

# Wahlpflichtfächer

der Studiengänge

- Angewandte Informatik,
- Technische Informatik und
- Wirtschaftsinformatik

Die in einem Semester zur Wahl angebotenen Wahlpflichtfächer sowie die aktuellen Randbedingungen (Voraussetzungen und Beschränkungen) sind den Aushängen zu entnehmen.

Modulbezeichnung	<b>3D Programmierung: Einführung in die Computergrafik</b>	Kürzel	WP-ECG WPP-ECG
Lehrveranstaltung(en)	Vorlesung: 3D Programmierung: Einführung in die Computergrafik Praktikum: 3D Programmierung: Einführung in die Computergrafik	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Philipp Jenke	SWS	2+2
Dozenten	Prof. Dr. Philipp Jenke	Sprache	deutsch
Voraussetzungen	PR1, PR2, Java	Häufigkeit	semesterweise zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen die Rendering-Pipeline, wissen also wie dreidimensionale Szenen repräsentiert und für die zweidimensionale Ausgabe aufbereitet werden</li> <li>• können moderne Algorithmen der Computergrafik unter Einsatz einer hardwaregestützten Grafikpipeline anwenden</li> <li>• können die grundlegenden relevanten Gebiete der algorithmischen Geometrie auf Anwendungen in der Computergrafik übertragen</li> <li>• verstehen die relevanten Datenstrukturen und Methoden zur effizienten Bildgenerierung und können diese in Beispielen umsetzen.</li> <li>• können kleine Anwendungen zur Repräsentation und Darstellung von dreidimensionalen Szenen erstellen.</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Elementare Begriffe wie Punkte, Linien, Vektoren, Transformationen, Matrizen, Netze</li> <li>• Grundlagen der Beleuchtungsrechnung mit Farbe, Licht und Texturen</li> <li>• Grundkonzepte der Datenrepräsentation und Modellierung mit Kurven, Flächen und polygonalen Netzen</li> <li>• Einführung in erweiterte Konzepte der Grafikprogrammierung mit Themen wie Sichtbarkeitsberechnung, Clipping, hierarchische Datenstrukturen</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Overhead-/Rechnerpräsentationen, freiwillige Übungsaufgaben, Kleingruppenarbeit Praktikum: Programmieren in Zweiergruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Gerhard Virag: Grundlagen der 3D-Programmierung - Mathematik und Praxis mit OpenGL, Open Source Press (2012)</li> <li>• Frank Klawonn: Grundkurs Computergrafik mit Java - Die Grundlagen verstehen und einfach umsetzen mit Java3D, Vieweg und Teubner, dritte Auflage (2010)</li> <li>• Tomas Akenine-Moeller, Eric Haines: Real-Time Rendering, dritte Auflage (2008)</li> <li>• John F. Hughes, Andries van Dam, Morgan McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akeley: Computer Graphics, Principles and Practice, zweite Auflage (1996), dritte Auflage (erscheint 2013)</li> <li>• eigene Skripte und Folien der Dozenten</li> </ul>		

Modulbezeichnung	<b>Abbildung betriebswirtschaftlicher Aufgaben</b>	Kürzel	WP-ABA WPP-ABA
Lehrveranstaltung(en)	Vorlesung: Abbildung betriebswirtschaftlicher Aufgaben Praktikum: Abbildung betriebswirtschaftlicher Aufgaben	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Brigitte Braun, NN2	SWS	3+1
Dozenten	Prof. Dr. Brigitte Braun, NN2	Sprache	deutsch
Voraussetzungen	ABWL-Grundlagen	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• können komplexe, betriebswirtschaftliche Aufgabenstellungen mit Hilfe von Tabellenkalkulationsprogrammen lösen</li> <li>• können Aufgabenstellungen analysieren und in geeignete Strukturen überführen</li> <li>• können Funktionen nutzen, Abläufe automatisieren und den Funktionsumfang durch Programmierung erweitern</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Anhand ausgewählter Fallbeispiele soll vermittelt werden, wie betriebswirtschaftliche Lösungsverfahren formuliert und geeignet umgesetzt werden. Beispiele sind: Wirtschaftlichkeitsrechnungen, Finanzmathematische Analysen, Was-wäre-wenn-Analysen, Häufigkeitsanalysen, ABC-Analysen in der Logistik, im Vertrieb und in der Kostenrechnung, Lineare Optimierung, Regressionen zur Kostenauflösung, in der Absatzplanung und für Prognosen, Pivottabellen, ...</li> <li>• Ferner soll vermittelt werden, wie durch die Nutzung der integrierten Entwicklungsumgebung Abläufe automatisiert, eigene Funktionen und Formulare erstellt und genutzt werden können</li> </ul>		
Lehr- und Lernformen	Vorlesung: seminaristischer Unterricht, Präsentationen Praktikum: Vertiefung der Aufgabenstellung und Bearbeitung von weiterführenden Aufgabenstellungen,  Die Mitarbeit am eigenen Notebook, Netbook o.ä. während des seminaristischen Unterrichts ist ausdrücklich erwünscht.		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Skript und Aufgabenheft, aktuelle Literaturliste ist jeweils im Skript enthalten</li> </ul>		

Modulbezeichnung	<b>Analytische Informationssysteme</b>	Kürzel	WP-AIS WPP-AIS
Lehrveranstaltung(en)	Vorlesung: Analytische Informationssysteme Praktikum: Analytische Informationssysteme	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Wolfgang Gerken	SWS	3+1
Dozenten	Prof. Dr. Wolfgang Gerken	Sprache	deutsch
Voraussetzungen	Datenbanken, BWL 2	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• verstehen die Architekturen und Beispiele analytischer Informationssysteme</li> <li>• kenne das Data Warehouse-Konzept und Anwendungen davon</li> <li>• können ein Data Warehouse modellieren</li> <li>• verstehen Data Mining-Algorithmen können diese anwenden</li> <li>• können Datenbankstrukturen im DWH-Kontext beurteilen</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Einführung</li> <li>• Data Warehouse</li> <li>• Modellierung eines DWH</li> <li>• Anwendungen eines Data Warehouse (Reporting, Online Analytical Processing, Data Mining)</li> <li>• Algorithmen zum Data Mining</li> <li>• Datenbanken bei analytischen Informationssystemen</li> </ul>		
Lehr- und Lernformen	Vorlesung: Tafel, Präsentation, Beispiele und Diskussion, freiwillige Übungsaufgaben, auch in Gruppen Praktikum: selbstständiges Lösung der Praktikumsaufgaben		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• A. Bauer, H. Günzel (Hrsg.), <i>Data Warehouse Systeme</i>, dpunkt 2004</li> <li>• W. Lehner, <i>Datenbanktechnologie für Data-Warehouse-Systeme</i>, dpunkt 2002</li> </ul>		

Modulbezeichnung	<b>Certified Tester</b>	Kürzel	WP-CT WPP-CT
Lehrveranstaltung(en)	Vorlesung: Certified Tester Praktikum: Certified Tester	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Bettina Buth	SWS	3+1
Dozenten	Prof. Dr. Bettina Buth, Prