Grundlagen der Linearen Algebra B019a Grundlagen der Statistik
Modulverantwortliche(r)	Dr. Ioana Serban
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Grundlagen der Mathematik 2“ ist ein Einführungsmodul. Zusammen mit dem Modul „Grundlagen der Mathematik 1“, stellt es die Grundlage für nahezu alle quantitativ ausgerichteten weiter führenden Module und Veranstaltungen des Studienverlaufs dar.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Vorausgesetzt wird ein solides Schulwissen der Mathematik und mindestens durchschnittliche mathematische Begabung.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

In diesem Modul werden grundlegende mathematische Kenntnisse aus den Bereichen lineare Algebra und Statistik, wie sie als Grundlage für ein quantitativ ausgerichtetes Studium unerlässlich sind, vermittelt. Der Fokus liegt auf der Vektor- und Matrizenrechnung, linearen Gleichungssystemen, statistischer Datenanalyse, Hypothesentests und wissenschaftlicher Versuchsauswertung.

Die Lernenden sind in der Lage ausgewählte Problemstellungen aus den Bereichen Technik, Naturwissenschaft, Ökonomie und Informatik, mittels der im Modul vermittelten mathemati-

schen Methoden zu modellieren und analysieren. Die Lernenden können für die vermittelten Inhalte praxisrelevante Anwendungsbeispiele benennen. Die Lernenden können eigenständig Lösungsmethoden für ausgewählte Problemstellungen auswählen, die Lösungsmethodik bis zum Ergebnis durchführen und die erhaltenen Ergebnisse kritisch bewerten.

4.11.1 Grundlagen der Linearen Algebra

Lehrveranstaltung	Grundlagen der Linearen Algebra
Dozent(en)	Ioana Serban
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.5
Lehr- und Medienform(en)	-

Lernziele

Nach dem erfolgreichen Besuch der Vorlesung sind die Lernenden in der Lage ...

- lineare algebraische Gleichungssysteme mittels des Gauß-Algorithmus in die Lösbarkeitskategorien (eindeutig lösbar, unendlich viele Lösungen, unlösbar) einzuteilen und ggfs. die Lösung anzugeben.
- die Techniken und Methoden der Vektorrechnung anzuwenden.
- die Techniken und Methoden der Matrixrechnung anzuwenden.
- die Determinante einer niedrigdimensionalen Matrix zu berechnen und den Zusammenhang der Determinante zur Lösungstheorie linearer Gleichungssysteme herzustellen.
- den Simplex-Algorithmus auf einfache lineare Optimierungsprobleme anzuwenden.
- Einfache technische oder ökonomische Systeme mittels der Techniken und Methoden der linearen Algebra zu modellieren und aus der ermittelten Lösung der mathematischen Formulierung das System quantitativ zu beurteilen.

Inhalt

- Lineare algebraische Gleichungssysteme
 - Gauß-Algorithmus
 - Systematisierung des Lösungsverhaltens
 - Unterbestimmte Systeme
- Matrixrechnung
 - Matrixalgebra
 - Inverse Matrix
 - Matrixgleichungen
 - Zusammenhang mit linearen Gleichungssystemen
- Determinanten
 - Definition
 - Zusammenhang mit linearen Gleichungssystemen
- Vektorrechnung
 - Geometrische Vektoren
 - Rechenregeln

- Lineare (Un-)Abhängigkeit
- Rang einer Matrix
- Nochmal Gleichungssysteme, Rangkriterium

- Simplex-Algorithmus

Literatur

- PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler,
Band 2, Teil I. 13. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2012
- HELM, Werner; PFEIFER, Andreas; OHSER, Joachim:
Mathematik für Wirtschaftswissenschaftler.
1. Aufl. München: Carl Hanser Verlag 2011
- GRAMLICH, Günter:
Lineare Algebra: Eine Einführung.
1. Aufl. München: Carl Hanser Verlag 2011
- TESCHL, Gerald; TESCHL, Susanne:
Mathematik für Informatiker,
Band 1: Diskrete Mathematik und lineare Algebra.
3. Aufl. Heidelberg: Springer Verlag 2008
- FISCHER, Gerd:
Lineare Algebra: Eine Einführung für Studienanfänger.
18. aktualisierte Aufl. Wiesbaden: Springer Verlag 2014

4.11.2 Grundlagen der Statistik

Lehrveranstaltung	Grundlagen der Statistik
Dozent(en)	Michael Anders
Hörtermin	2
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, E-Learning

Lernziele

Nach der Lehrveranstaltung können die Studierenden ...

- Statistische Daten verdichten und graphisch aussagekräftig darstellen.
- Mit diskreten und kontinuierlichen Verteilungen umgehen, mit bedingten Wahrscheinlichkeiten korrekt umgehen und diese verstehen.
- Zentralen Grenzwertsatz verstehen und anwenden.
- Konfidenzintervalle berechnen und Hypothesen testen.
- Herleitung der Formeln für lineare Regression nachvollziehen und lineare Regression verstehen.

Inhalt

- Beschreibende Statistik

- Wahrscheinlichkeitsrechnung
 - diskrete und stetige Verteilungen
 - zentraler Grenzwertsatz
 - Konfidenzintervalle
 - Testen von Hypothesen
 - Chi-Quadrat Anpassungstest
 - Regression und Korrelation
-

Literatur

- Spiegel, Murray R.; Stephens, Larry J.: Statistik.
1. Aufl. Bonn: Mitp-Verlag, 2003.
- Fahrmeyr, Ludwig; Künstler, Rita; Pigeot, Iris; Tutz, Gerhard: Statistik.
7. Aufl. Berlin: Springer, 2009.

4.12 Datenschutz und Wirtschaftsprivatrecht

B042 Datenschutz und Wirtschaftsprivatrecht

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B042
Modulbezeichnung	Datenschutz und Wirtschaftsprivatrecht
Lehrveranstaltung(en)	B042a Datenschutz B042a Wirtschaftsprivatrecht
Modulverantwortliche(r)	Prof. Dr. Gerd Beuster
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Keine.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Das Modul setzt ein grundlegendes Verständnis des deutschen Rechtssystem und seiner Begrifflichkeiten sowie elementare Kenntnisse über die Grundprinzipien deutscher Gesetzgebung voraus.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Wirtschaftliche Vorgänge sind in einem rechtlichen Ordnungsrahmen eingebettet. Nach Abschluss des Moduls kennen die Studierenden diese grundlegenden rechtlichen Anforderungen für ausgewählte Bereiche. Hierdurch sind die Studierenden in der Lage, rechtliche Anforderungen in typische Wirtschaftsvorgänge einzuordnen und zu verstehen. Sie sind somit fähig, Praxisfälle rechtlich zutreffend einzuordnen und unter Heranziehung einschlägiger Rechtsnormen zu würdigen. Die Studierenden sind ferner fähig, in ihrem späteren Wirkungskreis (datenschutz- und wirtschafts-) rechtliche Fragestellungen einzuordnen, um bei Bedarf auf Spezialistenunterstützung gezielt zurückgreifen zu können.

4.12.1 Datenschutz

Lehrveranstaltung	Datenschutz
Dozent(en)	Steffen Weiß
Hörtermin	2
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_IMCA16.0, B_Inf14.0, B_WInf14.0, B_WIng14.0, B_CGT14.0, B_EComI14.0, B_EComW14.0) Wahl (B_ITE15.0, B_MInf14.0)
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout

Lernziele

Die Studierenden sind fähig, in ihrem späteren Wirkungskreis datenschutzrechtliche Fragestellungen einzuordnen, um bei Bedarf auf Spezialistenunterstützung gezielt zurückgreifen zu können.

Inhalt

- Gesetzliche Grundlagen des Datenschutzes
 - Anwendung und praktische Umsetzung des Bundesdatenschutzgesetzes (BDSG)
 - Wesentliche Grundlagen aus ausgewählten bereichsspezifischen und bereichsübergreifenden Datenschutzgesetzen
 - Rechte, Pflichten und Aufgaben des betrieblichen Datenschutzbeauftragten zur Einrichtung des Datenschutzmanagements
 - Datenschutz in der Werbepraxis
- Technisch-organisatorischer Datenschutz
 - Grundanforderungen und Grundfunktionen der IT-Sicherheit in Bezug auf die Anforderungen der Datenschutzgesetze
 - Risikomanagement und Schlüsseltechnologien zur Realisierung des technisch-organisatorischen Datenschutzes
 - Kosten-/Nutzen des Datenschutzes
 - Verfahren zur Umsetzung des gesetzlichen Anforderungen des technisch-organisatorischen Datenschutzes
 - Auswahlverfahren zu geeigneten und angemessenen IT-Sicherheitsmechanismen

Literatur

- Bundesdatenschutzgesetz (BDSG) vom Januar 2003, novelliert im Juli 2009
- Koch (Hrsg.):
Handbuch des betrieblichen Datenschutzbeauftragten. 4. Aufl. Frechen: Datakontext
- MÜNCH, Peter:
Technisch-organisatorischer Datenschutz. 4. Aufl. Frechen: Datakontext, 2010

4.12.2 Wirtschaftsprivatrecht

Lehrveranstaltung	Wirtschaftsprivatrecht
Dozent(en)	Felix Reiche
Hörtermin	2

Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_IMCA16.0, B_Inf14.0, B_WInf14.0, B_WIng14.0) Wahl (B_ITE15.0)
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Einfache Sachverhalte des Zivilrechts rechtlich zutreffend zuordnen und unter Heranziehung einschlägiger Gesetzestexte würdigen können.

Inhalt

- Zivilrecht
- BGB
 - Allgemeiner Teil
 - Recht der Schuldverhältnisse
 - Sachenrecht
- HGB
 - Handelsstand
 - Handelsgeschäfte
- Gesellschaftsrecht
 - Personenhandelsgesellschaften
 - Juristische Personen
- Wettbewerbsrecht / Arbeitsrecht / Prozeßrecht werden fragmentarisch mit bearbeitet

Literatur

- MÜSSIG, Peter:
Wirtschaftsprivatrecht.
17. Aufl. Heidelberg: C.F. Müller, 2014.
- Bürgerliches Gesetzbuch.
Auflage 2014: dtv Beck-Texte,
- Handelsgesetzbuch.
Auflage 2014: dtv Beck-Texte,

4.13 Algorithmen und Datenstrukturen

B040 Algorithmen und Datenstrukturen

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B040
Modulbezeichnung	Algorithmen und Datenstrukturen
Lehrveranstaltung(en)	B040a Algorithmen und Datenstrukturen B040b Übg. Algorithmen & Datenstrukturen
Modulverantwortliche(r)	Prof. Dr. Uwe Schmidt
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll zu kombinieren mit Modulen über „Software-Design“ und objektorientierte Programmierung.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzungen für das Verständnis sind grundlegende Kenntnisse über strukturiertes und objektorientiertes Programmieren und Grundkenntnisse der Programmiersprache Java.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B040a), Abnahme (Teil B040b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls haben die Studierenden die praktische Verwendung von wesentlichen Sprachelementen der Programmiersprache Java und von objektorientierten Konzepten gefestigt. Sie kennen die Abläufe von strukturierten und objektorientierten Sprachkonstrukten und deren Kosten, Zeit und Speicher bei der Ausführung auf Neumann-Rechnern.

Ferner können die Studierenden sicher mit dynamischen Datenstrukturen, mit Referenzen und der dynamischen Speicherverwaltung umgehen. Sie verfügen über grundlegende Kenntnisse von Algorithmen für Felder, Matrizen, für Such- und Sortieralgorithmen und für Algorithmen zur Implementierung von Mengen, Verzeichnissen und hierarchischen Strukturen.

Nach Abschluss des Moduls sind die Studierenden in der Lage, mit der Komplexitätstheorie mit qualitativer Abschätzung der Laufzeit- und Speicherplatzeffizienz der verschiedenen Algorithmen zu arbeiten und diese anzuwenden.

4.13.1 Algorithmen und Datenstrukturen

Lehrveranstaltung	Algorithmen und Datenstrukturen
Dozent(en)	Uwe Schmidt
Hörtermin	3
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration

Lernziele

Die Studierenden ...

- lernen die Abläufe und deren Kosten (Zeit / Speicher) bei der Ausführung von Programmen in höheren Programmiersprachen auf den von Neumann-Rechnern kennen.
- erlangen einen sicheren Umgang mit dynamischen Datenstrukturen und Referenzen.
- erlangen grundlegende Kenntnisse über Algorithmen für Such- und Sortieralgorithmen und zur Implementierung von Mengen und Verzeichnissen.
- können die Komplexitätstheorie mit qualitativer Abschätzung der Laufzeit- und Speicherplatzeffizienz der vorgestellten Algorithmen praktisch anwenden.

Inhalt

- Dynamische Datenstrukturen
 - Verkettete Listen
 - Binäre Suchbäume
 - Vorrang-Warteschlangen
 - Hash-Tabellen
 - destruktive und persistente Datenstrukturen
- Such- und Sortieralgorithmen
 - Speicherplatz und Zeitabschätzungen
- Methoden als Daten
 - Verarbeitung aller Elemente eines Containers

Literatur

- Uwe Schmidt:
Algorithmen und Datenstrukturen in C, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~si/vorlesungen/c/c.html>
- Saake, G.; Sattler, K.-U.: Algorithmen und Datenstrukturen, Eine Einführung mit Java, 2004, dpunkt Verlag,
- Okasaki, Chris: Purely Functional Data Structures 1999, Cambridge University Press, ISBN 0-521-66350-4
- Robert Sedgewick, Kevin Wayne: Algorithms, 2011, 4th Revised edition Addison-Wesley Educational Publishers Inc, ISBN 978-0-321-57351-3

4.13.2 Übg. Algorithmen & Datenstrukturen

Lehrveranstaltung	Übg. Algorithmen & Datenstrukturen
Dozent(en)	Malte Heins
Hörtermin	3
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- vertiefen die Beherrschung der Programmiersprache Java und der objektorientierten Programmierung.
- erlangen die Fähigkeit zur Erstellung algorithmenorientierter Programme in Java.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache Java behandelt, die nicht Bestandteil der Vorlesung sind wie z. B. Dateieingabe und -ausgabe.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Algorithmen und Datenstrukturen

4.14 Systemnahe Programmierung

B043 Systemnahe Programmierung

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B043
Modulbezeichnung	Systemnahe Programmierung
Lehrveranstaltung(en)	B043a Systemnahe Programmierung B043b Übg. Systemnahe Programmierung
Modulverantwortliche(r)	Prof. Dr. Uwe Schmidt
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll mit dem Modul „Algorithmen und Datenstrukturen“ und dem Modul „Systemsoftware“ zu kombinieren.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzung sind grundlegende Kenntnissen im Umgang mit strukturierten und objektorientierten Programmiersprachen und grundlegende Kenntnisse über den Aufbau von Rechnern.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B043a), Abnahme (Teil B043b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Ziele dieses Moduls sind das praktisch sichere Beherrschen der wesentlichen Sprachelementen der Programmiersprache C. Weiter ist ein Verständnis vorhanden über die Abläufe in einem Rechner bei der Ausführung von Anweisungen und Operationen aus einer höheren Programmiersprache.

Die Studierenden kennen die wesentlichen Punkte des C-Laufzeitsystems, den Laufzeitkeller für die Speicherorganisation für rekursive Prozeduren und Funktionen, und die Halde für dynamische Datenstrukturen.

Weiter besitzen sie eine Vorstellung von der Repräsentation der Datentypen aus höheren Programmiersprachen wie Mengen, Felder, Verbunde, Zeiger/Referenzen, in einer Maschine. Eine weitere Software-technische Kompetenz besteht in dem Verständnis über die Gefahren und Fehlerquellen beim Arbeiten mit maschinennahen und ungetypten oder nur schwach getypten Sprachen, und über die Bedeutung der großen notwendigen Sorgfalt bei der Software-

Entwicklung.

4.14.1 Systemnahe Programmierung

Lehrveranstaltung	Systemnahe Programmierung
Dozent(en)	Uwe Schmidt
Hörtermin	3
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration

Lernziele

Die Studierenden ...

- beherrschen praktisch sicher die wesentlichen Sprachelementen der Programmiersprache C.
- verstehen die Abläufe in einem Rechner bei der Ausführung von Anweisungen und Operationen aus einer höheren Programmiersprache.
- erlangen das Wissen über die Repräsentation der Datentypen aus höheren Programmiersprachen wie Mengen, Felder, Verbunde, Zeiger / Referenzen in einer Maschine.

Inhalt

- Grundkonzepte der Sprache C
 - Einfache Datentypen
 - Präprozessor
 - Anweisungen
 - Ausdrücke
- Strukturierte Datentypen
 - Felder und Zeiger
 - struct und union
- Datenstrukturen und Algorithmen für Felder und Matrizen
- Funktionen und Funktionszeiger
 - Prozedurorganisation

Literatur

- Uwe Schmidt: Algorithmen und Datenstrukturen in C, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~si/vorlesungen/c/c.html>
- Harbison, Samuel; Steele, Guy L.: C - A Reference Manual, 5th edition, Prentice Hall, New Jersey, 2002, ISBN: 0-13-089592-X
- Kernighan, Brian W.; Ritchie, Dennis M.: C Programming Language, Prentice Hall, New Jersey, 1998, ISBN: 0-13-110370-9

4.14.2 Übg. Systemnahe Programmierung

Lehrveranstaltung	Übg. Systemnahe Programmierung
Dozent(en)	Malte Heins

Hörtermin	3
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- beherrschen die Programmiersprache C und die maschinennahen Konzepte der Sprache.
- erlangen die Fähigkeit zur Erstellung maschinennaher Programme.
- erlangen das Verständnis über die Abläufe in einer Maschine bei der Ausführung von Sprachkonstrukten aus höheren Programmiersprachen, wie zum Beispiel die Laufzeitor-
ganisation bei Funktionsaufrufen.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache C behandelt, die nicht Bestandteil der Vorlesung sind.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Systemnahe Programmierung

4.15 Datenbanken 1

B052 Datenbanken 1

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B052
Modulbezeichnung	Datenbanken 1
Lehrveranstaltung(en)	B052a Einführung in Datenbanken B052b Übg. Einführung in Datenbanken
Modulverantwortliche(r)	Prof. Dr. Ulrich Hoffmann
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Das Modul komplementiert Einführungen in die Programmierung („Einführung in die Programmierung“, „Programmstrukturen 1“) in allen Studiengängen. Es ist mit den fortgeschrittenen Modulen „Datenbanken 2“ (Bachelor) und „Datenbanken 3“ (Master) kombinierbar. Das Modul sollte in allen Studiengängen verwendet werden, in denen Datenhaltung wesentlich ist.
SWS des Moduls	3
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 29 Stunden Eigenstudium: 121 Stunden
Voraussetzungen	Vorausgesetzt werden Grundkenntnisse in Programmierung und die Fähigkeit, abstrakt zu denken.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B052a), Abnahme (Teil B052b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nachdem Studierende die Veranstaltungen des Moduls besucht haben, haben sie die Fähigkeit, eine relationale Datenbank unter Nutzung von SQL abzufragen, einzurichten und die betriebliche Informationsverarbeitung mittels relationaler Datenbanksysteme unter Nutzung von SQL zu planen und durchzuführen. Zudem haben sie die Fähigkeit, selbständig einen Datenbankentwurfsprozess unter Verwendung des Entity-Relationship-Datenmodells und des relationalen Datenmodells durchzuführen.

4.15.1 Einführung in Datenbanken

Lehrveranstaltung	Einführung in Datenbanken
Dozent(en)	Ulrich Hoffmann
Hörtermin	3
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_MInf14.0, B_WInf14.0, B_WIng14.0) Wahl (B_CGT14.0, B_TInf14.0)
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Handout

Lernziele

Die Studierenden ...

- beherrschen die Grundlagen der relationalen Datenbanktechnologie;
- erlangen die Fähigkeit, selbstständig einen Datenbankentwurfsprozess zu planen, eine relationale Datenbank unter Nutzung von SQL einzurichten und die Informationsverarbeitung mittels relationaler Datenbanksysteme unter Nutzung von SQL durchzuführen;
- erlangen die Fähigkeit, mit einem Entwurfstool einen Datenbankentwurfsprozess durchzuführen und mittels SQL selbstständig Anfragen an ein Datenbanksystem zu stellen.

Inhalt

- Einführung in die Datenbanktechnologie
- Datenbanksprache SQL - Einführung
- Datenbank-Abfrage mit SQL
- Datenbanksprache SQL - Einrichten der Datenbank
- Das Entity-Relationship-Datenmodell
- Das Relationale Datenmodell
 - Relationenschemata und Datenabhängigkeiten
 - Relationale Datenbanken
 - Normalformen
- Datenbank - Lebenszyklus

Literatur

- Elmasri, Ramez; Navathe, Shamkant B.: Grundlagen von Datenbanksystemen. 3. Aufl. München: Pearson -Verlag, 2009.
- Meier, Andreas: Relationale Datenbanken Leitfaden für die Praxis. Berlin: Springer-Verlag, 2004.
- Vetter, Max: Aufbau betrieblicher Informationssysteme mittels konzeptioneller Datenmodellierung. 8. Aufl. Stuttgart: Vieweg-Teubner, 1998.
- Vossen, Gottfried: Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme. 5. Aufl. Oldenbourg: Oldenbourg-Wissenschaftsverlag, 2008.

4.15.2 Übg. Einführung in Datenbanken

Lehrveranstaltung	Übg. Einführung in Datenbanken
Dozent(en)	Marcus Riemer
Hörtermin	3
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_MInf14.0, B_WInf14.0, B_WIng14.0) Wahl (B_CGT14.0, B_TInf14.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	Handout, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit, ein Datenbanksystem mit SQL zu befragen und in nicht-triviale textuelle Anfrageanforderungen in SQL zu überführen.
- haben grundlegende Kenntnisse über die Ausführung der von ihnen gestellten Anfragen.
- haben die Kompetenz, ein Datenbankentwurfswerkzeug grundlegend zu bedienen.

Inhalt

Vorlesungsbegleitende praktische Übungen in SQL und zum Datenbankentwurf

Literatur

Vorlesungsunterlagen

4.16 Rechnernetze

B037 Rechnernetze

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B037
Modulbezeichnung	Rechnernetze
Lehrveranstaltung(en)	B037a Rechnernetze B037b Prakt. Rechnernetze
Modulverantwortliche(r)	Dipl.-Ing. (FH) Ilja Kaleck
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll mit den Inhalten der Grundlagenmodule „Informationstechnik“ und „Programmstrukturen 1 und 2“ zu kombinieren.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Grundlegende Kenntnisse im Bereich der allgemeinen Informationstechnik; grundlegende Programmierkenntnisse in C, Objekt-Pascal (Delphi) oder Java erleichtern das Verständnis für Interprozesskommunikation im Rahmen gezeigter Beispielprogramme; Kenntnisse im Umgang mit aktuellen Desktop-Betriebssystemen (Windows, optional MacOS-X bzw. Linux) sind zur eigenständigen Durchführung praktischer Übungsanteile hilfreich.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B037a), Abnahme (Teil B037b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Beendigung dieses Moduls verfügen die Studierenden über fundiertes Wissen über den Aufbau, den Betrieb und die Arbeitsweise moderner Rechnernetze (Computer Networks); dieses sowohl in technischer Hinsicht als auch in Bezug auf den Ablauf der Kommunikation zwischen Prozessen in Unternehmensnetzen bzw. dem Internet. Die Studierenden beherrschen allgemeine Grundlagen der Datenkommunikation und kennen den Aufbau eines universellen

Kommunikationsmodells, erlernt am Beispiel des OSI-Referenzmodells.

Vertiefendes Wissen haben sie bezüglich des Aufbaus und die Kommunikation in der Internet-Architektur (IPv4, IPv6). Hierbei verfügen sie über Kenntnisse hinsichtlich der Eigenschaften der verfügbaren Transportprotokolle und haben das grundlegende Verständnis zur Realisierung einfacher Interprozesskommunikation.

Sie kennen die für den Betrieb eines IP-basierten Netzes essentiell notwendigen Anwendungsprotokolle und können dieses Wissen auch als Basis für die Gestaltung eigener Anwendungen sinnvoll nutzen. Ferner verfügen sie über Kenntnisse hinsichtlich der Theorie und den praktischen Einsatz von Verzeichnisdiensten zur Verwaltung größerer Netze.

Darüber hinaus haben sie ein hinreichendes Verständnis für den technischen Aufbau und den Betrieb moderner Unternehmensnetze. Hierzu gehören fundierte Kenntnisse über die Eigenschaften aktueller Netztechnologien im Bereich Lokaler Netze (LANs) als auch drahtloser Netze (WLANs).

Sie kennen auch die Arbeitsweise der dabei eingesetzten Koppellemente und deren Vermittlungsstrategien zum Aufbau größerer Netzstrukturen bzw. des Internets.

Durch den praktischen Anteil des Moduls verfügen die Studierenden über ein vertieftes und punktuell auch signifikant ergänztes Wissen zuvor behandelte Lehrinhalte, eigenständig erlernt am eigenen PC-System (Server) im zugehörigen Schulungslabor. Sie verfügen auch über ein praxisnahes Verständnis über den realen Datenfluss in Netzen und können so typische Fehlersituationen im Rahmen der Kommunikation zwischen Anwendungen analysieren und eigenständig beheben. Diese Fähigkeit bildet eine wesentliche Grundlage für eine effiziente Entwicklung verteilter Anwendungen im Rahmen komplexer Softwareprojekte.

4.16.1 Rechnernetze

Lehrveranstaltung	Rechnernetze
Dozent(en)	Ilja Kaleck
Hörtermin	3
Art der Lehrveranstaltung	Pflicht (B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_MInf14.0, B_STec16.0, B_TInf14.0, B_WInf14.0) Wahl (B_CGT14.0)
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Online-Aufbereitung, Softwareredemonstration, E-Learning

Lernziele

Die Studierenden erlangen ...

- ein grundlegendes Verständnis für den Aufbau einer herstellerneutralen Kommunikationsarchitektur (OSI).
- Kenntnisse über den Aufbau und die Funktion des Internet-Architekturmodells.
 - Kenntnis über IPv4-Adress- und Netzstrukturen.
 - Verständnis über die Arbeitsweise essentieller Anwendungsprotokolle.
 - Fähigkeit zum Verständnis des Ablaufs einfacher Interprozesskommunikation, u.a. als Basis für die Realisierung komplexerer verteilter Anwendungen.
 - die Arbeitsweise spezifischer Maßnahmen gegen den IPv4-Adressmangel im IPv4 (NAT, Proxyserver-Dienste) kennen.
 - Wissen über die Eigenschaften des neuen Internet-Protokolls Version 6 (IPv6) und Änderungen an bestehenden Internet-Protokollen (u. a. DNS, ICMP).
- Verständnis über den technischen Aufbau und den Betrieb Lokaler Netze (LANs).
 - Verständnis hinsichtlich des generellen Ablaufs der IP-Kommunikation in LANs.
 - Wissen um die Eigenschaften aktueller Netztechnologien (Schwerpunkt: Ethernet-Technik).
 - Kenntnisse zum Aufbau und Betrieb drahtloser Netze (IEEE 802.11 WLANs).
- Wissen um den technischen Aufbau von Netzstrukturen bzw. des Internets.
 - Wissen um die Aufgabe Funktionsweise der klassischen von Koppelemente in Netzen.
 - elementares Wissen um die Arbeitsweise praxisrelevanter Routingverfahren für kleinere und größere Netze (u. a. einfaches IP-Routing; hierarchisches Routing).
- Grundkenntnisse über den Aufbau und die Funktionsweise von Verzeichnisdiensten.

Inhalt

- Allgemeine Grundlagen und Begriffe
 - Allgemeine Strukturen in der Datenkommunikation
 - Protokolle und Protokollabläufe

- Netztopologien und Klassifizierung von Übertragungsnetzen
- Das ISO-OSI Referenzmodell
 - Prinzip der Schichtenbildung und Schichtenfunktionen im Überblick
 - Datenfluss im Modell
 - Aktuelle Koppелеlemente zum Netzaufbau im Kontext der OSI-Modells
- Die Internet-Architektur
 - Historie, Architekturübersicht, Standardisierungen
 - IPv4-Adressstrukturen und Netzaufbau, Subnetting
 - UDP-/TCP-Kommunikation, Sockets bzw. Socket-Kommunikation
 - Betrachtung ausgewählter Anwendungsprotokolle (DNS, TELNET / SSH, SMTP, HTTP, ...)
 - Network Address Translation (NAT) und der Einsatz von Proxy-Servern
 - Einführung in das neue Internet Protocol Version 6 (IPv6)
 - * Adress- und Netzstruktur, Migrationshinweise
 - * Änderungen an höheren Protokollen in Bezug auf das IPv6
- Technik Lokaler Netze (LANs)
 - Ablauf der Kommunikation in IEEE 802 LANs (Layer-2, IP, inkl. DHCP)
 - Schwerpunkt Betrachtung: Ethernet-Technik, Zugriffsverfahren und
 - Technische Umsetzungen (10Mbps / 100FE / 1GbE / 10GbE)
 - Überblick über andere LAN-Technologien
- Koppелеlemente und Vermittlungstechniken
 - Repeater, Brücken- bzw. Layer-2 Switching-Technologie
 - Virtuelle LANs (VLANs), Class-of-Services im LAN
 - Router bzw. IP-Routing, Link-State und Distanzvektor-Verfahren,
 - Hierarchisches Routing und IP-Multicasting
 - Drahtlose Netze nach IEEE 802.11,
 - * Struktur, Aufbau, Übertragungskonzepte, Sicherheitsbetrachtungen
- Verzeichnisdienste
 - Einführung und grundlegendes Konzept des X.500
 - Herstellerspezifische Lösungen (Active Directory)
 - Lightweight Directory Access Protocol (LDAP)

Literatur

- TANNENBAUM, Andrew S.:
Computer Netzwerke.
5. Aufl. München: Pearson Education, 2012, ISBN 978-3-86894-137-1
- KUROSE, James F.; ROSS, Keith W.:
Computer Netzwerke. Der Top-Down Ansatz.
6. Aufl. : Pearson Education, 2014, ISBN 978-3-86894-237-8
- HALSALL, Fred:
Computer Networking and the Internet.
5. Aufl. München: Addison-Wesley, 2005, ISBN 978-0321263582
- RECH, Jörg:
Ethernet. Technologien und Protokolle für die Computervernetzung.
2. Aufl. Heidelberg: dPunkt-Verlag, 2007, ISBN 978-3-936931-40-2
- RECH, Jörg:
Wireless LANs. 802.11-WLAN-Technologie und praktische Umsetzung im Detail.
4. Aufl. Heidelberg: dPunkt-Verlag, 2012, ISBN 978-3-936931-75-4

- BADACH, Anatol; HOFFMANN, Erwin:
Technik der IP-Netze. Funktionsweise, Protokolle und Dienste.
2. Aufl. München: Hanser, 2007, ISBN 978-3446215016
- DAVIES, Joseph:
Understanding IPv6. Covers Windows 8 and Windows Server 2012.
3rd Edition: Microsoft Press, 2012, ISBN 978-0-7356-5914-8
- SCHÄFER, Günther:
Netzwerksicherheit. Algorithmische Grundlagen und Protokolle.
Heidelberg: dPunkt-Verlag, 2003, ISBN 3-89864-212-7
- SPERZEL Christian:
Netzwerksicherheit. Schützen Sie Ihr Netzwerk vor dem Zugriff anderer
Online-Videotrainig, Video2brain GmbH, 2014,

- BUEROSSE, Jörg:
Sichere E-Mails. Verschlüsselung und digitale Signatur unter Windows, Linux, OS X,
iOS und Android.
Online-Videotrainig, Video2brain GmbH, 2014

- FRISCH; HÖLZEL; LINTERMANN; SCHAÄFER:
Vernetzte IT-Systeme.
6. Aufl.:Bildungsverlag EINS, 2013, ISBN 978-3-8237-1141-4
- GRABA, Jan:
An Introduction to Network Programming with Java, Java 7 Compatible
3rd Edition: Springer-Verlag, 2013, ISBN 978-1-4471-5253-8
- CIUBOTARU, Bogdan ; MUNTEAN, Gabriel-Miro:
Advanced Network Programming - Principles and Techniques. Network Application
Programming with Java.
Springer-Verlag, 2013, ISBN 978-1-4471-5291-0
- HAROLD, Elliotte Rusty:
Java Network Programming. Developing Networked Applications.
4th Edition, OReilly Media, 2013, ISBN 978-1-44935-767-2
- KLÜNTER, Dieter; LASER, Jochen:
LDAP verstehen, OpenLDAP einsetzen. Grundlagen und Praxiseinsatz.
2. Aufl. Heidelberg: dPunkt-Verlag, 2007, ISBN 978-3-89864-263-7

4.16.2 Prakt. Rechnernetze

Lehrveranstaltung	Prakt. Rechnernetze
Dozent(en)	Ilja Kaleck
Hörtermin	3
Art der Lehrveranstaltung	Pflicht (B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_MInf14.0, B_STec16.0, B_TInf14.0, B_WInf14.0) Wahl (B_CGT14.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Online-Aufbereitung, Software-demonstration, studentische Arbeit am Rechner, E-Learning

Lernziele

Die Studierenden erlangen ...

- die Fähigkeit zum praktischen Umgang mit der Internet-Technologie am eigenen PC.
 - die Fähigkeit zum Anschluss von Systemen an ein Unternehmensnetz.
 - die Fähigkeit zur grundlegenden Konfiguration des Internet-Protokolls (IPv4, IPv6).
 - das Verständnis für Sicherheitsrichtlinien auf Multi-User Systemen (Windows, Linux).
 - die Fähigkeit zur Analyse und Behebung typischer Fehlersituationen im Rahmen der Kommunikation von Anwendungen und Systemen im Netz.
 - die Fähigkeit zur Konfiguration grundlegender Internet-Dienste (u. a. DNS, HTTP, FTP).
- das Verständnis für Lösungsansätze aktueller Techniken zur Unix-/Windows Integration in heterogenen Unternehmensnetzen (NFS, SAMBA, X-Windows).
- das Verständnis über aktuelle Konzepte zur Benutzer- und Rechteverwaltung in Netzen.
 - die Fähigkeit zur Benutzerverwaltung mittels eines Domänenkonzeptes (Windows).
 - die Fähigkeit zur Einrichtung von Verzeichnisdiensten (LDAP, Active Directory).
- die Grundkenntnisse zum praktischen Einsatz von Virtualisierungstechniken auf dem Desktop.
 - die Fähigkeit zur Einrichtung einfacher IP-Routingfunktionen auf einem System.
- das Verständnis über den praktischen Aufbau und Betrieb eines WLANs und dessen interne Kommunikationsabläufe (inkl. Sicherheitsbetrachtungen).
- die Fähigkeit zum Einsatz eines LAN-Analyzers zur Analyse von Kommunikationsabläufen zwischen Anwendungen sowie zur Fehleranalyse in LANs und WLANs.
- grundlegende Kenntnisse digitaler Sprachübertragung in Netzen mittels der Voice-over-IP (VoIP) Technik (Wahlthema).
- grundlegende Kenntnisse zu Streaming-Media Technik und den Real-Time Protokollen zur Übertragung multimedialer Inhalte in Netzen (Wahlthema).

Inhalt

Durchführung eines Laborpraktikums durchgängig individuell am eigenen PC-System unter Einsatz dedizierter Wechselfestplatten (Teilnehmer; Arbeitsgruppe)

- Einrichtung eines Server-Betriebssystems und Konfiguration der grundlegenden Kommunikationsprotokolle (IPv4, IPv6).
 - Nutzung typischer Internetdienstprogramme und Betrachtung der dabei verwendeten Protokolle.
- Einsatz von Techniken zur Unix/Windows-Integration (NFS, SAMBA, X-Windows, Unix mit Posix-ACLs)
- Nutzung einfacher Benutzer- und Rechteverwaltung im Netz (Domänenkonzept).
- Einsatz von Virtualisierungstechniken auf dem Desktop

- Aufbau einer lokalen Netzinfrastruktur und Einrichtung des lokalen IP-Routings (inkl. NAT)
- Grundlegende Firewall-Konfiguration
- Einrichten und Arbeiten mit aktuellen Verzeichnisdiensten
 - Aufbau einer eigenen Verzeichnisstruktur (Directory)
 - Formulierung von Suchanfragen an Verzeichnisdienste (Active Directory, LDAP-Server)
- Konfiguration grundlegender Internet-Serverdienste (DNS, FTP, HTTP, Proxy-Server, TELNET / SSH)
 - Nutzung der SSH Port-Forwarding Funktion
- Protokollanalyse und Fehlersuche im LAN mit einem LAN-Analyzer
 - Nutzung einer Remote-Probes zur verteilten LAN-Analyse im Netz.
 - Einfache LAN-Performance Messungen
- Konfiguration einer Arbeitsstation in einem Wireless-LAN (Adhoc und Infrastrukturnetz)
 - Analyse des drahtlosen Daten- und Kontrollverkehrs mit einem WLAN-Analyzer
- Einrichtung eines Voice-over-IP (VoIP) Clients (Wahlaufgabe)
 - Betrachtung dabei genutzter VoIP-Technologien und Übertragungsprotokolle
 - Einsatz eines LAN-Analyzers zur VoIP-Übertragungsanalyse
- Einführung in die Multi-Media Übertragung in Netzen (Wahlaufgabe)
 - Einrichtung eines aktuellen Streaming-Servers
 - Betrachtung der beteiligten Realtime-Übertragungsprotokolle
- Weitere Wahlthemen nach Aktualität.

Literatur

- RECH, Jörg:
Wireless LANs. 802.11-WLAN-Technologie und praktische Umsetzung im Detail.
4. Aufl. Heidelberg: dPunkt-Verlag, 2012, ISBN 978-3-936931-75-4
- BADACH, Anatol:
Voice-over-IP. Grundlagen, Protokolle, Anwendungen, Migration, Sicherheit.
4. Aufl. München: Hanser, 2009, ISBN 978-3-446-41772-4
- LIU/MATTHEW/PARZIALE/DAVIS/FORRESTER/BRITT:
TCP/IP Tutorial and Technical Overview (PDF). 8th. Ed. 2006: IBM-Redbook Serie.
<http://www.redbooks.ibm.com/redbooks/> Aktualisierungsdatum 29.06.2014
- GROUPER IEEE 802.11: Aktuelle Spezifikationen zu IEEE 802.11.
<http://standards.ieee.org/getieee802/802.11.html> Aktualisierungsdatum 29.06.2014
- IETF: Internet-Draft Dokumente und aktuelle RFCs.
<http://www.ietf.org/> - Aktualisierungsdatum 29.06.2014
- CISCO SYSTEMS: Internetworking Technology Handbook.
<http://www.cisco.com/c/en/us/td/docs/internetworking/technology/handbook/itodoc.html>
Aktualisierungsdatum 29.06.2014
- SPERZEL, Christian:
Netzwerksicherheit. Schützen Sie Ihr Netzwerk vor dem Zugriff anderer
Online-Videotrainig, Video2brain GmbH, 2014,
<https://www.video2brain.com/de/videotraining/netzwerksicherheit> - Aktualisierungsdatum 29.06.2014
- BUEROSSE, Jörg:

Sichere E-Mails. Verschlüsselung und digitale Signatur unter Windows, Linux, OS X, iOS und Android.

Online-Videotrainig, Video2brain GmbH, 2014,

<https://www.video2brain.com/de/videotraining/sichere-e-mails> - Aktualisierungsdatum 29.06.2014

- DIVERSE:

Schulungskurse zum Thema “Virtualisierung”.

Online-Videotrainig, Video2brain GmbH, 2013,

<https://www.video2brain.com/de/search.htm?searchentry=Virtualisierung> - Aktualisierungsdatum 29.06.2014

- WOWZA MEDIA SYSTEMS:

Online Dokumentation zur “Wowza Streaming Engine”

<http://www.wowza.com/forums/content.php?188-documentation> - Aktualisierungsdatum 29.06.2014

4.17 Lineare Algebra

B045 Lineare Algebra

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B045
Modulbezeichnung	Lineare Algebra
Lehrveranstaltung(en)	B045a Lineare Algebra
Modulverantwortliche(r)	Dr. Ioana Serban
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Lineare Algebra“ baut auf den in der Veranstaltung „Einführung in die Lineare Algebra“ aus dem Modul „Grundlagen der Mathematik 2“ erworbenen Kenntnissen und Fähigkeiten auf. Die im Modul „Lineare Algebra“ erworbenen Kompetenzen stellen die Grundlage für zum Beispiel die weiterführenden Module „Grundlagen der Computergrafik“, „Systemmodellierung“ oder „Bildbearbeitung und -analyse“ dar.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Das Modul setzt grundlegende Kenntnisse der linearen Algebra voraus, wie sie zum Beispiel im Modul „B019: Grundlagen der Mathematik II“ vermittelt werden.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

In diesem Modul werden weiterführende mathematische Kenntnisse aus dem Bereich der linearen Algebra, wie sie für ein quantitativ ausgerichtetes Studium unerlässlich sind, vermittelt. Der Schwerpunkt liegt dabei auf der Vektorraumtheorie und der analytischen Geometrie. Die Lernenden sind in der Lage ausgewählte Problemstellungen aus den Bereichen Naturwissenschaft, Technik und Informatik, mittels der im Modul vermittelten mathematischen Methoden zu modellieren und analysieren. Die Lernenden können für die vermittelten Inhalte praxisrelevante Anwendungsbeispiele benennen. Die Lernenden können eigenständig Lösungsmethoden für ausgewählte Problemstellungen auswählen, die Lösungsmethodik bis zum Ergebnis durchführen und die erhaltenen Ergebnisse kritisch bewerten

4.17.1 Lineare Algebra

Lehrveranstaltung	Lineare Algebra
Dozent(en)	Ioana Serban
Hörtermin	3
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	5.0
Lehr- und Medienform(en)	-

Lernziele

Nach dem erfolgreichen Besuch der Veranstaltung können die Studierenden ...

- die Determinante einer Matrix beliebiger Dimension berechnen und den Zusammenhang zur Lösungstheorie linearer Gleichungssysteme herstellen.
- die Vektorraumaxiome nennen und eine gegebene Menge mit Verknüpfungen darauf überprüfen ob diese ein Vektorraum (über \mathbb{R} oder \mathbb{C}) ist.
- Die Definition eines Unterraums nennen; Teilmengen von Vektorräumen darauf überprüfen ob diese Unterräume sind.
- das Konzept der linearen Abhängigkeit von Vektoren erklären; Teilmengen von Vektorräumen auf lineare Abhängigkeit überprüfen.
- die Definition einer Basis nennen. Teilmengen von Vektorräumen darauf überprüfen ob diese eine Basis sind.
- die Definition eines Skalarproduktes nennen; verschiedene lineare Abbildungen auf Vektorräumen darauf überprüfen ob diese ein Skalarprodukt sind.
- die Definition einer Norm nennen; den Zusammenhang zwischen Skalarprodukt und Norm nennen.
- Die Definition einer Orthonormalbasis nennen; eine Orthonormalbasis aus einer gegebenen Basis konstruieren (Gram-Schmidt-Verfahren).
- die Parameter und Koordinatendarstellung von Geraden und Ebenen formulieren; Lagebeziehungen zwischen linearen geometrischen Objekten berechnen; Lagebeziehungen zwischen linearen und einfachen nichtlinearen Geometrischen Objekten berechnen.
- die Definition einer linearen Abbildung nennen; lineare Abbildungen mittels Matrix-Vektor-Schreibweise ausdrücken. Eigenschaften gegebener linearer Abbildungen bestimmen.
- die Definition einer affinen Abbildung nennen; affine Abbildungen mittels Matrix-Vektor-Schreibweise ausdrücken. Eigenschaften gegebener affiner Abbildungen bestimmen.
- Koordinatentransformationen als affine Abbildung durchführen; die affine Abbildung einer Koordinatentransformation berechnen; aktive und passive Koordinatentransformationen unterscheiden.
- das charakteristische Polynom einer Matrix aufstellen; die Eigenwerte einer Matrix berechnen; die Eigenvektoren einer Matrix berechnen.

- eine Matrix diagonalisieren.
- bestimmte Funktionen einer Matrix berechnen.

Inhalt

- Wiederholung: Grundlagen der linearen Algebra
- Determinanten
 - der Entwicklungssatz von Laplace
 - lineare Gleichungssysteme
- Vektorräume
 - Definition, Beispiele und Eigenschaften
 - Unterräume
 - Lineare Abhängigkeit, Basis und Dimension
- Euklidische und unitäre Vektorräume
 - Skalarprodukt und Norm
 - Orthogonalität
 - Orthogonal- und Orthonormalbasen
- Analytische Geometrie
 - Darstellung von Geraden und Ebenen
 - Lagebeziehung zwischen linearen geometrischen Objekten
 - Einfache nichtlineare Objekte am Beispiel
- Abbildungen
 - Lineare Abbildungen
 - Affine Abbildungen
 - Koordinatentransformationen
- Eigenwerte und Eigenvektoren
 - Charakteristisches Polynom, Eigenwerte, Eigenvektoren
 - Diagonalisierung
 - Matrixfunktionen

Literatur

- GRAMLICH, Günter M.:
Lineare Algebra: Eine Einführung.
3. aktualisierte Aufl. München: Carl Hanser Verlag 2011
- FISCHER, Gerd:
Lernbuch Lineare Algebra und Analytische Geometrie.
1. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2011
- FARIN, Gerald; HANSFORD, Dianne:
Lineare Algebra: Ein geometrischer Zugang,
Springer Verlag 2003
- FISCHER, Gerd:
Lineare Algebra: Eine Einführung für Studienanfänger.
18., aktualisierte Aufl. Wiesbaden: Springer Verlag 2013

- LIESEN, Jörg; MEHRMANN, Volker:
Lineare Algebra: Ein Lehrbuch über die Theorie mit Blick auf die Praxis.
1. Aufl. Wiesbaden: Vieweg + Teubner Verlag 2011
- ZIESCHANG, Heiner:
Lineare Algebra und Geometrie.
1. Aufl. Stuttgart, Teubner Verlag 1997

4.18 Einführung in die Betriebswirtschaft

B034 Einführung in die Betriebswirtschaft

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B034
Modulbezeichnung	Einführung in die Betriebswirtschaft
Lehrveranstaltung(en)	B034a Einführung in die Betriebswirtschaft
Modulverantwortliche(r)	Prof. Dr. Franziska Bönte
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Das Modul “Einführung in die Betriebswirtschaft” ist ein Einführungsmodul. Die erworbenen Kompetenzen stellen wesentliche Grundlagen für eine Vielzahl weiterer Module dar, wie zum Beispiel “Produktionsmanagement 1”, “Business Planning” oder “Unternehmensführung”.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Keine
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach erfolgreicher Teilnahme an diesem Modul können die Studierenden die Bedeutung von betriebswirtschaftlichen Denkweisen und Methoden für die moderne Unternehmensführung abschätzen. Sie kennen grundlegende Fragestellungen und Methoden zu deren Bearbeitung aus dem Bereich der Allgemeinen Betriebswirtschaftslehre. Die Studierenden sind befähigt, ausgewählte Aufgaben, wie sie sich in der Praxis des Unternehmens ergeben, unter Anwendung betriebswirtschaftlicher Methoden zu lösen.

Die Studierenden können wechselseitige Abhängigkeit zwischen den Aufgaben aus den Bereichen der Betriebswirtschaftslehre, den Ingenieurwissenschaften und der Informatik identifizieren und benennen.

4.18.1 Einführung in die Betriebswirtschaft

Lehrveranstaltung	Einführung in die Betriebswirtschaft
Dozent(en)	Franziska Bönte
Hörtermin	3
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_MInf14.0, B_STec16.0, B_TInf14.0, B_WInf14.0, B_WIng14.0) Wahl (B_CGT14.0, B_ITE15.0)
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	5.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Tutorien

Lernziele

Die Studierenden können ...

- das Erfahrungs- und Erkenntnisobjekt der Betriebswirtschaftslehre benennen,
- die Begriffe Wirtschaften und Ökonomisches Prinzip erklären sowie eine Break-Even-Analyse durchführen,
- Unternehmensziele aufzählen; die Aufgaben der Zielbildung erläutern sowie den Zielbildungsprozess wiedergeben,
- ausgewählte Kennzahlen ausrechnen,
- Ziele der Unternehmensführung erläutern, Führungsebenen voneinander abgrenzen, den Führungsprozess beschreiben sowie ausgewählte Führungsstile erläutern und -prinzipien erklären,
- Standortfaktoren identifizieren und Modelle zur Standortbewertung einsetzen,
- die Ziele der Materialwirtschaft wiedergeben und durch Anwendung von Methoden materialwirtschaftliche Analysen durchführen und Handlungsanweisungen ableiten,
- ausgewählte Erzeugnisstrukturdarstellungen für gegebene Problemstellungen erstellen und mit programmorientierten Verfahren die Materialbedarfsplanung durchführen,
- mit ausgewählten Verfahren die optimale Bestellmenge bestimmen,
- den Input, Throughput und Output von Produktionsprozessen beschreiben,
- das optimale Produktionsprogramm für ausgewählte Fälle ermitteln,
- ausgewählte Aufgaben der Produktionsprozessplanung ausführen,
- die Ziele des Marketings nennen, Methoden zur Ableitung der Marketing-Strategie beschreiben und anwenden sowie die Instrumente des Marketing-Mix erläutern,
- Investitionsarten voneinander abgrenzen; den Investitionsprozess beschreiben und die Aufgabe der Investitionskontrolle skizzieren sowie die Vorteilhaftigkeit einer Investition mittels Methoden beurteilen,
- die Ziele und Aufgaben der Finanzwirtschaft nennen; die Finanzierung aus Abschreibungen erläutern sowie den Financial-Leverage-Effekt an einem Beispiel demonstrieren,
- die Bedeutung informationstechnischer Systeme zur Bewältigung betriebswirtschaftlicher Aufgaben erläutern.

Inhalt

Die Studierenden erlernen Grundtatbestände der Betriebswirtschaftslehre, beginnend vom Erfahrungs- und Erkenntnisobjekt dieser wissenschaftlichen Disziplin, über zu fallende konstitutive Entscheidungen, bis hin zu den diversen betriebswirtschaftlichen Funktionen innerhalb eines Betriebes.

Letztere stehen im Mittelpunkt der Veranstaltung. Die theoretischen Inhalte werden durch Praxisbeispiele untersetzt.

Durch zahlreiche Übungen wird das Verständnis für die betriebswirtschaftlichen Prozesse und deren Zusammenhänge gefestigt sowie das eigenständige Arbeiten gefördert.

Inhalte der Veranstaltung sind im Einzelnen:

- Betriebswirtschaftslehre als wissenschaftliche Disziplin
- Grundlagen der Betriebswirtschaftslehre
- Standortwahl
- Unternehmensführung
- Materialwirtschaft
- Produktionswirtschaft
- Marketing & Absatz
- Investition & Finanzierung
- Umfangreiche Übungen zu verschiedenen Vorlesungsteilen

Literatur

- BECKER, Hans Paul: Investition und Finanzierung. 2. akt. Aufl. Wiesbaden: Gabler, 2008
- BERNECKER, Michael: Grundlagen der Betriebswirtschaftslehre. 4. Aufl. Köln: Johanna, 2011.
- DÄUMLER, Klaus-Dieter; GRABE, Jürgen: Grundlagen der Investitions- und Wirtschaftlichkeitsrechnung. 12. vollst. überarbeitete Aufl. Berlin; Herne: Neue Wirtschaftsbriefe, 2007
- JUNG, Hans: Allgemeine Betriebswirtschaftslehre. akt. 12. Aufl. München: Oldenbourg, 2010
- SPECHT, Olaf; SCHMITT, Ulrich: Betriebswirtschaftslehre für Ingenieure + Informatiker. 5. Aufl. München; Wien: Oldenbourg, 2000
- THOMMEN, Jean-Paul; ACHLEITNER, Ann-Kristin: Allgemeine Betriebswirtschaftslehre. 7. vollst. überarb. Aufl. Wiesbaden: Gabler, 2012
- VAHS, Dietmar; SCHÄFER-KUNZ, Jan: Einführung in die Betriebswirtschaftslehre. 8. überarb. Aufl. Stuttgart: Schäffer-Poeschel, 2012.
- WEBER, Wolfgang; KABST, Rüdiger: Einführung in die Betriebswirtschaftslehre. 8. akt. u. überarb. Aufl. Wiesbaden: Gabler, 2012
- WÖHE, Günter; DÖRING, Ulrich: Einführung in die Allgemeine Betriebswirtschaftslehre. 25. überarbeitete und aktualisierte Aufl. München: Vahlen, 2013

4.19 Fortgeschrittene Objektorientierte Programmierung

B057 Fortgeschrittene Objektorientierte Programmierung

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B057
Modulbezeichnung	Fortgeschrittene Objektorientierte Programmierung
Lehrveranstaltung(en)	B057a Fortgeschrittene Objektorientierte Programmierung B057b Übg. Fortgeschrittene Objektorientierte Programmierung
Modulverantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist mit allen Modulen aus der objektorientierten Programmierung und dem „Software-Design“ kombinierbar, wie auch mit „Algorithmen und Datenstrukturen“.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzungen für das Verständnis sind grundlegende Kenntnisse über strukturiertes und objektorientiertes Programmieren und Grundkenntnisse der Programmiersprache Java.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B057a), Abnahme (Teil B057b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls kennen die Studierenden den methodisch fundierten praktischen Umgang mit fortgeschrittenen objektorientierten Sprachkonzepten am Beispiel von Java. Sie sind in der Lage mit generischen Klassen und Schnittstelle zu arbeiten unter anderem für den systematischen Entwurf von Containerklassen. Sie haben die große Bedeutung der Schnittstellen-Entwicklung beim Programmieren im Großen erkannt.

Sie haben erste Erfahrungen vom Arbeiten mit Entwurfsmustern an Beispielen für Kompositum, Proxy, Singleton, Fliegengewicht und anderen Mustern gesammelt. Des Weiteren verfügen die Studierenden über Kenntnisse über die systematische Software-Konstruktion unter Beachtung von Vor- und Nachbedingungen und sauberer Fehler- und Ausnahmebehandlung. Außerdem werden Kenntnisse in paralleler und nebenläufiger Programmierung am Beispiel von Java-Threads und eine Einführung in die Metaprogrammierung mit Java-Reflection vermittelt.

4.19.1 Fortgeschrittene Objektorientierte Programmierung

Lehrveranstaltung	Fortgeschrittene Objektorientierte Programmierung
Dozent(en)	Christian Uhlig
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_Inf14.0, B_MInf14.0, B_TInf14.0, B_WInf14.0) Wahl (B_STec16.0)
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, interaktive Entwicklung und Diskussion von Modellen

Lernziele

Die Studierenden ...

- nennen und erläutern die methodischen Grundlagen von objektorientierten Programmiersprachen am Beispiel von Java
- wenden fortgeschrittene Konzepte objektorientierter Programmiersprachen wie parametrische Polymorphie, Mehrfachvererbung und Funktionswerte zur Konstruktion wiederverwendbarer Softwarekomponenten an
- nennen und erläutern die Grundlagen generischer abstrakter Datentypen und ihre Korrespondenz mit Konzepten objektorientierter Sprachen (Schnittstellen, abstrakte Klassen, konkrete Klassen, Polymorphie)
- entwerfen und gestalten generische abstrakte Datentypen und ihre Implementierung
- nutzen in systematischer Weise vorgefertigte Containerstrukturen wie das Java Collections Framework
- nutzen Elemente des Design-by-Contract-Paradigmas zur Konstruktion korrekter Softwaresysteme
- nennen und erläutern Motivation, Grundlagen und Probleme nebenläufiger Programmierung
- wenden Primitiven nebenläufiger Programmierung in Java an (Erzeugen von Threads, Thread-Kommunikation / Synchronisation, usw.)
- nennen und erläutern Motivation und Konzepte zur Persistierung von Objekten
- wenden die Java Persistence API an.

Inhalt

- Generizität / Java Generics
- Abstrakte Datentypen / Container
- Java Collections
- Funktionswerte in Java (Funktionale Interfaces, Lambda-Ausdrücke, Methodenreferenzen)

- Multithreading in Java (Threads, Monitore, Waitsets, volatile Variablen, Java Streams API)
- Laufzeit-Typinformationen (Java Reflection)
- Persistierung von Objekten am Beispiel der Java Persistence API

Literatur

- Christian Uhlig: Fortgeschrittene Objektorientierte Programmierung, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~uhl/foop.html>
- Meyer, Bertrand: Objektorientierte Softwareentwicklung, Hanser, München, 1990, ISBN: 3-446-15773-5
- Gosling, James; Joy, Bill; Steele, Guy: The Java Language Specification, 2nd Edition, Addison-Wesley, Reading, 2000, ISBN: 0-201-31008-2
- Campione, Mary; Walrath, Kathy: The Java Tutorial, Third Edition, Object-Oriented Programming for the Internet, Addison-Wesley, Reading, 2000, ISBN: 0-201-31007-4
- Christian Ullenboom: Java ist auch eine Insel, 11. Auflage, Galileo Press GmbH, 2014, ISBN: 978-3-8362-2873-2

4.19.2 Übg. Fortgeschrittene Objektorientierte Programmierung

Lehrveranstaltung	Übg. Fortgeschrittene Objektorientierte Programmierung
Dozent(en)	Malte Heins
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_Inf14.0, B_MInf14.0, B_TInf14.0, B_WInf14.0) Wahl (B_STec16.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- erlangen ein Verständnis über die Abläufe beim Arbeiten mit parallelen und nebenläufigen Berechnungen mit Threads.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte der fortgeschrittenen OOP behandelt, die nicht Bestandteil der Vorlesung sind.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung Fortgeschrittene Objektorientierte Programmierung

4.20 Datenbanken 2

B117 Datenbanken 2

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B117
Modulbezeichnung	Datenbanken 2
Lehrveranstaltung(en)	B117a Datenbanktheorie und -implementierung
Modulverantwortliche(r)	Prof. Dr. Ulrich Hoffmann
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll mit dem grundlegenden Module „Datenbanken 2“ kombinierbar. In einem konsekutiven Studiengang steht es im Zentrum eines Datenbank-Curriculums und lässt sich gut durch das Modul „Datenbanken 3“ (Master) ergänzen. Das Modul sollte in allen Studiengängen verwendet werden, in denen Datenhaltung ein Studienschwerpunkt ist.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Es werden grundlegende Kenntnisse in der Benutzung von Datenbanksystemen und Kenntnisse in SQL vorausgesetzt.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Besuch der Veranstaltungen dieses Moduls besitzen die Studierenden eingehende Kenntnisse der für die Implementierung von Datenbanksystemen wichtigen Architekturprinzipien, ihrer Datenstrukturen und Algorithmen. Sie besitzen die Fähigkeit Datenbanksysteme in Hinblick auf ihren Aufbau und der internen Funktionsweise zu bewerten und zu beurteilen. Sie sind in der Lage selbst Modifikationen an Datenbanksystem-Implementierungen zu planen und durchzuführen. Die Studierenden besitzen die Fähigkeit, die Arbeitsweise von Datenbanksystemen zu optimieren und auch selbst Architekturen für die Datenhaltung zu entwerfen. Sie besitzen grundlegende Kenntnisse der Administration von Datenbanken.

4.20.1 Datenbanktheorie und -implementierung

Lehrveranstaltung	Datenbanktheorie und -implementierung
Dozent(en)	Ulrich Hoffmann
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_Inf14.0) Wahl (B_EComI14.0, B_WInf14.0)
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assig.
ECTS	5.0
Lehr- und Medienform(en)	Handout

Lernziele

Die Studierenden besitzen ...

- Kenntnisse der relationalen Datenbanktheorie und ihrer wesentlichen Fragestellungen;
- detaillierte Kenntnisse der für die Implementierung von Datenbanksystemen wichtigen Architekturprinzipien, Datenstrukturen und Algorithmen und damit des Aufbau und der interne Arbeit eines großen komplexen Softwaresystems;
- die Fähigkeit, die Arbeitsweise von Datenbanksystemen zu optimieren bzw. selbst Architekturen für große komplexe Softwaresysteme zu entwerfen;
- Fähigkeit eines Datenbankadministrators für Datenbanksysteme.

Inhalt

- Relationale Datenbanksysteme
 - Das 3 - Ebenen - Architekturkonzept
 - Transaktionskonzept
 - Relationale Operationen und Relationenalgebra
 - Komponenten eines DBMSs und Betrieb eines DBSs
- Aufbau eines Datenbankverwaltungssystems (DBMS)
 - Systementwurf und Schichtenmodell
 - Architektur eines DBMS
 - Komponenten eines DBMS
- Speichersystem
 - Speicherverwaltung
 - Systempufferverwaltung
- Speicherungsstrukturen und Zugriffspfade
 - Freispeicherverwaltung
 - Abbildung von Datensätzen in Seiten
 - Zugriffspfade
- Satzorientierte Datenbankschnittstelle
- Mengenorientierte Datenbankschnittstelle

- Datenwörterbuch
- Sicherung der Integrität in Datenbanken
 - Semantische Integrität
 - Operationale Integrität
 - Fehlerbehandlung - Recovery

Literatur

- Elmasri, Ramez; Navathe, Shamkant B.:
Grundlagen von Datenbanksystemen.
3. Auflage. München: Pearson -Verlag, 2009.
- Heuer, Andreas:
Datenbanken kompakt.
Bonn: mitp-Verlag, 2003.
- Meier, Andreas:
Relationale Datenbanken Leitfaden für die Praxis.
Berlin: Springer-Verlag, 2004.
- Vetter, Max:
Aufbau betrieblicher Informationssysteme mittels konzeptioneller Datenmodellierung.
8. Auflage. Stuttgart: Vieweg-Teubner, 1998.
- Vossen, Gottfried:
Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme.
5. Auflage. Oldenbourg: Oldenbourg-Wissenschaftsverlag, 2008
- Saake, Gunter; Sattler, Kai-Uwe; Heuer, Andreas:
Datenbanken Implementierungstechniken.
3. Auflage. mitp-Verlag, 2011

4.21 Software-Design

B058 Software-Design

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B058
Modulbezeichnung	Software-Design
Lehrveranstaltung(en)	B058a Software-Design
Modulverantwortliche(r)	Prof. Dr. Uwe Schmidt
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Dieses Modul bietet gute Grundlagen für größere Projekte, zum Beispiel für das Modul „Software-Projekt“.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzungen für dieses Modul sind Kenntnisse in imperativen Programmiersprachen insbesondere Java, und über Datentypen und Typkonstruktoren in höheren Programmiersprachen. Kenntnisse über die Funktionale Programmierung mit Haskell sind nicht zwingend notwendig aber nützlich.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls kennen die Studierenden die Bedeutung und den praktischen Umgang mit der Modellbildung im Software-Design. Sie sind in der Lage, an Hand der Modellierung überschaubare, aber nicht triviale Fallstudien sowohl mit informellen Methoden, wie UML, als auch mit formalen Spezifikationsmethoden und mit abstrakter Syntax in Haskell-Syntax, Konsistenz, Vollständigkeit, Erweiterbarkeit und Qualität von Modellen zu diskutieren und zu bewerten.

Sie können die Qualität der Modelle durch lauffähige Prototypen überprüfen und demonstrieren. Sie können hierfür die Sprache Haskell als ausführbare Spezifikationssprache einsetzen. Die Studierenden besitzen sichere Kenntnisse über die gängigen Entwurfsmuster und deren Vor- und Nachteile. Sie sind in der Lage aus den entwickelten Modellen auf systematische Weise Klassen-Strukturen in Java abzuleiten.

4.21.1 Software-Design

Lehrveranstaltung	Software-Design
Dozent(en)	Uwe Schmidt
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_EComI14.0, B_Inf14.0, B_MInf14.0, B_WInf14.0) Wahl (B_IMCA16.0, B_TInf14.0)
Lehrform / SWS	Vorlesung
ECTS	5.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration, interaktive Entwicklung und Diskussion von Modellen

Lernziele

Die Studierenden ...

- erkennen die Bedeutung und den praktischen Umgang mit der Modellbildung im Softwaredesign.
- lernen die zentralen Entwurfsmuster und ihre Anwendungsfelder kennen.
- modellieren überschaubare, aber nicht triviale, Fallstudien sowohl mit informellen Methoden, wie UML, als auch mit formalen Spezifikationsmethoden und mit abstrakter Syntax.
- entwickeln lauffähige Prototypen mit der als ausführbarer Spezifikationsprache eingesetzten funktionalen Sprache Haskell.

Inhalt

- Methoden, Techniken und Werkzeuge im Software-Entwurf
 - OMT, UML
 - formale Methoden
 - Abstrakte Syntax zur Datenmodellierung
- Entwurfsmuster
 - Strukturmuster
 - Verhaltensmuster
 - Erzeugungsmuster
- Fallstudien (Beispiele)
 - Modellierung von Adressbeständen
 - XML Strukturbaum
 - Indexierer für Freitextsuche
 - Projektstagebuch
 - Medienkatalog

Literatur

- Uwe Schmidt: Softwaredesign, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/~si/vorlesungen/softwaredesign/design.html>

- Gamma, Erich e., a.: Entwurfsmuster: Elemente wiederverwendbarer objektorientierter Software, Addison-Wesley, Bonn, 2001 (korrigierter Nachdruck)
- Fowler, Martin; Scott, Kendall: UML Distilled. Applying The Standard Object Modelling Language, Addison-Wesley Longman, Amsterdam, 1997
- Hutton, Graham: Programming in Haskell, Cambridge University Press, 2007, ISBN 0-521-69269-5

4.22 Web-Anwendungen

B059 Web-Anwendungen

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B059
Modulbezeichnung	Web-Anwendungen
Lehrveranstaltung(en)	B059a Web-Anwendungen B059b Übg. Web-Anwendungen
Modulverantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul baut auf den Kompetenzen auf, die durch Module des Themenbereichs Programmierung in Informatik-Studiengängen, insbesondere „Programmstrukturen 1“, „Programmstrukturen 2“ und „Algorithmen und Datenstrukturen“, vermittelt werden. Es schafft die Voraussetzungen für Module im fortgeschrittenen Studienverlauf, in denen Kenntnisse zur Realisierung von Web-Anwendungen benötigt werden. Dies kann beispielsweise in den Modulen „Software-Projekt“, „Konzepte des E-Commerce“ und der Bachelor-Thesis der Fall sein.
SWS des Moduls	5
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 47 Stunden Eigenstudium: 103 Stunden
Voraussetzungen	Grundkenntnisse der imperativen Programmierung, Kenntnis der Kernbestandteile der Programmiersprache Java.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B059a), Abnahme (Teil B059b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Die Studierenden kennen die speziellen technischen Randbedingungen und Besonderheiten der Entwicklung von Web-Anwendungen im Vergleich zu lokal laufenden Applikationen. Sie kennen die wichtigen Konzepte zur Realisierung von Web-Anwendungen und die Sprachen, die bei der Erstellung und im Umfeld des Einsatzes von Web-Anwendungen zum Einsatz kommen.

Die Studierenden kennen die Möglichkeiten der Cascading Style Sheets, wesentliche Bestandteile der Programmiersprachen Javascript und PHP sowie Java-basierte Konzepte

zur Realisierung von Web-Anwendungen. Sie kennen die Möglichkeiten des Einsatzes von Frameworks zur Unterstützung der Entwicklung und können den Nutzen solcher Frameworks einschätzen. Sie kennen das Sprachkonzept von XML und können dieses für einfache Anwendungssituationen nutzen.

Sie können auf Basis dieser Kenntnisse eigenständig Web-Seiten realisieren, die einfache Formen der Dynamik sowohl client-seitig als auch server-seitig enthalten.

Sie kennen die wesentlichen Erweiterungen von HTML 5 im Vergleich zu früheren HTML-Versionen und können diese auszugsweise zur Realisierung von Webseiten einsetzen.

4.22.1 Web-Anwendungen

Lehrveranstaltung	Web-Anwendungen
Dozent(en)	Andreas Häuslein
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_CGT14.0, B_ECom14.0, B_Inf14.0, B_MInf14.0, B_WInf14.0) Wahl (B_IMCA16.0)
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration, Gastreferenten

Lernziele

Die Studierenden ...

- führen die technischen Randbedingungen des Internet auf und benennen ihre Auswirkungen auf die Entwicklung von Software.
- beschreiben die konzeptionellen Aspekte von Stylesheets und der zentralen Möglichkeiten zur Festlegung der Darstellung in den Cascading Stylesheets und nutzen diese zur Erzeugung angestrebter Darstellungsweisen.
- zählen wichtige Konzepte, Sprachen, Frameworks und Architekturen zur Realisierung dynamischer Webseiten auf, wählen zwischen diesen problembezogen aus und nutzen sie zur Erstellung einfacher dynamischer Webseiten.
- erläutern das Sprachkonzept von XML und der damit verbundenen anwendungsneutralen Techniken.
- bewerten die Einsatzmöglichkeiten der XML-Techniken und definieren neue anwendungsspezifische XML-Sprachen
- benennen die Möglichkeiten der XML-basierten Transformation von XML-Dokumenten.
- beschreiben die Basiskonzepte der auf XML-basierenden Techniken zur Realisierung dynamischer Web-Seiten (AJAX, Flex) und beurteilen diese im Vergleich zu anderen Techniken.
- geben die zusätzliche Konzepte und Sprachelemente von HTML 5 an und entwerfen damit Webseiten.
- nutzen die theoretisch vermittelten Inhalte zur eigenständigen Realisierung von Webanwendungen begrenzter Komplexität.

Inhalt

- Basiskonzepte des WWW
 - Klassische Auszeichnungsmöglichkeiten in HTML
 - HTML-Formulare und ihre Möglichkeiten
 - Style Sheets
- Dynamik in Web-Seiten
 - Client-seitige Dynamik

- * Programmiersprache Javascript
- * Javascript-Frameworks
- Server-seitige Dynamik
 - * Programmiersprache PHP
 - * Java-basierte Realisierungsansätze
- XML und damit verbundene Sprachkonzepte
 - Grundstruktur von XML-Dokumenten
 - XML-Auszeichnungsregeln
 - Definition von XML-Sprachen mit Document Type Definitions
 - XML Schema zur Definition von XML-Sprachen
 - Verarbeitung und Transformation von XML-Dokumenten
- AJAX - Asynchronous Javascript and XML
- Flex
- Ausgewählte Neuerungen und Sprachbestandteile von HTML 5

Literatur

- LABORENZ, Kai: CSS: Das umfassende Handbuch. Galileo Computing, 2011
- BALZERT, Heide: Basiswissen Web-Programmierung. 2. Aufl., W3L, 2011
- LUBKOWITZ, Mark: Webseiten programmieren und gestalten: Bonn: Galileo Press, 2007
- POMASKA, Günther: Webseiten-Programmierung: Sprachen, Werkzeuge, Entwicklung, Springer Vieweg, 2012
- WISSMANN, Dieter: JavaServer Pages: Dynamische Websites mit JSP erstellen, W3L, 2012
- SKONNARD, Aaron; GUDGIN, Martin: Essential XML Quick Reference. Boston: Pearson Education, 2003
- REINHARDT, Gerald: Praxiswissen Flex 3. Köln: O'Reilly, 2009
- SCHÜRMAN, Tim: Moderne Web-Programmierung. O'Reilly, 2011
- GASSTON, Peter: Moderne Webentwicklung: Geräteunabhängige Entwicklung - Techniken und Trends in HTML5, CSS3 und JavaScript, dpunkt.verlag, 2014
- THE PHP GROUP: PHP Documentation. <http://www.php.net/docs.php>. Aktualisierungsdatum: 19.6.2014
- WORLD WIDE WEB CONSORTIUM: HTML 5. <http://www.w3.org/TR/2014/WD-html5-20140617/>

4.22.2 Übg. Web-Anwendungen

Lehrveranstaltung	Übg. Web-Anwendungen
Dozent(en)	Nadim Kolodziej
Hörtermin	4
Art der Lehrveranstaltung	Pflicht (B_CGT14.0, B_EComI14.0, B_Inf14.0, B_MInf14.0, B_WInf14.0) Wahl (B_IMCA16.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- sind in der Lage, die in der Vorlesung vermittelten theoretischen Hintergründe selbst praktisch anzuwenden.
- haben umfangreiche Kenntnisse und praktische Erfahrungen zu den Themen HTML, CSS, serverseitiger Dynamik und Datenbankanbindung mit PHP, clientseitiger Dynamik mit JavaScript und AJAX, Einsatz von XML und JSON zum Austausch von Daten zwischen Client und Server, Einsatz von Cookies und Sessions zum temporären Speichern von Daten.
- steigern ihre Teamfähigkeit durch intensive Arbeit in Zweiertteams und Kommunikation über auftretende Probleme in der ganzen Gruppe.

Inhalt

Bearbeitung von Übungsaufgaben, die sich am Stoff der Vorlesung orientieren, in Zweiergruppen mit Abnahme der Lösungen. Erstellt wird eine im Verlaufe der einzelnen Übungseinheiten komplexer werdende Web-Anwendung, wobei die einzelnen Schritte aufeinander aufbauen, so dass am Ende eine komplexe Web-Anwendung entsteht, die einen Großteil der in der Vorlesung erlernten Techniken und Konzepte nutzt.

Literatur

- Im Rahmen der Übungsveranstaltungen werden Beispielanwendungen zur Verfügung gestellt:
<http://www.fh-wedel.de/mitarbeiter/mle/uebung-softwaretechniken-fuer-internetanwendungen/>
- Vorlesungsmaterial von Prof. Dr. Häuslein:
<https://stud.fh-wedel.de/handout/Haeuslein/>

4.23 UNIX & Shell-Programmierung

B044 UNIX & Shell-Programmierung

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B044
Modulbezeichnung	UNIX & Shell-Programmierung
Lehrveranstaltung(en)	B044a UNIX & Shell Programmierung B044a Übg. UNIX & Shell-Programmierung
Modulverantwortliche(r)	PhD Martin Dietze
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul ist sinnvoll kombinierbar mit den Themengebieten Betriebssysteme, Compilerbau, Skriptsprachen und Webentwicklung.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzung sind grundlegende Kenntnissen im Umgang mit strukturierten und objektorientierten Programmiersprachen.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Abnahme
Anteil an Gesamtnote	0
Sprache	deutsch

Lernziele des Moduls

Mittels der Veranstaltung über UNIX können die Studierenden mit Skriptsprachen arbeiten. Sie kennen die Vor- und Nachteile von Skriptsprachen bei der Software-Entwicklung. Ferner kennen sie die regulären Ausdrücke zur Verarbeitung von Texten und die Mächtigkeit und die Grenzen von regulären Ausdrücken. Sie verstehen die einfache und elegante Art der Kombinierbarkeit von Programmen, insbesondere an Hand von Filtern und Pipes.

Die Studierenden können die Vor- und Nachteile von Skriptsprachen einschätzen. Sie kennen die Flexibilität aber auch die Fehleranfälligkeit von dynamischen Sprachen und haben eine Vorstellung davon, wann und sich wie die Produktivität beim Arbeiten mit Skriptsprachen im Vergleich zu kompilierten Sprachen verändert.

4.23.1 UNIX & Shell Programmierung

Lehrveranstaltung	UNIX & Shell Programmierung
Dozent(en)	Martin Dietze
Hörtermin	4
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	2.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- erlangen das Einschätzungsvermögen zu den Vorteilen und Gefahren von Skriptsprachen in der Software-Entwicklung am Beispiel der bash.
- erlangen die Fähigkeit zum praktischen Arbeiten mit regulären Ausdrücken.
- erlangen Kenntnisse über die Mächtigkeit und der Grenzen von regulären Ausdrücken.
- beherrschen reguläre Ausdrücke für die Verarbeitung von Texten und Auszeichnungssprachen.
- erlangen die Fähigkeit zum Arbeiten mit Filtern und Pipes.
- erwerben das Grundverständnis über interne Abläufe im UNIX-Kern bei der Prozessverwaltung.

Inhalt

- Unix
 - Systemstruktur
 - einfache Shell Kommandos
 - Dateisystem
 - Filter und Pipelines
 - Skriptprogrammierung mit der Shell
- Reguläre Ausdrücke
 - Reguläre Mengen
 - Mächtigkeit und Grenzen von regulären Ausdrücken
 - Suchen, Zerlegen und Editieren mit regulären Ausdrücken
- Skriptsprachen
 - Einfache bash-Programme
 - Systematisches Kombinieren kleiner Programme
- Das make-System
- Prozessverwaltung

Literatur

- Uwe Schmidt: Unix und Internet, Vorlesungsunterlagen im Web:

<http://www.fh-wedel.de/~si/vorlesungen/internet/internet.html>

- Kofler, Michael: Linux, Das umfassende Handbuch, 2013, Galileo Press, ISBN: 978-3-8362-2591-5

4.23.2 Übg. UNIX & Shell-Programmierung

Lehrveranstaltung	Übg. UNIX & Shell-Programmierung
Dozent(en)	Malte Heins
Hörtermin	4
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	3.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- wenden praktisch die Inhalte aus der Vorlesung an.
- beherrschen den praktischen Umgang mit der bash und den gängigsten UNIX Kommandos.
- erlangen die Fähigkeit zur Erstellung kleiner nichttrivialer bash-Programme durch das Arbeiten mit Pipes und Filtern.
- erlangen das Verständnis über die Abläufe in einer Maschine bei der Prozessverwaltung und dem Starten und Synchronisieren von Prozessen.

Inhalt

Bearbeitung von Übungsaufgaben parallel zum Stoff der Vorlesung in Zweiergruppen mit Abnahme und Diskussion der Lösungen. Zusätzlich werden praxisrelevante Aspekte des Betriebssystems UNIX und der bash behandelt, die nicht Bestandteil der Vorlesung sind.

Literatur

- Unterlagen zur Übung im Web
- siehe auch Vorlesung UNIX und Shell-Programmierung

4.24 Implementierung von Geschäftsprozessen in ERP-Systemen

B080 Implementierung von Geschäftsprozessen in ERP-Systemen

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B080
Modulbezeichnung	Implementierung von Geschäftsprozessen in ERP-Systemen
Lehrveranstaltung(en)	B080a Implementierung von Geschäftsprozessen in ERP-Systemen B080b Übg. Implementierung von Geschäftsprozessen in ERP-Systemen
Modulverantwortliche(r)	Dipl.-Inform. (FH) Birger Wolter
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Implementierung von Geschäftsprozessen in ERP-Systemen“ baut auf die unter anderem im Modul „Einführung in die Betriebswirtschaft“ erworbenen Kenntnisse und Fähigkeiten auf und vertieft diese in praktischer Hinsicht. Die betriebswirtschaftliche Sichtweise wird im Modul „Anwendungsentwicklung in ERP-Systemen“ um die programmiertechnische Sichtweise ergänzt.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Grundkenntnisse betriebswirtschaftlicher Zusammenhänge in Unternehmen.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B080a), Übung (Teil B080b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls verfügen die Studierenden über Wissen hinsichtlich der Realisierung der betriebswirtschaftlichen Grundfunktionalitäten innerhalb eines ERP-Systems. Sie verfügen über die wichtige Kompetenz, wirtschaftliche und informationstechnische Vorgänge integrativ zu betrachten, die sie sich im Rahmen einer geschäftsprozessorientierten Unternehmensorganisation selbstständig erarbeitet haben. Praktisch beherrschen die Studierenden das ERP-System der SAP AG.

4.24.1 Implementierung von Geschäftsprozessen in ERP-Systemen

Lehrveranstaltung	Implementierung von Geschäftsprozessen in ERP-Systemen
Dozent(en)	Birger Wolter
Hörtermin	4
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assig. m.
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Softwaredemonstration, interaktive Entwicklung und Diskussion von Modellen

Lernziele

Die Studierenden können ...

- wesentliche Begriffe im Umfeld von ERP-Systemen und Geschäftsprozessen definieren und diese in Beziehung zueinander setzen
- die wesentlichen Eigenschaften von ERP-Systemen und deren Architekturen herausstellen
- können die Vorgehensweise zur Abbildung betriebswirtschaftlicher Prozesse in ERP-Systemen erläutern.

Inhalt

- Struktur der IDES-Modellfirma
- Prozesse der Produktionsplanung und deren Abbildung im ERP-System
- Prozesse des Controllings und deren Abbildung im ERP-System
- Prozesse der Logistik und deren Abbildung im ERP-System
- Übungsaufgaben Controlling und Logistik

Literatur

- FRICK, Detlev; GADATSCH, Andreas; SCHÄFFER-KÜLZ, Ute G.: Grundkurs SAP ERP. Geschäftsprozessorientierte Einführung mit durchgehendem Fallbeispiel. Springer Fachmedien Wiesbaden GmbH, 1. Auflage 2008 (Neuaufgabe 2015)
- WOLTER, Birger: Veranstaltungsmaterialien auf dem Handout-Server. <http://www.fh-wedel.de/mitarbeiter/wol/veranstaltungen/erp-software/> (kontinuierliche Aktualisierung)

4.24.2 Übg. Implementierung von Geschäftsprozessen in ERP-Systemen

Lehrveranstaltung	Übg. Implementierung von Geschäftsprozessen in ERP-Systemen
Dozent(en)	Birger Wolter
Hörtermin	4
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden können ...

- die erworbenen theoretischen Kenntnisse auf praktische Anwendungsfälle übertragen
- die grundlegende Funktionen von ERP-Systemen verwenden
- betriebswirtschaftliche Aufgabenstellungen und zusammenhängende Geschäftsvorfälle in verschiedenen Problemfeldern mit Hilfe von ERP-Systemen bearbeiten.

Inhalt

- Navigation im SAP ERP-System
- Produktionsplanung-Fallstudie
- Controlling-Fallstudie
- Logistik-Fallstudie
- Zusammenhängendes Fallbeispiel zu den behandelten Themen

Literatur

- WOLTER, Birger: Übungsmaterialien auf dem Handout-Server. <http://www.fh-wedel.de/mitarbeiter/vsoftware/> (kontinuierliche Aktualisierung)

4.25 Systemsoftware

B096 Systemsoftware

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B096
Modulbezeichnung	Systemsoftware
Lehrveranstaltung(en)	B096a Konzepte der Betriebssysteme B096a Compilerbau
Modulverantwortliche(r)	PD Dr. Dennis Säring
Zuordnung zum Curriculum	Informatik (Bachelor) Medieninformatik (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Die Inhalte dieses Moduls können sinnvoll in Projekten, zum Beispiel im Modul „Software-Projekt“ oder in der Bachelor-Thesis, genutzt werden.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Voraussetzung für eine erfolgreiche Absolvierung dieses Moduls sind grundlegende Kenntnisse über die Nutzung aktueller Betriebssysteme sowie ein Grundverständnis der Funktionsweise der Komponenten eines informationsverarbeitenden Systems und der darin zur Verfügung stehenden verschiedenen Sprachebenen.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Grundlegende Kompetenz zum Erwerb eines tieferen Verständnisses für Konzepte der Systemsoftware, sowohl im Bereich der Implementierungsstrategien moderner Multitasking-Betriebssysteme als auch bezüglich der Thematik des Übersetzerbaus und der Formalen Sprachen.

Dabei geht es um die Erfassung der Detailfunktionalität wesentlicher Systemfunktionen an ausgewählten Beispielen und die Erkennung der Optimierungsmöglichkeiten für die Arbeitsabläufe bei modernen Multitasking-Betriebssystemen. Ziel ist die Fähigkeit zur angemessenen Einschätzung des Systemverhaltens im Rahmen der Softwareentwicklung und -anwendung sowie zur Erkennung und Bewertung der Eigenschaften und Unterschiede realer Betriebssysteme.

Zusätzlich soll eine Durchdringung insbesondere der ersten Phasen bei der Übersetzerkonstruktion erreicht werden, nämlich der lexikalischen und der Syntaxanalyse. Dabei geht

es auch um die Fähigkeit zur Nutzung dieser Kenntnisse im Web-Bereich mit den vielen unterschiedlichen Auszeichnungssprachen wie HTML, XML, XSL, XSLT, WML.

Die Studierenden erfahren die Vorteile des Einsatzes von Domänen-spezifischen Sprachen (DSLs) und deren Realisierung mit Techniken aus dem Compilerbau zur Implementierung komplexer Systeme. Insgesamt dient die Beschäftigung mit der Thematik des Compilerbaus der Fähigkeit zum gewinnbringenden Einsatz dieser Techniken bei der Erstellung von effizienten und zuverlässigen Programmen.

4.25.1 Konzepte der Betriebssysteme

Lehrveranstaltung	Konzepte der Betriebssysteme
Dozent(en)	Dennis Säring
Hörtermin	5
Art der Lehrveranstaltung	Pflicht (B_Inf14.0, B_TInf14.0) Wahl (B_MInf14.0, B_WInf14.0)
Lehrform / SWS	Vorlesung
ECTS	2.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, E-Learning

Lernziele

Die Studierenden ...

- besitzen eine grundlegende Kompetenz zum Verständnis technischer und algorithmischer Konzepte von aktuellen Betriebssystemen
- sind vertraut mit Prozessen und Prozessverwaltung aktueller Betriebssysteme
- kennen die wesentlichen Ansätze zur Speicherverwaltung
- haben Kenntnisse über die Strukturen aktueller Dateisysteme
- kennen die Ansätze anderer Betriebssysteme (z.B. mobile Betriebssysteme)

Inhalt

- Grundlagen eines Betriebssystems
- Prozesse und Prozessverwaltung
- Threads und Semaphore
- Speicherverwaltung
- Prozesskommunikation
- Festplatten I/O
- Dateisysteme
- Mobile Betriebssysteme

Literatur

- Tanenbaum: Moderne Betriebssysteme, Verlag Pearson Studium, 2002

4.25.2 Compilerbau

Lehrveranstaltung	Compilerbau
Dozent(en)	Uwe Schmidt
Hörtermin	5
Art der Lehrveranstaltung	Pflicht (B_Inf14.0, B_TInf14.0) Wahl (B_MInf14.0, B_WInf14.0)
Lehrform / SWS	Vorlesung
ECTS	2.5

Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Software demonstration
---------------------------------	--

Lernziele

Die Studierenden erlangen ...

- tiefgreifendes Verständnis, insbesondere der ersten Phasen bei der Übersetzerkonstruktion, nämlich der lexikalischen und der Syntaxanalyse.
- die Fähigkeit zur Nutzung dieser Kenntnisse im Web-Bereich mit den vielen unterschiedlichen Auszeichnungssprachen wie HTML, XML, XSL, XSLT, WML.
- die Fähigkeit zum gewinnbringenden Einsatz dieser Techniken bei der Erstellung von effizienten und zuverlässigen Programmen.
- das Grundverständnis über die Transformation höherer Programmiersprachen in Maschinensprache.
- Kenntnisse über die Arbeitsweise von Compilern und Interpretierern.
- das Grundverständnis zu virtuellen Maschinen.

Inhalt

- Compiler im Überblick
 - Compilerphasen Portierung und Bootstrapping
 - Compiler und Interpretierer
- Grundlagen der formalen Sprachen und der Automatentheorie
- Lexikalische Analyse
 - Reguläre Ausdrücke
 - Nichtdeterministische und deterministische endliche Automaten
 - Scanner und Scanner-Generatoren
- Syntaxanalyse
 - Rekursiver Abstieg
 - LL- und LR- Parser
 - Parser-Generatoren
- Semantische Analyse
 - Typüberprüfung
- Codeerzeugung
- Virtuelle Maschinen

Literatur

- Uwe Schmidt: Compilerbau, Vorlesungsunterlagen im Web: <http://www.fh-wedel.de/si/vorlesungen/c>
- Aho, Alfred V.; Sethi, Ravi; Ullman, Jeffrey D.: Compilers, Principles, Techniques and Tools, Addison-Wesley Longman, Amsterdam, 2000 (Repr.)
- Appel, Andrew W.; Palsberg, Jens: Modern Compiler Implementation in Java, 2.nd edition, Cambridge University Press, 2002, ISBN: 0-521-82060-X

- Wirth, Niklaus: Grundlagen und Techniken des Compilerbaus, Oldenbourg, 1995, ISBN: 3-486-24374-8

4.26 Anwendungen der Künstlichen Intelligenz

B095 Anwendungen der Künstlichen Intelligenz

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B095
Modulbezeichnung	Anwendungen der Künstlichen Intelligenz
Lehrveranstaltung(en)	B095a Anwendungen der Künstlichen Intelligenz
Modulverantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul liefert praktische Anwendungen aus verschiedenen Bereichen der teilnehmenden Studiengänge. Es gibt Ideen für das Praktikum und die anschließende Bachelor-Thesis. Es liefert Grundlagen, die zur Aufnahme eines Masterstudiums motivieren.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Erwartet werden Kenntnisse der Diskreten Mathematik sowie gute Programmierkenntnisse. Die Teilnehmer sollten bereits größere Programme geschrieben haben und Problemstellungen aus der Praxis kennengelernt haben (mindestens im Rahmen angewandter Vorlesungen). Vertrautheit mit objekt-orientierter Programmierung ist von Vorteil.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur + ggf. Bonus
Anteil an Gesamtnote	2,7
Sprache	deutsch/englisch

Lernziele des Moduls

Nach Abschluss des Moduls kennen die Studierenden die grundsätzlichen Ziele der Künstlichen Intelligenz und deren Anwendungsbezug. Die Studierenden kennen komplexe Anwendungsbeispiele und sind in der Lage, elementare Techniken der Künstlichen Intelligenz in Implementierungen anzuwenden. Hierfür verfügen sie über eine grundlegende Kenntnis wichtiger Basistechnologien der Künstlichen Intelligenz.

4.26.1 Anwendungen der Künstlichen Intelligenz

Lehrveranstaltung	Anwendungen der Künstlichen Intelligenz
Dozent(en)	Sebastian Iwanowski
Hörtermin	5
Art der Lehrveranstaltung	Pflicht (B_CGT14.0, B_Inf14.0, B_ITE15.0) Wahl (B_EComI14.0, B_MInf14.0, B_STec16.0, B_TInf14.0, B_WInf14.0)
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	5.0
Lehr- und Medienform(en)	-

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Kenntnis und Interesse für die grundsätzlichen Ziele der Künstlichen Intelligenz.
- Kenntnis der Basistechnologien der Künstlichen Intelligenz.
- Fähigkeit, elementare Techniken der Künstlichen Intelligenz in Implementierungen anzuwenden.
- Kenntnis verschiedener komplexer Anwendungsbeispiele.

Inhalt

- Einführung
 - Definition und Ziele der KI
 - Überblick über die Basistechnologien der KI
 - Auswahl von Anwendungsbeispielen
- Basistechnologien
 - Expertensysteme und Wissensbasierte Systeme
 - Suchstrategien
 - Schwarmintelligenz
- Anwendungen
 - Verkehrsinformation und -navigation
 - Logistische Fragestellungen
 - Technische Diagnose
 - Spiele

Literatur

- Marco Dorigo / Thomas Stützle:
Ant Colony Optimization,
MIT Press 2004, ISBN 0-262-04219-3
- Ute Schmid / Günter Görz / Josef Schneeberger:
Handbuch der Künstlichen Intelligenz,
Oldenbourg 2013 (5. Auflage), ISBN 978-3-486-71307-7
- Stuart Russell / Peter Norvig:

Künstliche Intelligenz: Ein moderner Ansatz,
Pearson Studium 2004 (2. Auflage), ISBN 3-8273-7089-2

4.27 Software-Qualität

B093 Software-Qualität

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B093
Modulbezeichnung	Software-Qualität
Lehrveranstaltung(en)	B093a Softwarequalität
Modulverantwortliche(r)	Prof. Dr. Gerd Beuster
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul setzt grundlegende Kenntnisse der Softwareerstellung, insbesondere der Programmierung in einer höheren Programmiersprache, voraus. Die Studierenden sind mit den Kenntnissen des Moduls in der Lage, allgemeine Methoden der Qualitätssicherung auf Software anzuwenden. Sie sind auch mit den speziellen Methoden der Qualitätssicherung von Software, insbesondere Methoden zur systematischen Erstellung von Softwaretests, vertraut.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Kenntnis grundlegender Eigenschaften von Software-Systemen und ihrer Realisierung, Fähigkeit zur Analyse von Software, um ein Verständnis der darin enthaltenen Zusammenhänge zu erlangen, Kenntnisse hinsichtlich der Vernetzung von Rechnern und der Software-Konzepte zur Nutzung der Vernetzung.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur + ggf. Bonus
Anteil an Gesamtnote	2,7
Sprache	deutsch/englisch

Lernziele des Moduls

Nach Abschluss des Moduls verfügen die Studierenden über Kenntnisse hinsichtlich verschiedener Qualitätsaspekte von Software-Systemen. Die Studierenden kennen Qualitätsmerkmale und -kriterien und die Methoden zur Erreichung entsprechender Qualitätsziele.

Die Studierenden wissen, welche Bedeutung weitere Qualitätsmerkmale von Software haben. Darüber hinaus kennen sie die Gründe für das Zustandekommen von Qualitätsdefizite und die Maßnahmen zur Gewährleistung eines geforderten Qualitätsniveaus.

Die Studierenden haben systematisches Testen als Mittel zur Qualitätssicherung und -kontrolle

kennengelernt. Sie können die gängigen Methoden und Verfahren zum White-Box-Testing (Testen unter Kenntnis der Spezifikation und/oder Implementierung) und Black-Box-Testing (Testen ohne Kenntnis der internen Funktionsweise des IT-Systems) theoretisch begründen und praktisch umsetzen.

4.27.1 Softwarequalität

Lehrveranstaltung	Softwarequalität
Dozent(en)	Jochen Brunnstein
Hörtermin	5
Art der Lehrveranstaltung	Pflicht (B_Inf14.0, B_MInf14.0, B_WInf14.0) Wahl (B_EComI14.0, B_IMCA16.0)
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigmn.
ECTS	5.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden kennen Qualitätsmerkmale und -kriterien und die Methoden zur Erreichung entsprechender Qualitätsziele. Sie können die gängigen Methoden und Verfahren zur Sicherstellung von Softwarequalität umsetzen.

- Kenntnis der wesentlichen Qualitätsmerkmale von Software und ihrer wechselseitigen Abhängigkeiten.
- Kenntnis der typischen Defizite der Software-Qualität und ihrer Gründe.
- Kenntnis der Aufgabenbereiche des Software-Qualitätsmanagement und Überblick über die wesentlichen Managementkonzepte.
- Kenntnis des Konzepts der Qualitätsmodelle und der relevanten Qualitätsmerkmale und -metriken.
- Überblick über mögliche Maßnahmen der Software-Qualitätssicherung, Kenntnis der wesentlichen konstruktiven und analytischen Maßnahmen der Software-Qualitätssicherung.
- Fähigkeit, ausgewählte Maßnahmen der Qualitätssicherung umzusetzen.
- Erkenntnis der besonderen Bedeutung der Usability als benutzerzentriertes Qualitätsmerkmal.
- Kenntnis der wesentlichen Ansätze, die Usability einer Software zu bewerten und zu gestalten.
- Kenntnis der methodischen Ansätze zur angemessenen Einbeziehung ergonomischer Aspekte in Software-Entwicklungsprozesse, insbesondere Potentiale und Probleme partizipativer Software-Entwicklung.

Inhalt

- Einführung und Motivation
 - Definition des Begriffs „Software-Qualität“
 - Bedeutung der Software-Qualität
- Merkmale der Software-Qualität
- Software-Maße und -Metriken
- Modelle der Software-Qualität
- Einschränkungen der Software-Qualität und ihre Gründe

- Software-Qualitätsmanagement
 - Aufgabenbereiche
 - Grundlegende Prinzipien
- Maßnahmen der Software-Qualitätssicherung
 - Konstruktive Maßnahmen
 - Prozessbezogene Maßnahmen
 - Produktbezogene Maßnahmen
- Analytische Maßnahmen
 - Statische Prüftechniken
 - Dynamische Prüftechniken
- Testen als Maßnahme der Qualitätssicherung
- Black-Box- und White-Box-Testing
- Verfahren des Black-Box-Testing
- Verfahren des White-Box-Testing
 - Graphenbasierte Testfallgenerierung
 - Schnittstellensignaturbasierte Testfallgenerierung
 - Testfallgenerierung nach logischen Kriterien
 - Syntaxbasierte Testfallgenerierung
- Testen eingebetteter Systeme

Literatur

- Ammann, Paul; Offutt, Jeff: Introduction to Software Testing. 1. Auflage. Cambridge, UK: Cambridge University Press, 2008.
- Balzert, Helmut: Lehrbuch der Softwaretechnik : Basiskonzepte und Requirements Engineering. 3. Auflage. Heidelberg: Spektrum Akademischer Verlag, 2009.
- Balzert, Helmut: Lehrbuch der Softwaretechnik : Softwaremanagement. 2. Auflage. Heidelberg: Spektrum Akademischer Verlag, 2008.
- Hoffmann, Dirk W.: Software-Qualität. Berlin: Springer-Verlag, 2008.
- Kneuper, Ralf: CMMI : Verbesserung von Software- und Systementwicklungsprozessen mit Capability Maturity Model Integration. Heidelberg: Dpunkt Verlag, 2007
- Kahn, Stephen H.: Metrics and Models in Software Quality Engineering. 2. Auflage. Boston (MA), USA: Addison-Wesley, 2002.
- Liggesmeyer, Peter: Software-Qualität : Testen, Analysieren und Verifizieren von Software. 2. Auflag. Heidelberg: Spektrum Akademischer Verlag, 2009.
- Schneider, Kurt: Abenteuer Softwarequalität : Grundlagen und Verfahren für Qualitätssicherung und Qualitätsmanagement. 2. Auflage. Heidelberg: Dpunkt Verlag, 2012
- Tian, Jeff : Software Quality Engineering. 1. Auflage. Hoboken (NJ), USA: John Wiley & Sons, 2005.
- Wallmüller, Ernest: Software Quality Engineering : Ein Leitfaden für bessere Software-Qualität. 3. Auflage. München: Carl Hanser Verlag, 2011.

4.28 Systemmodellierung

B087 Systemmodellierung

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B087
Modulbezeichnung	Systemmodellierung
Lehrveranstaltung(en)	B087a Systemanalyse B087b Prozessmodellierung
Modulverantwortliche(r)	Prof. Dr. Andreas Häuslein
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Das Modul baut auf grundlegenden Kenntnissen der Programmierung auf (zum Beispiel „Programmstrukturen 1“ und „Einführung in die Programmierung“). Da es die Sichtweise auf die Programmierung auf die vorgelagerten Phasen der Programmierung im engeren Sinne ausweitet, bildet es Kompetenzen aus, die in allen Modulen verwendbar sind, in denen die Ermittlung von fachlichen Anforderungen als Grundlage einer anschließenden Systementwicklung erforderlich ist. Beispielsweise sind dies die Module „Prozessmodellimplementa-tion“, „Betriebswirtschaftliche Prozesse mit ERP-Systemen“, „Software-Projekt“ und „Bachelor-Thesis“. Das Modul ist sowohl in Informatik- als auch in Wirtschaftsstudiengängen verwendbar.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Kenntnisse der grundlegenden Vorgehensweise bei der Software-Entwicklung, Wissen hinsichtlich der Grundkonzepte von Programmiersprachen, Kenntnisse bezogen auf die Grundfunktionen eines Unternehmens und seinen Aufbau, Fähigkeit zur Abstraktion
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B087a), Abnahme (Teil B087b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Das Modul dient der Vermittlung einer gleichermaßen fach- wie systembezogenen Sicht auf Anwendungs- und entsprechende Software-Systeme. Ein Schwerpunkt ist der Erwerb von

Kenntnis der wesentlichen Entwicklungstätigkeiten und ihrer methodischen Grundlagen, die der Implementierung von Software vorgelagert sind.

Die Studierenden erlangen ein Einschätzungsvermögen hinsichtlich der Notwendigkeit und Grenzen von Systemanalysen, insbesondere in Bezug auf die Gestaltung betrieblicher Informationssysteme. Sie erwerben Kenntnisse der wesentlichen Techniken zur Informationsgewinnung in Unternehmen einschließlich ihrer Vor- und Nachteile. Sie kennen die im Unternehmensumfeld praktisch relevanten methodischen Ansätze zur Systemmodellierung und der damit verbundenen Modellnotationen.

Sie besitzen die Fähigkeit zur Nutzung der Modellierungsmittel zum Aufbau von Analysemodellen für wirtschaftliche Problemstellungen begrenzter Komplexität. Sie können somit an der Ermittlung von fachlichen Anforderungen für eine Systementwicklung mitwirken und eine Systemspezifikation als Ausgangspunkt einer solchen Entwicklung zu erstellen. Sie sind in der Lage, eine prozessorientierte Sichtweise auf die Abläufe in einem Unternehmen einzunehmen und diese für die Modellierung der Systeme als Grundlage zu nutzen.

4.28.1 Systemanalyse

Lehrveranstaltung	Systemanalyse
Dozent(en)	Andreas Häuslein
Hörtermin	5
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Overheadfolien, Handout

Lernziele

Die Studierenden ...

- beurteilen die generellen Möglichkeiten und Grenzen von Systemanalysen, insbesondere in Bezug auf die Gestaltung betrieblicher Informationssysteme.
- unterscheiden die wesentlichen Techniken zur Informationsgewinnung in Unternehmen einschließlich ihrer Vor- und Nachteile, bewerten Techniken im Kontext einer konkreten Informationsgewinnung.
- führen eine methodisch fundierte Informationsgewinnung in einem überschaubaren Problemkontext durch.
- erklären wichtige Bestandteile und Schritte der Systemaufnahme als Vorphase zur Systemmodellierung, sie setzen ausgewählte Formalismen zur Dokumentation der Aufnahmeergebnisse ein.
- stellen die im Unternehmensumfeld praktisch relevanten methodischen Ansätze zur Systemmodellierung dar und beurteilen diese hinsichtlich ihrer Eignung für bestimmte Erkenntnisziele.
- beschreiben die zu den methodischen Ansätzen gehörenden Modellnotationen und setzen diese angemessen zur Modellierung ein.
- nutzen die Modellierungsmittel zum Aufbau von Analysemodellen begrenzter Komplexität für betriebswirtschaftlich ausgerichtete Informationssysteme und diesbezügliche Problemstellungen.

Inhalt

- Grundbegriffe der Systemanalyse
 - Gegenstand und Zielsetzung im Unternehmensumfeld
 - Methodische Grundlagen
- Systemaufnahme
 - Rahmenbedingungen und Techniken der Informationsgewinnung
 - Untersuchungsbereiche bei der Analyse betrieblicher Informationssysteme
- Systemmodellierung
 - Ereignisgesteuerte Prozessketten zur Modellierung von Geschäftsprozessen
 - * Modellelemente schlanker EPK-Modelle
 - * Modellelemente erweiterter EPK-Modelle

- Business Process Model and Notation BPMN
 - * Ausgewählte Modellelemente
 - * Beispielmodelle
 - Strukturierte Analyse und Essenzielle Modellierung
 - * Darstellungs- und Modellierungsmittel
 - * Konsistenzbedingungen
 - * Modellierungsprinzipien der Essenziellen Modellierung
 - * Schritte des Vorgehensmodells
 - Objektorientierte Analyse
 - * Statische Modelle
 - * Dynamische Modelle
- Besonderheiten der Ist-Analyse

Literatur

- KRALLMANN, H.; BOBRIK, A.; LEVINA, O.:Systemanalyse im Unternehmen - Prozessorientierte Methoden der Wirtschaftsinformatik, Oldenbourg, 2013
- RUPP, Chr.:Systemanalyse kompakt, Springer Verlag, 2013
- HEINRICH, G.:Allgemeine Systemanalyse, Oldenbourg, 2007
- HÄUSLEIN, A.:Systemanalyse. vde-Verlag, 2004
- KRÜGER, J.; UHLIG, Ch.:Praxis der Geschäftsprozessmodellierung. VDE Verlag, 2009
- SCHEER, A.-W.:Architektur integrierter Informationssysteme. Springer-Verlag, Berlin, 1991
- OBJECT MANAGEMENT GROUP OMG:Business Process Model and Notation (BPMN), Version 2.0, URL: <http://www.omg.org/spec/BPMN/2.0>, 2011
- BALZERT, Heide:Lehrbuch der Objektmodellierung - Analyse und Entwurf mit der UML 2. Spektrum Akademischer Verlag, 2011
- OESTERREICH, B.:Analyse und Design mit UML 2.3: Objektorientierte Softwareentwicklung. Oldenbourg, 2009
- KECHER, Chr.: UML 2: Das umfassende Handbuch. Galileo Computing, 2011

4.28.2 Prozessmodellierung

Lehrveranstaltung	Prozessmodellierung
Dozent(en)	Christian Uhlig
Hörtermin	5
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assignm.
ECTS	2.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden ...

- nennen und erläutern die theoretischen Grundlagen des Geschäftsprozessmanagement und seine Begriffswelt
- nennen und erläutern die Grundlagen der Geschäftsprozessmodellierung sowie ihre

Einordnung in das Geschäftsprozessmanagement und andere Themenbereiche wie Softwareengineering, Datenbanken und Systemanalyse

- stellen die Grundlagen der ARIS-Methode dar
- wenden wesentliche Modellierungskonzepte des ARIS-Softwaresystems (insbesondere Objekte und Kanten, Definitions- und Ausprägungsebenen sowie Hinterlegungen) an
- nutzen das ARIS-Softwaresystem in seinen wesentlichen Bedienkonzepten und -elementen zur Erstellung von miteinander vernetzten Modellen
- analysieren komplexe textuelle Fachkonzeptbeschreibungen und unterscheiden dabei Inhalte der verschiedenen ARIS-Modellierungssichten (Organisation, Daten, Leistungen, Funktionen, Steuerung)
- entwerfen und gestalten in ästhetisch ansprechender Weise Modelle zentraler Modelltypen (ER-Modell, EPK, WSK, BPMN Process und Collaboration, Organigramm) zu komplexen Fachkonzeptbeschreibungen

Inhalt

- Grundlagen des Geschäftsprozessmanagement
 - Motivation
 - Begriffe
 - Einordnung der Geschäftsprozessmodellierung
 - Bezüge zur Systemanalyse und zum Software-Engineering
- ARIS-Methode
 - Sichtenkonzept
 - Schichtenkonzept
 - Überblick über Modelltypen und ihre Vernetzung
- ARIS-Softwaresystem
- Modellierung der Aufbauorganisation
- ER-Datenmodellierung
- Funktionsmodellierung
- Prozessmodellierung
 - Wertschöpfungsketten (WSK) und Prozesslandkarten
 - EPK / eEPK
 - Vernetzung mit anderen ARIS-Sichten (Daten, Aufbauorganisation)
 - BPMN (Process und Collaboration Diagrams)
- Praktische Aufgabenstellungen
 - Ausschnittsweise und formfreie Modellierung von Prozessen aus einem beispielhaften Fachkonzept
 - Modellierung des Datenmodells zu einem beispielhaften Fachkonzept (ERD)
 - Ausschnittsweise Modellierung von Prozessen zu einem beispielhaften Fachkonzept (WSK / EPK und BPMN)

-
- Ganzheitliche Modellierung von Aufbauorganisation, Datenmodell und Prozessen zu einer Fallstudie (Organigramm, ERD, WSK, BPMN)
-

Literatur

- Krüger, J., Uhlig, C.:
Praxis der Geschäftsprozessmodellierung - ARIS erfolgreich anwenden,
VDE Verlag, 2009
- Lehmann, F.:
Integrierte Prozessmodellierung mit ARIS,
dpunkt.verlag, 2007
- Scheer, A.-W.:
ARIS. Vom Geschäftsprozess zum Anwendungssystem,
4., durchges. Auflage,
Springer, 2006
- Scheer, A.-W.:
ARIS-Modellierungs-Methoden, Metamodelle, Anwendungen,
4. Auflage,
Springer, 2001
- OMG:
Business Process Model and Notation (BPMN) Version 2.0,
2011
- Software AG:
ARIS-Dokumentation (Methodenhandbuch, Bedienhandbücher),
jeweils aktuellste Fassung

4.29 Seminar Informatik

B146 Seminar Informatik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B146
Modulbezeichnung	Seminar Informatik
Lehrveranstaltung(en)	B146a Seminar Informatik
Modulverantwortliche(r)	Prof. Dr. Ulrich Hoffmann
Zuordnung zum Curriculum	Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul bildet eine sinnvolle Vorbereitung auf die Bachelor-Thesis.
SWS des Moduls	2
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 20 Stunden Eigenstudium: 130 Stunden
Voraussetzungen	Veranstaltungsinhalte der ersten vier Studiensemester
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Ziel des Moduls Seminar ist, die Fähigkeit zu erwerben, sich eigenständig in ein anspruchsvolles Informatik-Thema einzuarbeiten zu können und dieses geeignet, sowohl im Rahmen eines Vortrags als auch in Form einer schriftlichen Ausarbeitung darstellen zu können. Als Themen werden dabei aktuelle Entwicklungen in der angewandten Informatik aber auch grundlegende Themen der theoretischen Informatik gewählt. Studierende erlernen, gezielte Literaturrecherchen, insbesondere unter Berücksichtigung der Quellen des Internets, durchzuführen. Sie entwickeln und optimieren ihre Fähigkeiten zur Präsentation des Themas in freien Vorträgen, beim Umgang mit Präsentationsmedien und durch die offene Diskussion wissenschaftlicher Themen in der Gruppe. Die Erstellung einer stilistisch und fachlich ansprechenden Ausarbeitung dient der Vorbereitung auf die Bachelor-Arbeit.

4.29.1 Seminar Informatik

Lehrveranstaltung	Seminar Informatik
Dozent(en)	jeweiliger Dozent
Hörtermin	5
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Seminar
ECTS	5.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout

Lernziele

Die Studierenden ...

- arbeiten sich eigenständig in ein anspruchsvolles Thema über aktuelle Entwicklungen in der angewandten Informatik ein.
- recherchieren gezielt Literatur, insbesondere unter Berücksichtigung der Quellen des Internets.
- optimieren die Fähigkeit des freien Vortragens, des Umgangs mit Präsentationsmedien und der offenen Diskussion wissenschaftlicher Themen in der Gruppe.
- erstellen eine stilistisch und fachlich ansprechende Ausarbeitung als Vorbereitung für die Bachelor-Arbeit.

Inhalt

- nach Aufgabenstellung unterschiedlich
- ca. 10 Einzelfachvorträge von Seminarteilnehmern pro Semester
- Abschlussbericht zum jeweiligen Einzelthema

Literatur

Recherche nach Aufgabenstellung

4.30 Anwendungsentwicklung in ERP-Systemen

B098 Anwendungsentwicklung in ERP-Systemen

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B098
Modulbezeichnung	Anwendungsentwicklung in ERP-Systemen
Lehrveranstaltung(en)	B098a Anwendungsentwicklung in ERP-Systemen B098b Übg. Anwendungsentwicklung in ERP-Systemen
Modulverantwortliche(r)	Dipl.-Inform. (FH) Birger Wolter
Zuordnung zum Curriculum	Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Anwendungsentwicklung in ERP-Systemen“ baut auf die unter anderem in den Modulen „Programmstrukturen 1“ und „Datenbanken 1“ erworbenen Kenntnisse und Fähigkeiten auf und ergänzt die betriebswirtschaftliche Sichtweise des Moduls „Implementierung von Geschäftsprozessen in ERP-Systemen“ um die programmiertechnische Sichtweise.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Programmstrukturen und Datenbankgrundlagen.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B098a), Übung (Teil B098b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls verfügen die Studierenden über grundlegende und weiterführende Kenntnisse von Anwendungssystemen zur Planung und Steuerung betrieblicher Ressourcen (ERP-Systeme) - insbesondere der Programmierung und dem Customizing. Praktisch beherrschen die Studierenden das ERP-System der SAP AG bzw. die Programmiersprache ABAP.

4.30.1 Anwendungsentwicklung in ERP-Systemen

Lehrveranstaltung	Anwendungsentwicklung in ERP-Systemen
Dozent(en)	Birger Wolter
Hörtermin	5
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assignm.
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Softwaredemonstration

Lernziele

Die Studierenden ...

- erwerben systematische Kenntnisse der ABAP-Programmiersprache, als bedeutender Träger der betriebswirtschaftlichen Anwendungslogik in SAP-Systemen.
- erlangen die Fähigkeit, SAP-Systeme zu verstehen und einfache Erweiterungen für spezielle Anwendungsaufgaben vorzunehmen.

Inhalt

- Einführung und erste Schritte
- Werkzeuge der Entwicklungsumgebung
- Grundlegende Konzepte
- Datenbankzugriffe
- Listenverarbeitung (Reports)
- Gestaltung von Bildelementen (Dynpros)
- Besonderheiten von Unternehmenssoftware
- Ausblick: ABAP Objects, Business Server Pages und Web Dynpro

Literatur

- KELLER, Horst; KRÜGER, Sascha: ABAP Objects. ABAP-Programmierung mit SAP NetWeaver. Galileo Press, 3. Auflage 2006
- KELLER, Horst; THÜMMEL, Wolf Hagen: ABAP - Das umfassende Handbuch. Galileo Press, 1. Auflage 2014
- WOLTER, Birger: Veranstaltungsmaterialien auf dem Handout-Server. <http://www.fh-wedel.de/mitarbeiter/wol/veranstaltungen/erp-systeme/> (kontinuierliche Aktualisierung)

4.30.2 Übg. Anwendungsentwicklung in ERP-Systemen

Lehrveranstaltung	Übg. Anwendungsentwicklung in ERP-Systemen
Dozent(en)	Birger Wolter
Hörtermin	5
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0

Lehr- und Medienform(en)	Softwaredemonstration, studentische Arbeit am Rechner
---------------------------------	---

Lernziele

Die Studierenden können - durch Programmierübungen mit den Werkzeugen der Entwicklungsumgebung - die praktische Umsetzung der theoretisch erworbenen Kenntnisse.

Inhalt

- Datenbankzugriffe
- Listenverarbeitung (Reports)
- Gestaltung von Bildelementen (Dynpros)
- Business Server Pages und Web Dynpro

Literatur

- WOLTER, Birger: Übungsmaterialien auf dem Handout-Server. <http://www.fh-wedel.de/mitarbeiter/wol/verssysteme/> (kontinuierliche Aktualisierung)

4.31 IT-Sicherheit

B122 IT-Sicherheit

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B122
Modulbezeichnung	IT-Sicherheit
Lehrveranstaltung(en)	B122a IT-Sicherheit
Modulverantwortliche(r)	Prof. Dr. Gerd Beuster
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul setzt grundlegende Kenntnisse der Programmierung und des Aufbaus eines Computersystems sowie von Computernetzen voraus. Nach Abschluss des Moduls verfügen die Studierenden über weiterführende Kenntnisse auf den Gebieten Computernetze, Kryptographie und Programmierung. Dies ist insbesondere verwendbar für Tätigkeiten und weiterführende Veranstaltungen im Bereich IT-Sicherheit.
SWS des Moduls	4
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 38 Stunden Eigenstudium: 112 Stunden
Voraussetzungen	Die Studierenden besitzen grundlegende Kenntnisse der Programmierung und des Aufbaus eines Computersystems sowie von Computernetzen.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur + ggf. Bonus
Anteil an Gesamtnote	2,7
Sprache	deutsch/englisch

Lernziele des Moduls

Die Studierenden erwerben die notwendigen Kenntnisse, um Software-Systeme und ihre betrieblichen Einsatzszenarien in Hinblick auf Ihre Sicherheit einschätzen zu können. Weiterhin sind die Studierenden nach Absolvierung des Moduls in der Lage, bei der Konzeption und Entwicklung von Software-Systemen und in ihrem Unternehmenseinsatz relevante Sicherheitsaspekte zu berücksichtigen. Die Studierenden erwerben Kenntnisse im Bereich IT-Sicherheit nicht nur bezogen auf einzelne Software-Systeme, sondern auch im Hinblick auf die IT-Infrastruktur. Die Studierenden verfügen über das Wissen der verschiedenen Bedrohungs- und Angriffsarten. Sie kennen die jeweiligen Maßnahmen zur Abwehr der Bedrohungen, insbesondere bei vernetzten Anwendungen.

4.31.1 IT-Sicherheit

Lehrveranstaltung	IT-Sicherheit
Dozent(en)	Gerd Beuster
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	5.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Overheadfolien, Handout, Gastreferenten, E-Learning

Lernziele

Die Studierenden erwerben die notwendigen Kenntnisse, um Softwaresysteme und ihre betrieblichen Einsatzszenarien in Hinblick auf ihre Sicherheit einschätzen zu können. Sie sind in der Lage, bei der Konzeption und Entwicklung von Softwaresystemen und in ihrem Unternehmenseinsatz relevante Sicherheitsaspekte zu berücksichtigen.

- Kenntnis der unterschiedlichen Bedrohungsszenarien und -arten.
- Kenntnis der besonderen Gefahren bei internetbasierten Anwendungen.
- Kenntnis typischer primärer Sicherheitsziele (Vertraulichkeit, Authentifizierung, Verbindlichkeit, u.a.).
- Kenntnis der Verfahren zur Gewährleistung der unterschiedlichen Sicherheitsziele.
- Kenntnis der praxisrelevanten kryptografischen Verfahren und Protokolle.
- Kenntnis der Sicherungsmaßnahmen in Rechnernetzen.
- Fähigkeit, grundlegende Sicherungsmaßnahmen für Web-Anwendungen umzusetzen.
- Kenntnis der Bestandteile einer IT-Sicherheitsinfrastruktur und ihrer zentralen Funktionalitäten.
- Kenntnis der Verfahren zur Risikoabschätzung und Bewertung der Sicherheit von IT-Systemen und die Fähigkeit, diese anzuwenden.

Inhalt

- Gegenstandsbereich der IT-Sicherheit
- Aktuelle Richtlinien, Standards, Normen und Gesetze
- Bedrohungen der IT-Sicherheit und daraus resultierende Risiken
- Primäre Sicherheitsziele
- Überblick über Verfahren zur Erreichung der Ziele
- Kryptografische Verfahren
 - Verschlüsselungsverfahren
 - * Symmetrische Verschlüsselungsverfahren
 - * Asymmetrische Verschlüsselungsverfahren
 - Hash-Funktionen

- Schlüsselmanagement
- Zertifikate
- Kryptografische Protokolle
 - * Digitale Signatur
 - * Zeitstempel
 - * SSL / TLS-Protokoll
- Authentifizierungsverfahren
- Übertragungssicherheit in Netzen
 - Sichere IP-Kommunikation
 - VPN-Technologien
- Sicherheitsarchitekturen und ihre Komponenten
 - Sicherheitsaspekte von Web-Servern
 - Firewall-Systeme
 - Intrusion Detection-Systeme
- Sicherheit von Web-Anwendungen
- Technisch / organisatorische Maßnahmen zur Erhöhung der IT-Sicherheit
- Risiko- und Sicherheitsmanagement

Literatur

- Anderson, Ross J.: Security Engineering : A Guide to Building Dependable Distributed Systems. 2. Auflage. Hoboken (NJ), USA: Wiley & Sons, 2008.
- BSI - Bundesamt für Sicherheit in der Informationstechnik (Hrsg.): Informationssicherheit und IT-Grundschutz : BSI-Standards 100-1, 100-2 und 100-3. 2. Auflage. Köln : Bundesanzeiger Verlag, 2008.
- Eckert, Claudia: IT-Sicherheit : Konzepte - Verfahren - Protokolle. 8. Auflage München : Oldenbourg, 2013.
- Ferguson, Niels; Schneier Bruce, Kohno; Tadayoshi: Cryptography Engineering : Design Principles and Practical Applications. Hoboken (NJ), USA: Wiley & Sons, 2010.
- Kersten, Heinrich; Klett, Gerhard: Der IT Security Manager. 3. Auflage. Wiesbaden: Springer Vieweg, 2013.
- Pfleeger, Charls P.;Pfleeger, Shari Lawrence: Security in Computing. 4. Auflage. München: Prentice Hall, 2012.
- Progunke, Werner: Basiswissen IT-Sicherheit : Das Wichtigste für den Schutz von Systemen & Daten. 3. Auflage. Herdecke: W3L-Verlag, 2013.
- Stallings, William: Computer Security : Principles and Practice. 2. Auflage. München: Pearson, 2012.
- Stallings, William: Cryptography and Network Security : Principles and Practice. 6. Auflage. München: Pearson, 2014.
- Swoboda, Joachim; Spitz, Stephan; Pramateftakis, Michael: Kryptographie und IT-Sicherheit : Grundlagen und Anwendungen. 2. Auflage Wiesbaden: Vieweg + Teubner Verlag, 2011.
- Witt, Bernhard Carsten: IT-Sicherheit kompakt und verständlich : Eine praxisorientierte Einführung. Wiesbaden: Vieweg & Sohn, 2006.

4.32 Soft Skills

B118 Soft Skills

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B118
Modulbezeichnung	Soft Skills
Lehrveranstaltung(en)	B118a Assistenz B118b Communication Skills
Modulverantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Die Inhalte dieses Moduls können gewinnbringend in Projekten, der Bachelor-Thesis und im täglichen Berufsleben genutzt werden. Das Modul sollte in allen Studiengängen verwendet werden.
SWS des Moduls	5
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 47 Stunden Eigenstudium: 103 Stunden
Voraussetzungen	Fachliche Inhalte der ersten 4 Studiensemester
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Anteil an Gesamtnote	0
Sprache	deutsch

Lernziele des Moduls

Nach Abschluss des Moduls haben die Studierenden die Fähigkeit erworben, in Kooperation mit den Dozenten und Assistenten, ihr Wissen und ihre Erfahrungen aus früheren Veranstaltungen der Betriebswirtschaftslehre, Mathematik und Informatik an Studierende jüngerer Semester weiter zu geben. Mit zunehmender Dauer des Semesters verbinden die Studierenden Kenntnisse aus der Veranstaltung „Communication Skills“ mit ihrer Assistenz Tätigkeit.

4.32.1 Assistenz

Lehrveranstaltung	Assistenz
Dozent(en)	verschiedene Dozenten
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_CGT14.0, B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_STec16.0, B_TInf14.0) Wahl (B_MInf14.0, B_WInf14.0)
Lehrform / SWS	Assistenz
ECTS	3.0
Lehr- und Medienform(en)	Tutorien

Lernziele

Die Studierenden entwickeln unter Anleitung eines Hochschullehrers die Fähigkeiten ...

- fachspezifische Aufgabenstellungen zu analysieren
- problemspezifische Lösungen zu konzipieren und
- als Ergebnis begründet zu präsentieren.

Inhalt

Im Rahmen der Assistenz werden die Studierenden von den Hochschullehrern mit konkreten (Teil)-Projekten betraut. Diese können ein weites Spektrum umfassen. So sind z.B. die Durchführung kleinerer empirischer Umfragen oder auch die eigenständige Recherche und Ausarbeitung spezieller Fachinhalte denkbar. Ebenso in Betracht kommen die Durchführung von Tutorien oder Übungen. Die Assistenz ist selbständig zu bearbeiten und kann die Abstimmung mit anderen Studierenden erfordern.

Literatur

keine

4.32.2 Communication Skills

Lehrveranstaltung	Communication Skills
Dozent(en)	Hans-Joachim Göttner
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_WIng14.0, B_BWL14.0, B_CGT14.0, B_EComI14.0, B_EComW14.0, B_IMCA16.0, B_Inf14.0, B_ITE15.0, B_STec16.0, B_TInf14.0) Wahl (B_MInf14.0, B_WInf14.0)
Lehrform / SWS	Workshop
ECTS	2.0
Lehr- und Medienform(en)	Beamerpräsentation, studentische Arbeit am Rechner

Lernziele

Die Studierenden verfügen nach dem Besuch der Lehrveranstaltung über folgende Kompetenzen:

- Besitz verbesserter persönlicher Soft Skills, wie sie für Studium oder Beruf erforderlich

sind

- Sensibilität für menschliche Interaktionen und Betriebsprozesse
- Besitz erweiterter rhetorischer Fähigkeiten im Rahmen von Präsentationen, Vorträgen und Referaten sowie sozialer Kompetenz
- Kenntnis der Bedeutung von verbalen und nonverbalen Signalen für die eigene Kommunikation sowie die Fähigkeit, diese zu erkennen
- Fähigkeit zum angemessenen Verhalten bei Teamarbeit oder Projekten
- Fähigkeit zur Selbstdarstellung bei Bewerbungen, Interviews, Assessment-Centern.

Inhalt

- Anwendung des Kommunikationsmodell von Schulz von Thun
 - Üben situativer und personenbezogener Gesprächsführung
 - Konflikt-handhabung und Klärungsgespräche
- Gruppenarbeit und Ergebnispräsentation
 - betriebliche Fallstudienbearbeitung
 - berufliche Meetings / Protokollführung
 - Verhaltenstraining bei Verkaufsgesprächen
- Unternehmerische Entscheidungsfindung
 - praxisbezogene Postkorbübungen
 - Gesprächsführung mit Mitarbeitern / Fördergespräche / Kritikmanagement
 - Hinweise zur interkulturellen Kompetenz / Verhandlungen

Literatur

- ARNOLD, Frank:
Management von den besten lernen.
München: Hans Hauser Verlag, 2010
- APPELMANN, Björn:
Führen mit emotionaler Intelligenz.
Bielefeld: Bertelsmann Verlag, 2009
- BIERKENBIEHL, Vera F.:
Rhetorik, Redetraining für jeden Anlass. Besser reden, verhandeln, diskutieren.
12. Aufl. München: Ariston Verlag, 2010
- BOLLES, Nelson:
Durchstarten zum Traumjob. Das ultimative Handbuch für Ein-, Um- und Aufsteiger.
2. Aufl. Frankfurt/New York: Campus Verlag, 2009
- DUDENREDAKTION mit HUTH, Siegfried A.:
Reden halten - leicht gemacht. Ein Ratgeber.
Mannheim/Leipzig: Dudenverlag, 2007
- GRÜNING; Carolin; MIELKE; Gregor:
Präsentieren und Überzeugen. Das Kienbaum Trainingskonzept.
Freiburg: Haufe-Lexware Verlag, 2004
- HERTEL, Anita von:
Professionelle Konfliktlösung. Führen mit Mediationskompetenz.
Handelsblatt, Bd., 6, Kompetent managen.
Frankfurt: Campus Verlag, 2009

- HESSE, Jürgen; SCHRADER, Hans Christian:
Assessment-Center für Hochschulabsolventen.
5. Auflage, Eichborn: Eichborn Verlag, 2009
- MENTZEL, Wolfgang; GROTZFELD, Svenja; HAUB, Christine:
Mitarbeitergespräche.
Freiburg: Haufe-Lexware Verlag, 2009
- MORITZ, Andr; RIMBACH, Felix:
Soft Skills für Young Professional. Alles was Sie für ihre Karriere wissen müssen.
2. Aufl. Offenbach: Gabal Verlag, 2008
- PERTL, Klaus N.:
Karrierefaktor Selbstmanagement. So erreichen Sie ihre Ziele.
Freiburg: Haufe-Verlag, 2005
- PORTNER, Jutta:
Besser verhandeln. Das Trainingsbuch.
Offenbach: Gabal Verlag, 2010
- PÜTTJER, Christian; SCHNIERDA, Uwe:
Assessment-Center. Training für Führungskräfte.
Frankfurt/New York: Campus Verlag, 2009
- PÜTTJER, Christian; SCHNIERDA, Uwe:
Das große Bewerbungshandbuch.
Frankfurt: Campus Verlag, 2010
- SCHULZ VON THUN, Friedemann; RUPPEL, Johannes; STRATMANN, Roswitha:
Miteinander Reden. Kommunikationspsychologie für Führungskräfte.
10. Auflage, Reinbek bei Hamburg: rororo, 2003

4.33 Software-Projekt

B121 Software-Projekt

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B121
Modulbezeichnung	Software-Projekt
Lehrveranstaltung(en)	B121a Projektmanagement B121b Softwareprojekt
Modulverantwortliche(r)	Prof. Dr. Uwe Schmidt
Zuordnung zum Curriculum	Informatik (Bachelor) Medieninformatik (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Dieses Modul dient als Vorbereitung für die praktischen Anteile einer Bachelor-Thesis.
SWS des Moduls	6
ECTS des Moduls	10
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 244 Stunden
Voraussetzungen	Voraussetzungen für dieses Modul sind Kenntnisse aus den Veranstaltungen Algorithmen und Datenstrukturen und Fortgeschrittener Objektorientierter Programmierung. Weiter werden Techniken zur Modellierung von Software (Entwurfsmuster, Abstrakte Syntax) vorausgesetzt. Elementare Fähigkeit zur Teamarbeit und Selbstorganisation sind ebenfalls notwendig.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Klausur (Teil B121a), Schriftl. Ausarbeitung (ggf. mit Präsentation) (Teil B121b)
Anteil an Gesamtnote	0
Sprache	deutsch

Lernziele des Moduls

Ziel dieses Projekts ist es, den Einsatz von von OO-Techniken in einer größeren Aufgabe in einem Team zu erfahren. Dabei ist die Themenstellung so gewählt dass zwischen den verschiedenen 2-er-Gruppen die Notwendigkeit der Koordination, Abstimmung und Diskussion, insbesondere über die Schnittstellen der Teilaufgaben, besteht, so dass die Bedeutung der sauberen Entwicklung von Schnittstellen erkannt wird.

Weiter wird die Modellierung und der Entwurf eines Software-Systems an nichttrivialen praxisnahen Problemstellungen trainiert, und so das Wissen aus der Veranstaltung über Software-Design angewandt und gefestigt.

Die soziale Kompetenz, Teamfähigkeit und Eigenverantwortung wird durch selbständige Projektplanung und Projektorganisation einschließlich Aufgabenaufteilung, Zeitplanung und Aufwandsschätzung trainiert.

Die Teamfähigkeit und die Kommunikationsfähigkeit wird gestärkt. Durch die Verwendung

von fertigen Teilsystemen, Bibliotheken und Fremdsoftware, und auch dem Einsatz von Sprachen, die nicht intensiv in den Programmiersprachvorlesungen behandelt werden (Ruby, Python, ...) wird das selbständige Einarbeiten in neue Umgebungen und Systeme trainiert und gefördert

In der Veranstaltung Projektmanagement ist das Ziel das Kennenlernen der grundlegenden Begriffe und Techniken entsprechend den Phasen eines Projekts und die Vermittlung grundlegender Kommunikationsfähigkeiten (präsentieren, diskutieren, moderieren und verhandeln). Als Software-technischer Aspekt steht das Arbeiten im Team und das praktische Anwenden der Techniken des Programmieren im Großen im Mittelpunkt.

4.33.1 Projektmanagement

Lehrveranstaltung	Projektmanagement
Dozent(en)	Martin Schultz
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_WIng14.0) Wahl (B_Inf14.0, B_MInf14.0, B_WInf14.0, B_EComI14.0, B_EComW14.0, B_ITE15.0, B_TInf14.0, B_ITE15.0, B_IMCA16.0)
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- können wesentliche Begriffe des Projektmanagements definieren und diese in Beziehung zueinander setzen
- können die Methoden und Verfahren zur Projektstrukturierung, -organisation, -planung und -steuerung sowie des Projektcontrollings beschreiben und anwenden.
- besitzen die Fähigkeit geeignete Methoden in Abhängigkeit von den Eigenschaften des jeweiligen Projekts auszuwählen.
- besitzen die Fähigkeit Konflikte in Engpasssituationen und im Schnittstellenbereich aus Planungstechniken (Mengengerüst) und Kostengesichtspunkten (monetär bewertetes Mengengerüst) zu identifizieren und dazu Stellung zu nehmen.
- können die vielfältigen Einflussfaktoren auf die erfolgreiche Planung und Steuerung von Projekten im Unternehmen beschreiben und klassifizieren.

Inhalt

Die Bearbeitung von Problem- und Aufgabenstellungen in Form von Projekten gewinnt in der Unternehmenspraxis zunehmend an Bedeutung. Vor diesem Hintergrund soll die Vorlesung wichtige Grundlagen des Projektmanagements vermitteln. Neben den wesentlichen Begriffen werden Methoden zur Projektorganisation, -planung und -steuerung sowie zum Projektcontrolling und Risikomanagement eingeführt und deren Anwendung anhand praxisnaher Beispiele erläutert. Der Aufbau der Vorlesung orientiert sich an den typischen Phasen eines Projekts (Projektdefinition mit Projektantrag, Projektplanung mit Projektplan, Projektkontrolle mit Projektbericht, Projektabschluss mit Abschlussbericht). Darauf aufbauend werden weiterführende Themen des Projektmanagements behandelt wie Programmmanagement, Projektportfoliomanagement, agile Methoden des Projektmanagements, Konfliktmanagement und die Führung interdisziplinärer Teams. Diese Themen werden den Studierenden von ausgewählten Referenten aus der Unternehmenspraxis vorgestellt.

Kurzgliederung

- Grundlegende Begriffe des Projektmanagements
- Konzepte des Projektmanagement
- Projektorganisation

- Projektplanung
- Projektcontrolling, Risikomanagement
- Programmmanagement, Projektportfoliomanagement
- Teamentwicklung und Führung, Konfliktmanagement
- Aktuelle Entwicklungen in der Projektmanagementpraxis

Literatur

- Burghardt, Manfred: Projektmanagement: Leitfaden für die Planung, Überwachung und Steuerung von Projekten. 9. überarb. u. erw. Auflage. Aufl. Erlangen: PUBLICIS, 2012
- Burghardt, Manfred: Einführung in Projektmanagement: Definition, Planung, Kontrolle, Abschluss. 6. aktualis. u. erw. Auflage. Aufl. Erlangen: PUBLICIS, 2013
- Cronenbroeck, Wolfgang: Handbuch internationales Projektmanagement: Grundlagen, Organisation, Projektstandards, interkulturelle Aspekte, angepasste Kommunikationsformen. Berlin: Cornelsen Scriptor, 2004
- Fitzsimons, Conor John; Hoffmann, Hans-Erland; Schoper, Yvonne-Gabriele: Internationales Projektmanagement: Interkulturelle Zusammenarbeit in der Praxis. Auflage: 1. Auflage. Aufl. München: Deutscher Taschenbuch Verlag, 2004
- DeMarco, Tom; Märtine, Doris: Der Termin: ein Roman über Projektmanagement. München : Hanser Verlag, 1998
- Tumascheit, Klaus D: Überleben im Projekt: 10 Projektfallen und wie man sie umgeht. Heidelberg: Redline Wirtschaft, 2007

4.33.2 Softwareprojekt

Lehrveranstaltung	Softwareprojekt
Dozent(en)	Christian-Arved Bohn
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Projekt
ECTS	8.0
Lehr- und Medienform(en)	-

Lernziele

Ziel dieses Projekts ist es, den Einsatz von von OO-Techniken in einer größeren Aufgabe in einem Team zu üben. Dabei ist die Themenstellung so gewählt dass zwischen den verschiedenen 2-er-Gruppen die Notwendigkeit der Koordination, Abstimmung und Diskussion, insbesondere über die Schnittstellen der Teilaufgaben, besteht.

Weiter sollen die Modellierung und der Entwurf eines Software-Systems an nichttrivialen praxisnahen Problemstellungen trainiert werden, und so das Wissen aus der Veranstaltung über Software-Design angewandt und gefestigt werden.

Die soziale Kompetenz, Teamfähigkeit und Eigenverantwortung wird durch selbständige Projektplanung und Projektorganisation einschließlich Aufgabenaufteilung, Zeitplanung und Aufwandsschätzung trainiert. Die Teamfähigkeit und die Kommunikationsfähigkeit werden gestärkt. Durch die Verwendung von fertigen Teilsystemen, Bibliotheken und Fremdsoftware, und auch dem Einsatz von Sprachen, die nicht intensiv in den Programmiersprachvorlesungen behandelt werden (Ruby, Python, ...) wird das selbständiges Einarbeiten in neue Umgebungen

und Systeme trainiert und gefördert.

Inhalt

Variierende Themen für Software-Projekte, unter anderem aus dem Web-Bereich, die mit objektorientierten Techniken zu lösen sind.

Literatur

Uwe Schmidt: Software-Projekt: Organisation und Themen, Unterlagen im Web:<http://www.fh-wedel.de/~si/praktika/SoftwarePraktikum/index.html>

4.34 Operations Research

B082 Operations Research

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B082
Modulbezeichnung	Operations Research
Lehrveranstaltung(en)	B082a Operations Research B082b Übg. Operations Research
Modulverantwortliche(r)	Prof. Dr. Sebastian Iwanowski
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Die Veranstaltung setzt Grundkenntnisse der linearen Algebra, wie sie beispielsweise in der Veranstaltung „Grundlagen der Linearen Algebra“ im Modul „Grundlagen der Mathematik 2“ erworben werden, voraus. Die Kenntnisse aus diesem Modul finden dort Anwendung, wo Optimierungsprobleme mathematisch gelöst werden.
SWS des Moduls	5
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 47 Stunden Eigenstudium: 103 Stunden
Voraussetzungen	Grundlegende Kenntnisse der linearen Algebra.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B082a), Abnahme (Teil B082b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Wichtigstes Lernziel des Moduls ist die Weiterentwicklung des Abstraktionsvermögens der Studierenden. Nach Abschluss des Moduls sind sie in der Lage, Problemstellungen als Operations Research-Aufgaben zu erkennen, aus den Problemstellungen mathematische Modelle abzuleiten und diese in Standardformen zu transformieren. Sie können die gelernten Lösungsverfahren der linearen Optimierung nutzen. Desweiteren verfügen sie über die Fähigkeit, die errechneten Ergebnisse als Lösungen für die gegebene Problemstellung zu interpretieren und kritisch zu bewerten.

4.34.1 Operations Research

Lehrveranstaltung	Operations Research
Dozent(en)	Sebastian Iwanowski
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_EComI14.0, B_EComW14.0, B_ITE15.0, B_WInf14.0, B_WIng14.0) Wahl (B_Inf14.0)
Lehrform / SWS	Vorlesung
ECTS	4.0
Lehr- und Medienform(en)	-

Lernziele

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Kenntnis der mathematischen Methoden des Operations Research.
- Fähigkeit, Problemstellungen als Operations Research-Aufgaben zu erkennen, mathematische Modelle zu entwickeln und diese so in standardisierte Modelle zu transformieren, dass die gelernten Lösungsverfahren angewandt werden können.
- Fähigkeit, im Team komplexe Optimierungsprobleme anhand von Problemstellungen aus der Wirtschaft zu analysieren und dafür die mathematischen Modelle so zu entwickeln und zu transformieren, dass sie unter Verwendung eines Softwaresystems gelöst werden können.
- Fähigkeit zur Ergebnisinterpretation.
- Fähigkeit, sowohl beim Entwurf von Anwendungssystemen Methoden des Operations Research in dieselben zu integrieren als auch die Ergebnisse des Einsatzes von Operations Research-Methoden in diese Systeme als Grundlage für betriebliche Entscheidungsprozesse zu verwenden.

Inhalt

- Einführung in die lineare Optimierung
- Der Simplex-Algorithmus
- Sensitivitätsanalyse
- Das duale Problem
- Ganzzahlige lineare Optimierung
- Das Transportproblem
- Das Zuordnungsproblem
- Zielprogrammierung
- Markov-Ketten
- Netzplantechnik

Literatur

- Domschke, Wolfgang; Drexl, Andreas; Klein, Robert; Scholl, Armin; Voß, Stefan: Übungen und Fallbeispiele zum Operations Research. 7. Auflage. Springer: Berlin, 2011.
- Ellinger, Theodor; Beuermann, Günter; Leisten, Rainer: Operations Research : Eine Einführung. 6. Auflage Berlin Heidelberg: Springer-Verlag, 2003.
- Runzheimer, Bodo: Operations Research 1 : Lineare Planungsrechnung und Netzplantechnik. 8. Auflage. Wiesbaden: Springer Gabler, 2005.
- Suhl, Leena; Mellouli, Taieb: Optimierungssysteme : Modelle, Verfahren, Software, Anwendungen. 3. Auflage. Wiesbaden: Springer Gabler, 2013.
- Taha, Hamdy A.: Operations Research : An Introduction. 9. Auflage. München: Pearson, 2010.
- Werners, Brigitte: Grundlagen des Operations Research : Mit Aufgaben und Lösungen. 3. Auflage. Wiesbaden: Springer Gabler, 2013.
- Winston, Wayne L.: Operations Research : Applications and Algorithms. 4. Auflage. Boston (MA), USA: Cengage Learning Emea, 2003.
- Zimmermann, Werner; Stache, Ulrich: Operations Research : Quantitative Methoden zur Entscheidungsvorbereitung, 10. Auflage. Oldenbourg: Oldenbourg-Verlag, 2001.

4.34.2 Übg. Operations Research

Lehrveranstaltung	Übg. Operations Research
Dozent(en)	Christian Uhlig
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_BWL14.0, B_EComI14.0, B_EComW14.0, B_ITE15.0, B_WInf14.0, B_WIng14.0) Wahl (B_Inf14.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	1.0
Lehr- und Medienform(en)	Studentische Arbeit am Rechner

Lernziele

Studierende ...

- leiten lineare Programme aus informell umschriebenen praktischen Optimierungsproblemen ab
- wenden die Simplexmethode zur eigenständigen Lösung linearer Programme an
- nutzen Softwaretools zur Lösung linearer Optimierungsprobleme
- interpretieren selbständig Lösungstableaus in Hinblick auf ökonomische Planungsprobleme

Inhalt

- Lösung eines linearen Optimierungsproblems mittels Simplexmethode
 - Aufstellen des mathematischen Modells und Ableitung der kanonischen Form des primalen Problems
 - Lösung des primalen Problems
 - Aufstellen des mathematischen Modells und Ableitung der kanonischen Form des dualen Problems
 - Lösung des dualen Problems

- Ergebnis-Interpretation
 - Angabe und Interpretation der Lösung (Basisvariable, Nichtbasisvariable)
 - Interpretation von Schattenpreisen
 - Einordnung in den ökonomischen Kontext
 - Abbildung zwischen dualer und primaler Lösung
 - Erkennen und Interpretieren von Entartung / Mehrdeutigkeit
- Lösung eines linearen Optimierungsproblems mittels Excel-Solver
- Lösung eines linearen Optimierungsproblems mittels GLPK / MathProg
- Mündliche Abnahme der Ergebnisse

Literatur

s. Vorlesung Operations Research

4.35 Grundlagen der Computergrafik

B085 Grundlagen der Computergrafik

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B085
Modulbezeichnung	Grundlagen der Computergrafik
Lehrveranstaltung(en)	B085a Grundlagen der Computergrafik B085b Prakt. Grundlagen der Computergrafik
Modulverantwortliche(r)	Prof. Dr. Christian-Arved Bohn
Zuordnung zum Curriculum	Computer Games Technology (Bachelor) Informatik (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Die generative 3D-Computergrafik berührt inhaltlich einige Grundgedanken der 2D Bildbearbeitung. Das Modul „Bildbearbeitung und -analyse“ bietet sich daher als Kombination mit diesem Modul an.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Grundkenntnisse der Mathematik, insbesondere der linearen Algebra und Vektorrechnung
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B085a), Abnahme (Teil B085b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Lernziele des Moduls sind die erlernten theoretischen Kompetenzen der Vorlesung *Grundlagen der Computergrafik* und die Fähigkeit, diese auch praktisch einsetzen zu können, was im zugehörigen Praktikum vermittelt wird.

4.35.1 Grundlagen der Computergrafik

Lehrveranstaltung	Grundlagen der Computergrafik
Dozent(en)	Christian-Arved Bohn
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_CGT14.0, B_MInf14.0) Wahl (B_Inf14.0, B_STec16.0, B_TInf14.0)
Lehrform / SWS	Vorlesung
ECTS	2.0
Lehr- und Medienform(en)	Handout, Softwaredemonstration

Lernziele

Studierenden werden Fähigkeiten,

- grundlegende Probleme der generativen Computergrafik einzuordnen und zu klassifizieren und
- entsprechende Lösungsstrategien vorzuschlagen und zu implementieren,

vermittelt.

Inhalt

Die Veranstaltung gibt einen Überblick über wesentliche Aspekte der generativen Computergrafik. Konkret werden die Algorithmen des (a) Raytracings und (b) der Projektion als geometrische Abbildung als Basis behandelt und jeweils die Teilaspekte *Projektion*, *Verdeckung* und *Beleuchtungsrechnung* dargestellt. In (b) werden intensiv Methoden der linearen Algebra besprochen, die geometrische Abbildungen zur Konstruktion von virtuellen Szenen und zur Projektion verwenden. Es folgen praktische Aspekte, die es hier zu beachten gilt und deren hardwarenahe Realisierung (z.B. *Clipping*, *Buffer*). Die Technik der Texturierung wird aus mathematischer Sicht behandelt und anhand von praktischen Beispielen erläutert. Einen Einblick in weiterführende Probleme der Computergrafik geben die Grundlagen der globalen Beleuchtungsrechnung (*Rendering Equation*).

Literatur

- Hearn, M. P. Baker: Computer Graphics with OpenGL, Pearson Education International, 2004.
- Peter Shirley et al.: Fundamentals of Computer Graphics, A K Peters, 2005.

4.35.2 Prakt. Grundlagen der Computergrafik

Lehrveranstaltung	Prakt. Grundlagen der Computergrafik
Dozent(en)	Lars Neumann
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_CGT14.0, B_MInf14.0) Wahl (B_Inf14.0, B_STec16.0, B_TInf14.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	3.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Softwaredemonstration

Lernziele

Die Studierenden ...

- sind in der Lage, die in der Vorlesung vermittelten theoretischen Hintergründe selbst praktisch mit OpenGL anzuwenden und so sicher zu handhaben
- haben umfangreiche Kenntnisse und praktische Erfahrungen zu den Themen OpenGL, Callback-Prinzip, 2D-Anwendungen, Matrixstacks, 3D-Szenen, Displaylisten, lokale Beleuchtung, Texturierung, Picking, Viewports und Blending
- haben sich die Grundlagen der Shaderprogrammierung erarbeitet
- steigern ihre Teamfähigkeit durch intensive Arbeit in Zweiertteams und Kommunikation über auftretende Probleme in der ganzen Gruppe

Inhalt

OpenGL, affine Transformationen, 2D- und 3D-Anwendungen, lokale Beleuchtungsmodelle, Texturierung, Picking, Viewports, Transparenz, Shadow-Volumes

Literatur

Skript:

- Vorlesungsskript unter <http://cg.viswiz.de/> => Lehrveranstaltungen => Computergrafik 1
- Weiteres Material unter <http://www.fh-wedel.de/mitarbeiter/ne/praktikum-grundlagen-der-material/>

Online-Quellen:

- The OpenGL Programming Guide - The Redbook (<http://www.glprogramming.com/red/>)
- The OpenGL Reference Manual - The Bluebook (<http://www.glprogramming.com/blue/>)
- Nate Robbins - OpenGL (<http://user.xmission.com/~nate/tutors.html>)
- NeHe Productions (<http://nehe.gamedev.net/>)

Bücher:

- Computergrafik und OpenGL - Eine systematische Einführung, Dieter Orlamünder / Wilfried Mascolus, Hanser, 2004, ISBN: 3-446-22837-3
- Jetzt lerne ich OpenGL : der einfache Einstieg in die Schnittstellenprogrammierung, Lorenz Burggraf, Markt und Technik, 2003, ISBN: 3-8272-6237-2

4.36 Echtzeitsysteme

B101 Echtzeitsysteme

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B101
Modulbezeichnung	Echtzeitsysteme
Lehrveranstaltung(en)	B101a Echtzeitsysteme B101a Interface-Technologie B101b Prakt. Echtzeitsysteme
Modulverantwortliche(r)	Prof. Dr. Sergei Sawitzki
Zuordnung zum Curriculum	Informatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul „Echtzeitsysteme“ baut auf den in den Modulen „Algorithmen und Datenstrukturen“ und (im geringeren Maße) „Systemnahe Programmierung“ erworbenen Kenntnissen und Fähigkeiten auf und vertieft diese im Bezug auf Echtzeitanwendungen. Das Modul kann sinnvoll durch die Module, die allgemeine Aspekte von Betriebssystemen (ohne besonderen Augenmerk auf Echtzeitfähigkeit) vermitteln, ergänzt werden. Auch eine Vertiefung durch die Module, die aktuelle industrielle Standards und Anwendungen betrachten, ist denkbar.
SWS des Moduls	6
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 56 Stunden Eigenstudium: 94 Stunden
Voraussetzungen	Es werden Kenntnisse der Programmiergrundlagen (Datentypen, Programmstrukturen), Basiskenntnisse aus der Informations- und Digitaltechnik (Zahlendarstellungen, Ablauf der Befehlsausführung, Daten- und Steuerfluss) sowie Beherrschung der grundlegenden Methoden und Verfahren der linearen Algebra (Matrixoperationen, Lösung von linearen Gleichungssystemen) vorausgesetzt.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Klausur (Teil B101a), Abnahme (Teil B101b)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Das Modul vermittelt Kenntnisse und Fertigkeiten, die für die Konzipierung, Entwurf, Inbetriebnahme sowie den Umgang mit Echtzeitsystemen benötigt werden. Dabei werden gleichermaßen Software- und Hardware-Konzepte betrachtet. Einerseits werden Methoden und Mechanismen vorgestellt, mit denen Systeme von nebenläufigen, kooperierenden oder

konkurrierenden Prozessen modelliert und implementiert werden. Andererseits werden hardwaretechnische Voraussetzungen für Echtzeit-Betrieb diskutiert, sowie die Unterschiede, die ein Echtzeit-Betriebssystem im Vergleich mit gewöhnlichem Betriebssystem aufweist. Schließlich werden relevante Aspekte der Ereigniserfassung und -verarbeitung in einem Rechnersystem besprochen, untermauert durch Einsatzbeispiele aus der industriellen Praxis. Auch Sicherheits- und Zuverlässigkeitsaspekte werden betrachtet. Durch selbständiges Lösen einer praktischen Aufgabenstellung aus dem Bereich Echtzeitsysteme sollen die Studierenden ihren Lernerfolg überprüfen.

4.36.1 Echtzeitsysteme

Lehrveranstaltung	Echtzeitsysteme
Dozent(en)	Sergei Sawitzki
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_TInf14.0) Wahl (B_Inf14.0, B_STec16.0)
Lehrform / SWS	Vorlesung
ECTS	1.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, Softwaredemonstration

Lernziele

Die Studierenden ...

- bekommen die Kenntnisse vermittelt, die für die systemnahe Programmierung benötigt werden; insbesondere diejenigen Softwarekenntnisse, die für die Programmierung von Embedded Systems, Systemen mit kleinen Hardwareressourcen und Echtzeitsystemen erforderlich sind.
- erlernen Modellierungstechniken und -methoden bei Programmierung von nebenläufigen Prozessen, Modellierung und Anwendung von Prozesskommunikations- und Synchronisationsmechanismen; insbesondere die Fähigkeit, Aufgabenstellungen auf Systeme nebenläufiger (kooperierender und konkurrierender) Prozesse abzubilden und diese softwaretechnisch umzusetzen.
- bekommen eine Einführung in die Methodik der Programmierung paralleler Prozesse, unter besonderer Berücksichtigung von Echtzeitanforderungen vermittelt und erwerben die Befähigung, einfache Echtzeitsysteme zu konzipieren und zu realisieren. Neben Kenntnis wesentlicher theoretischer Konzepte wird besonders auch Fähigkeit zu deren praktischer Umsetzung, die im Projekt Echtzeitsysteme durchgeführt wird, vermittelt.
- weisen durch Entwurf und Realisierung eines Echtzeit-Multitasking-Programms die Kompetenz nach, derartige Systeme aus einem Anforderungskatalog zu entwerfen und praktisch umzusetzen.
- trainieren die Arbeitskoordination, Teamfähigkeit und soziale Kompetenz durch die im Projekt praktizierte Teamarbeit.

Inhalt

- Einleitung
 - Lernziele
 - Organisatorisches
 - Literatur
 - Bezeichnungen und Konventionen
 - Definition und Einordnung
 - Historische Entwicklung
- Prozesse
 - Grundbegriffe
 - Technische Umsetzung

- Aufgaben des Betriebssystems
- Probleme und Lösungsansätze
- Kommunikationsmechanismen
 - Übersicht
 - Semaphore
 - Monitore
 - Mailbox-Kommunikation
 - Nachrichtenaustausch
 - Weitere Mechanismen
 - Äquivalenzen und Beispiele
- Modellierung
 - Einleitung
 - Flussdiagramme
 - Petri-Netze
 - Weitere Modellierungstechniken
- Scheduling
 - Einleitung
 - Strategien
 - Zeitverwaltung
 - Beispiele

Literatur

- Tanenbaum, Andrew: Moderne Betriebssysteme, Pearson Studium, 3. Auflage, 2009
- Witzak, Michael: Echtzeit Betriebssysteme, Franzis Verlag, 2000
- Baumgarten, Bernd: Petri-Netze, Wissenschaftsverlag, 1990
- Labrosse, Jean: MicroC/OS-II, CMP Books, 2002
- Quade, Jürgen; Mächtel, Michael: Moderne Realzeitsysteme kompakt, dpunkt.verlag, 2012

4.36.2 Interface-Technologie

Lehrveranstaltung	Interface-Technologie
Dozent(en)	Dennis Säring
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_TInf14.0) Wahl (B_Inf14.0, B_STec16.0)
Lehrform / SWS	Vorlesung
ECTS	1.5
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Handout, E-Learning

Lernziele

Die Studierenden ...

- sind vertraut mit den grundlegenden Begriffen der Prozessdatenverarbeitung

- besitzen ein Verständnis für die sensorgestützte Aufnahme und Verarbeitung von Echtzeitdaten
- kennen die mathematischen Modelle für eine prädiktive Zustandsschätzungen auf Basis zurückliegender Messergebnisse
- können die Anforderungen an moderne BUS-Systeme im Kontext von komplexen Kommunikationsstrukturen (z.B. PKW, Flugzeug) einschätzen

Inhalt

- Grundlagen der Prozessdatenverarbeitung (PDV)
- Sensortechniken
- Strategien zur Fusionierung von Prozessdaten
- Prädiktionsmodell Kalman- und Partikel-Filter
- Moderne BUS-Systeme

Literatur

- Börcsök: Prozeßrechner- und Automation, Heise-Verlag, 1997
- Jacobsen: Einführung in die Prozeßdatenverarbeitung, Hanser-Verlag, 1996
- Wittgruber: Digitale Schnittstellen und BUS-Systeme, Vieweg-Verlag, 1999

4.36.3 Prakt. Echtzeitsysteme

Lehrveranstaltung	Prakt. Echtzeitsysteme
Dozent(en)	Timm Bostelmann
Hörtermin	6
Art der Lehrveranstaltung	Pflicht (B_TInf14.0) Wahl (B_Inf14.0, B_STec16.0)
Lehrform / SWS	Übung/Praktikum/Planspiel
ECTS	2.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Nachdem Studierende die Veranstaltung besucht haben, können sie

- ausgehend von einer Aufgabenstellung in Form eines Anforderungskatalogs ein System nebenläufiger (kooperierender und konkurrierender) Prozesse entwerfen.
- ein solches System unter Verwendung eines Echtzeitbetriebssystems praktisch umsetzen.
- sich mit Hilfe eines Handbuches in die Schnittstelle eines Echtzeitbetriebssystems einarbeiten.
- ihre Lösung auf System- und Implementierungsebene in einer schriftlichen Ausarbeitung dokumentieren.

Inhalt

- Einführungsvorlesung
 - Motivation zur Veranstaltung

- Beschreibung der Systemumgebung
- Einführung in die Verwendung des Echtzeitkerns
- Einführungsaufgabe
 - Geführtes Erstellen eines Beispielprojektes
 - Teilweise geführte Programmierung eines Prozesses
 - Selbstständige Programmierung eines Prozesses
- Eine Aufgabe aus dem Umfeld Echtzeit, Multitasking, Simulation wird gestellt und steht den Studierenden als Anforderungskatalog zur Verfügung
 - Struktureller Programmentwurf
 - Kodierung und Test
 - Erstellung einer Dokumentation
 - Abnahme durch den Betreuer

Literatur

Labrosse, Jean: MicroC/OS-II, CMP Books, 2002

4.37 Prozessmodellimplementation

B123 Prozessmodellimplementation

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B123
Modulbezeichnung	Prozessmodellimplementation
Lehrveranstaltung(en)	B123a Prozessmodellimplementation
Modulverantwortliche(r)	M.Sc. Christian Uhlig
Zuordnung zum Curriculum	E-Commerce (Bachelor) Informatik (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Wirtschaftsinformatik (Bachelor)
Verwendbarkeit des Moduls	Das Modul baut auf vorgelagerten Modulen zur Software-technik, insbesondere objektorientierter Softwareentwicklung in Java (unter anderem „Fortgeschrittene Objektorientierte Programmierung“), zu Datenbanken („Datenbanken 1“), zur Modellierung von Geschäftsprozessen („Systemmodellierung“) und zu „Web-Anwendungen“ auf. Es vertieft die entsprechenden theoretischen und praktischen Kenntnisse zur Entwicklung eines betrieblichen Anwendungssystems. Es kann sinnvoll mit anderen Modulen zur projektorientierten Softwareentwicklung (zum Beispiel „Software-Projekt“) kombiniert werden.
SWS des Moduls	2
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 20 Stunden Eigenstudium: 130 Stunden
Voraussetzungen	Es wird vorausgesetzt, dass die Teilnehmer Modelle der in der Veranstaltung genutzten Modelltypen (BPMN Process Diagram, BPMN Collaboration Diagram, ERM/ERD) interpretieren und mit dem ARIS-Softwaresystem erstellen können, daher wird als Vorleistung die Veranstaltung Prozessmodellierung gefordert. Für die in der Veranstaltung gelehrt Implementierung einer Web-Anwendung ausgehend von Modellen zu einem gegebenen Fachkonzept wird vorausgesetzt, dass die Teilnehmer die Programmiersprache Java beherrschen und praktische Kenntnisse zu relationalen Datenbanken (insbesondere Formulierung von SQL-Statements und Erstellung von Datenbank-Schemas) und zur Erstellung von Web-Anwendungen besitzen. Daher werden die Veranstaltungen Programmstrukturen 2, Übg. Einführung in Datenbanken und Übg. Web-Anwendungen als Vorleistung gefordert.
Dauer	1 Semester
Häufigkeit	jährlich
Prüfungsformen	Abnahme

Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

Die Studierenden stellen die Charakteristika betrieblicher Anwendungssysteme dar, insbesondere die Unterstützung durch eine zentrale Persistierung von Unternehmensdaten, eine Client-Server-Architektur für den Mehrbenutzerbetrieb und die auf die Pflege von Geschäftsobjekten fokussierte Benutzerführung.

Sie nutzen und vernetzen Modelle zur Beschreibung von Geschäftsprozessen sowie von Modellen des Software-Engineerings, insbesondere im Rahmen der ARIS-Methode und unter Verwendung des ARIS-Softwaresystems. Zu den genutzten Modelltypen zählen schwerpunktmäßig BPMN Process Diagrams, BPMN Collaboration Diagrams sowie ER-Modelle. Die Teilnehmer leiten das Gerüst und Teile der Implementierung eines betrieblichen Anwendungssystems aus gegebenen Modellen zur Spezifikation des Fachkonzeptes ab und dokumentieren das entstehende Software-Systeme unter Zuhilfenahme von Modellen.

Die Teilnehmer erklären die Abgrenzung von Desktop- und Web-Anwendungen im Kontext betrieblicher Anwendungssysteme und wählen geeignet für ein Anwendungsszenario aus. Sie wenden die Sprache Java und dafür verfügbare Spezifikationen (z., B. Java Persistence API) und Frameworks (z., B. Google Web Toolkit) an, um betriebliche Anwendungssysteme als Web-Anwendungen zu implementieren.

4.37.1 Prozessmodellimplementierung

Lehrveranstaltung	Prozessmodellimplementierung
Dozent(en)	Christian Uhlig
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Vorlesung mit integrierter Übung/Workshop/Assigm.
ECTS	5.0
Lehr- und Medienform(en)	Beamerpräsentation, Handout, Softwaredemonstration, studentische Arbeit am Rechner

Lernziele

Die Studierenden ...

- skizzieren die Grundlagen von Softwareprojekten und des Softwareentwicklungszyklus (Spezifikation, Entwurf und Implementierung).
- setzen Geschäftsprozessmodelle (EPK, BPMN) zur Fachkonzeptbeschreibung in Beziehung zum Entwurf und zur Implementierung unterstützender Anwendungssysteme.
- verbinden Geschäftsprozessmodelle (BPMN) mit Modell-Elementen des Softwareentwurfs (z. B. Datenobjekte, Maskendesign, Klassenhierarchie, usw.).
- identifizieren und entwerfen die notwendigen Bedienelemente (insbesondere Bildschirmmasken) zur Unterstützung bestimmter Funktionen eines gegebenen Geschäftsprozesses.
- unterscheiden Desktop- und Web-Anwendungen und ihre wesentlichen Merkmale.
- vergleichen kritisch verschiedene Ansätze zur Persistierung und zur Umsetzung von Web-Anwendungen.
- wählen aus und nutzen Konzepte, Programmierschnittstellen und Frameworks der Java-Landschaft zur Erstellung von webbasierten Anwendungssystemen mit Datenbankunterstützung (J2EE, JPA, JTA, usw.).
- entwerfen und entwickeln datenbankgestützte Web-Anwendungen zur Unterstützung von Geschäftsprozessen auf Basis der Programmiersprache. Java und wichtiger damit verbundener Konzepte und Frameworks
- setzen eine Modelllandschaft in Bezug zu Einheiten der Softwaretechnik (Klassen zu Business Objects, Klassen zu Masken, usw.) und leiten dabei wesentliche Teile des Systemgerüsts systematisch aus Modellinhalten ab.
- wenden das Google Web Toolkit (GWT) an.
- wenden die Java Persistence API (JPA) an.

Inhalt

- Architekturen betrieblicher Anwendungssysteme
 - Client-Server-Architekturen mit persistenter Datenhaltung
 - Web-Applikationen und Web-Services
- Objektorientierte Web-Anwendungsentwicklung mit Java

- Applikationsframeworks für Web-Anwendungen
- Google Web Toolkit (GWT)
- Objekt-relationale Abbildung mit der Java Persistence API
- Modelle der ARIS-Methode zur Verbindung von Software-Engineering mit Geschäftsprozessmodellierung
- Praktische Aufgabenstellungen
 - Prozess- und Datenmodellierung zu einem gegebenen Fachkonzept
 - Modelle zur Spezifikation des zu implementierenden Anwendungssystems (z. B. Maskendesign)
 - Entwurf und Implementierung eines Anwendungssystems zu einem Ausschnitt des modellierten Fachkonzeptes

Literatur

- Krüger, J., Uhlig, C.: Praxis der Geschäftsprozessmodellierung-ARIS erfolgreich anwenden, VDE Verlag, 2009
- Lehmann, F.: Integrierte Prozessmodellierung mit ARIS, dpunkt.verlag, 2007
- Scheer, A.-W.: ARIS. Vom Geschäftsprozess zum Anwendungssystem, 4., durchges. Auflage, Springer, 2006
- Scheer, A.-W.: ARIS-Modellierungs-Methoden, Metamodelle, Anwendungen, 4. Auflage, Springer, 2001
- Tacy, A., Hanson, R., Essington J.: Gwt in Action, Manning Publications, 2013
- Müller, B., Wehr, H.: Java Persistence API 2: Hibernate, EclipseLink, OpenJPA und Erweiterungen, Hanser Fachbuch, 2012
- DeMichiel, L., Shannon, B.: Java Platform, Enterprise Edition (Java EE) Specification v7, Oracle 2013
- Java Persistence 2.0 Expert Group: JSR 317: Java Persistence API, Version 2.0, Sun Microsystems, 2009
- Software AG: ARIS-Dokumentation (Methodenhandbuch, Bedienhandbücher), jeweils aktuellste Fassung

4.38 Auslandssemester

B099 Auslandssemester

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B099
Modulbezeichnung	Auslandssemester
Lehrveranstaltung(en)	B099a Auslandssemester
Modulverantwortliche(r)	Dipl.-Soz. (FH) Nicole Haß
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Beschreibung Modulverwendbarkeit fehlt
SWS des Moduls	25
ECTS des Moduls	30
Arbeitsaufwand	Präsenzstudium: 227 Stunden Eigenstudium: 673 Stunden
Voraussetzungen	Für eine Zulassung müssen alle Übergangsleistungen gemäß § 16a der Prüfungsverfahrensordnung und insgesamt mindestens 45 ECTS-Punkte erfolgreich absolviert sein.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Ausland
Anteil an Gesamtnote	28,11
Sprache	deutsch

Lernziele des Moduls

Die fachlichen Lernziele dieses Moduls werden von den ausländischen Hochschulen festgelegt. Die FH Wedel prüft, ob diese Ziele inhaltlich vergleichbar und aner kennbar sind mit den eigenen Zielen.

Im Bereich soziale Kompetenz ist das Ziel das Kennenlernen einer anderen sprachlichen und kulturellen Umgebung und das Arbeiten und Kommunizieren in dieser. Außerdem natürlich das Erlernen und/oder Festigen einer Fremdsprache.

4.38.1 Auslandssemester

Lehrveranstaltung	Auslandssemester
Dozent(en)	Nicole Haß
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Veranstaltungen an ausländischer Hochschule
ECTS	30.0
Lehr- und Medienform(en)	Keine

Lernziele

Nach Abschluss des Auslandssemester besitzen die Studierenden ...

- fundierte Sprachkompetenzen in englischer, französischer oder spanischer Sprache.
- erweiterte Kenntnisse über die Kultur des Gastlandes.

Inhalt

Verpflichtendes Auslandssemester:

Für ein verpflichtendes Auslandssemester muss der Umfang der erfolgreich zu erbringenden Leistungen (ohne Englisch-Sprachkurse) mindestens 30 ECTS-Punkte betragen oder einen entsprechenden gleichwertigen Umfang in lokalen Credits aufweisen. An der ausländischen Hochschule sind fachspezifische weiterführende und keine Grundlagenkurse zu belegen. Diese sollen im Zusammenhang mit dem Wedeler Studiengang stehen (hinsichtlich der zu belegenden Fächer gemäß Modulhandbuch).

Freiwilliges Auslandssemester:

Für ein freiwilliges Auslandssemester ist der Umfang der zu leistenden ECTS-Punkte (bzw. der gleichwertige Umfang in lokalen Credits) in der jeweiligen Studienordnung vorgegeben. An der ausländischen Hochschule sind fachspezifische Kurse zu belegen, die mit dem in Wedel belegten Studiengang in ergänzendem Zusammenhang stehen. Das Studienprogramm wird vor der Abreise individuell mit dem International Office vereinbart.

Literatur

abhängig von der ausländischen Hochschule

4.39 Praxissemester (dual)

B176 Praxissemester (dual)

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B176
Modulbezeichnung	Praxissemester (dual)
Lehrveranstaltung(en)	B176a Praxissemester (dual)
Modulverantwortliche(r)	Dipl.-Kauff. (FH) Journalistin Anke Amsel
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Verwendung der erworbenen Fähigkeiten in der späteren praxisorientierten Bachelor-Thesis.
SWS des Moduls	20
ECTS des Moduls	25
Arbeitsaufwand	Präsenzstudium: 182 Stunden Eigenstudium: 568 Stunden
Voraussetzungen	
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Praktikumsbericht / Protokoll
Anteil an Gesamtnote	0
Sprache	deutsch

Lernziele des Moduls

Das Praxissemester bietet den Studierenden die Möglichkeit eine Verbindung von studien-gangsspezifischem und unternehmensspezifischem Kompetenzprofil herauszubilden.

Die Studierenden sollen im Kooperationsunternehmen in einer Vielzahl von Tätigkeitsfeldern qualifiziert an einem größeren Projekt mit Bezug zum Studiumsziel in eigener Verantwortung unter Anleitung erfahrener Mitarbeiter mitwirken. Die projektbezogene betriebliche Tätigkeit kann sich auf mehrere unabhängige Teilprojekte erstrecken.

Dabei sollen sich die Studierenden mit Leitfragen ihres Studiengangs auseinandersetzen. Die inhaltliche Vertiefung kann durch die Einbindung des Kooperationsunternehmens teilweise über das Lehrangebot der FH Wedel hinausgehen.

Durch das projektbezogene Arbeiten werden analytische, organisatorische, kommunikative und repräsentative Techniken eingeübt sowie die Verbindung zu Anwendungsgebieten des Studiengangs hergestellt. Ziel ist der Theorietransfer in die jeweiligen betrieblichen Funktionsbereichen. Berufliche Realität soll erlebt und erlernt werden. Die Studierenden wählen

wissenschaftliche Methoden, um Aufgaben des Berufslebens zu lösen.

Den Nachweis, dass sie ihr erlerntes Wissen auf eine anwendungsbezogene Aufgabenstellung aus einem Fachgebiet selbstständig auf wissenschaftlicher Grundlage im Rahmen des festgelegten Themas anwenden können, erbringen die Studierenden im Rahmen einer wissenschaftlichen Arbeit.

4.39.1 Praxissemester (dual)

Lehrveranstaltung	Praxissemester (dual)
Dozent(en)	Anke Amsel
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Praktikum
ECTS	25.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- können das Wissen aus dem bisherigen Studium in der Praxis anwenden und hinsichtlich der Tauglichkeit kritisch bewerten
- erweitern ihre wissenschaftlichen Ausbildung durch systematische praktische Erfahrungen
- können Projekten vorbereiten, analysieren und im Nachgang evaluieren
- bewerten Problemstellungen und können Lösungsansätze dafür entwickeln
- können Projektmanagement betreiben, Aktivitäten koordinieren, Planabweichungen hinterfragen.
- sehen und bewerten unternehmensweite und gesellschaftliche Zusammenhänge der eigenen Tätigkeit und zeigen ihre professionelle persönliche Qualifikation in der Zusammenarbeit mit Vorgesetzten und Kollegen
- nehmen Stellung zu den sozial-, datenschutz- oder umweltschutzbedingten Restriktionen bei der Umsetzung von betrieblichen Anforderungen
- reflektieren ihre Qualifikation und ihre eigene Tätigkeit
- sind zum selbständigen und eigenverantwortlichen Arbeiten fähig
- übernehmen Verantwortung für die Qualität der eigenständig übernommenen Arbeit
- entscheiden sich für systematische Vorgehensweisen und Arbeitstechniken
- klassifizieren ihre Tätigkeit zu Anwendungsgebieten des Studiengangs
- sind in der Lage das Wissen aus dem bisherigen Studium in der Praxis anzuwenden, fortzubilden und hinsichtlich der Tauglichkeit kritisch zu bewerten.

Inhalt

Der Inhalt des „Praxissemesters“ muss mit der Zielrichtung des Studienganges vereinbar sein. Zur Erreichung dieses Ziel werden je Studiengang entsprechende Leitfragen formuliert. Von diesen sollen während des Praxissemester mindestens vier Themenkomplexen abgearbeitet und mindestens ein Themenkomplex vertieft werden.

Ausgehend von den Modulzielen des jeweiligen Studiengangs legt die/der Hochschulbetreuer in Absprache mit der/dem Studierenden fest, welche Themenkomplexe im Unternehmen bearbeitet werden sollen.

Literatur

themenabhängig

4.40 Wissenschaftliche Ausarbeitung (dual)

B179 Wissenschaftliche Ausarbeitung (dual)

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B179
Modulbezeichnung	Wissenschaftliche Ausarbeitung (dual)
Lehrveranstaltung(en)	B179a Wissenschaftliche Ausarbeitung (dual)
Modulverantwortliche(r)	Dipl.-Kauff. (FH) Journalistin Anke Amsel
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	
SWS des Moduls	3
ECTS des Moduls	5
Arbeitsaufwand	Präsenzstudium: 29 Stunden Eigenstudium: 121 Stunden
Voraussetzungen	
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Anteil an Gesamtnote	2,7
Sprache	deutsch

Lernziele des Moduls

4.40.1 Wissenschaftliche Ausarbeitung (dual)

Lehrveranstaltung	Wissenschaftliche Ausarbeitung (dual)
Dozent(en)	jeweiliger Dozent
Hörtermin	6
Art der Lehrveranstaltung	Wahl
Lehrform / SWS	Praktikum
ECTS	5.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden sind in der Lage, innerhalb einer vorgegebenen Frist ein Themenkomplex des Praxissemesters selbstständig wissenschaftlich zu bearbeiten und diesen kritisch zur praktischen Anwendung zu betrachten.

Inhalt

Die wissenschaftliche Arbeit ist als abschließende, vom Studierenden eigenständig aber hochschul- und unternehmensseitig betreute Ausarbeitung zum Praxissemester zu verstehen. Der Themenkomplex wird im Sinne der Zielsetzung des Praxissemesters mit der/dem hochschulseitigen Betreuer(in) abgestimmt und soll Bezüge zur betrieblichen Praxis aufweisen.

Literatur

themenabhängig

4.41 Betriebspraktikum

B159 Betriebspraktikum

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B159
Modulbezeichnung	Betriebspraktikum
Lehrveranstaltung(en)	B159a Betriebspraktikum
Modulverantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	
SWS des Moduls	0
ECTS des Moduls	17
Arbeitsaufwand	Präsenzstudium: 2 Stunden Eigenstudium: 508 Stunden
Voraussetzungen	
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Praktikumsbericht / Protokoll
Anteil an Gesamtnote	0
Sprache	deutsch

Lernziele des Moduls

Die Studierenden sammeln Erfahrungen für die spätere berufliche Tätigkeit in einem Unternehmen.

Dies dient der Stärkung der beruflichen und sozialen Kompetenzen: Selbständigkeit, eigenverantwortliches Handeln, Kommunikation, Teamfähigkeit und Zeitmanagement.

4.41.1 Betriebspraktikum

Lehrveranstaltung	Betriebspraktikum
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Betriebliches Praktikum
ECTS	17.0
Lehr- und Medienform(en)	-

Lernziele

Das Betriebspraktikum ist ein zentraler Baustein für die berufliche Profilbildung der Studierenden. Es ermöglicht im Rahmen des Studiums einen direkten Kontakt zu Unternehmen, die von den Studierenden eigenständig angesprochen werden. Der Kontakt zum Unternehmen soll helfen, die im bisherigen Verlauf des Studiums angeeignete Fach- und Methodenkompetenz auf ausgewählte Abläufe und Problemstellungen des betrieblichen Alltags zu übertragen. Hierbei werden auch soziale Kompetenzen erprobt und gefestigt.

Inhalt

Das Betriebspraktikum soll vertieften Einblick in Prozesse und Aufbau eines Betriebes geben. Der oder die Studierende sucht sich das Betriebspraktikum mit Hilfe der Praktikadatenbank der Fachhochschule Wedel oder anderen Informationsquellen (z.B. Aushänge, Internetseiten des Wedeler Hochschulbundes). Bei Problemen bietet die Hochschule Hilfestellung. Die Tätigkeit kann im Rahmen des Tagesgeschäftes oder in einer Projektarbeit durchgeführt werden. Es wird aus Sicht der Hochschule angestrebt, dass das Betriebspraktikum als Vorlaufphase für eine sich unmittelbar anschließende Bachelorarbeit beim gleichen Unternehmen genutzt wird. Das Betriebspraktikum soll daher inhaltlich eine Brücke zur nachfolgenden Bachelorarbeit sein. Einsatzfelder sind in Absprache mit dem Unternehmen und dem oder der Dozent/in so zu wählen, dass sie auch gut geeignet sind, eine Fragestellung für eine mögliche nachfolgende Bachelorarbeit zu entwickeln.

Literatur

themenabhängig

4.42 Bachelor-Thesis

B150 Bachelor-Thesis

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B150
Modulbezeichnung	Bachelor-Thesis
Lehrveranstaltung(en)	B150a Bachelor-Thesis
Modulverantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Keine.
SWS des Moduls	0
ECTS des Moduls	12
Arbeitsaufwand	Präsenzstudium: 2 Stunden Eigenstudium: 358 Stunden
Voraussetzungen	Voraussetzung ist das Wissen aus den Veranstaltungen der sechs vorangegangenen Semester, insbesondere der Veranstaltungen, die mit dem Themengebiet der Abschlussarbeit zusammenhängen.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Anteil an Gesamtnote	12,97
Sprache	deutsch

Lernziele des Moduls

An das Betriebspraktikum schließt sich die Bachelor-Arbeit an, die sehr praxisorientiert fast ausschließlich in Unternehmen angefertigt wird und deren Themenstellung sich in enger Kooperation zwischen FH Wedel und dem jeweiligen Unternehmen in der Regel aus dem betrieblichen Umfeld ergibt.

Die Studierenden sollen mit ihrer Arbeit den Nachweis erbringen, dass sie ihr erlerntes Wissen auf eine anwendungsbezogene Aufgabenstellung aus einem Fachgebiet selbstständig auf wissenschaftlicher Grundlage im Rahmen des festgelegten Themas anzuwenden. Wesentlich sind strukturierte und argumentierte Inhalte sowie das Einhalten üblicher Formalia.

Dies dient der Vertiefung und des konkreten Einsatzes der fachliche Kompetenzen: Methodisches Arbeiten und praktisches Anwenden der im Studium erlernten Kenntnisse.

4.42.1 Bachelor-Thesis

Lehrveranstaltung	Bachelor-Thesis
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Thesis
ECTS	12.0
Lehr- und Medienform(en)	-

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit zur Durchführung einer praxisorientierten Arbeit.
- können eine Fragestellung selbständig erarbeiten.
- können die zu erarbeitende Problematik klar strukturieren.
- können die Vorgehensweise und Ergebnisse in einer Ausarbeitung übersichtlich darstellen.
- stärken ihre praktischen Fähigkeiten im Projektmanagement-Bereich und zur Selbstorganisation.

Inhalt

Die Bachelor-Thesis soll im Regelfall in Kooperation mit einem Unternehmen erarbeitet werden. Themen aus den Arbeitsgruppen und Laboren der Hochschule sind ebenfalls möglich. Die Arbeit ist als abschließende, vom Studierenden eigenständig aber hochschul- und unternehmensseitig betreutes Projekt zu verstehen. Im Sinne der Zielsetzung der Bachelor-Ausbildung, der Erlangung des ersten berufsqualifizierenden Abschlusses, ist die Arbeit thematisch an einer Problemstellung eines kooperierenden Unternehmens orientiert oder sie besteht aus einer praxisrelevanten hochschulinternen Aufgabe.

Literatur

themenabhängig

4.43 Bachelor-Kolloquium

B160 Bachelor-Kolloquium

Studiengang	Bachelor-Studiengang Informatik
Modulkürzel	B160
Modulbezeichnung	Bachelor-Kolloquium
Lehrveranstaltung(en)	B160a Kolloquium
Modulverantwortliche(r)	jeweiliger Dozent
Zuordnung zum Curriculum	Betriebswirtschaftslehre (Bachelor) Computer Games Technology (Bachelor) E-Commerce (Bachelor) Informatik (Bachelor) IT-Engineering (Bachelor) IT-Management, -Consulting & -Auditing (Bachelor) Medieninformatik (Bachelor) Smart Technology (Bachelor) Technische Informatik (Bachelor) Wirtschaftsinformatik (Bachelor) Wirtschaftsingenieurwesen (Bachelor)
Verwendbarkeit des Moduls	Keine.
SWS des Moduls	1
ECTS des Moduls	1
Arbeitsaufwand	Präsenzstudium: 11 Stunden Eigenstudium: 19 Stunden
Voraussetzungen	Zulassungsvoraussetzung zum Kolloquium ist eine mit mindestens "ausreichend" bewertete Bachelor-Thesis.
Dauer	1 Semester
Häufigkeit	jedes Semester
Prüfungsformen	Kolloquium
Anteil an Gesamtnote	0,54
Sprache	deutsch

Lernziele des Moduls

Das Kolloquium ist eine fächerübergreifende mündliche Prüfung, ausgehend vom Themenkreis der Bachelor-Thesis, und ist die letzte Prüfungsleistung, welche das Studium abschließt. In der mündlichen Abschlussprüfung halten die Studierenden einen Fachvortrag über das von ihnen bearbeitete Thema und verteidigen ihre Bachelor-Thesis in einer anschließenden Diskussion. Dies stärkt die Fähigkeit, ein intensiv bearbeitetes Themengebiet, zusammenfassend darzustellen und professionell zu vertreten.

4.43.1 Kolloquium

Lehrveranstaltung	Kolloquium
Dozent(en)	jeweiliger Dozent
Hörtermin	7
Art der Lehrveranstaltung	Pflicht
Lehrform / SWS	Kolloquium
ECTS	1.0
Lehr- und Medienform(en)	Tafel, Beamerpräsentation

Lernziele

Die Studierenden ...

- besitzen die Fähigkeit der konzentrierten Darstellung eines intensiv bearbeiteten Fachthemas.
- verfestigen die Kompetenz, eine fachliche Diskussion über eine Problemlösung und deren Qualität zu führen.
- verfügen über ausgeprägte Kommunikations- und Präsentationsfähigkeiten.

Inhalt

- nach Thema der Bachelor-Arbeit unterschiedlich
- Fachvortrag über das Ergebnis der Bachelor-Arbeit
- Diskussion der Qualität der gewählten Lösung
- Fragen und Diskussion zum Thema der Bachelor-Arbeit und verwandten Gebieten

Literatur

themenabhängig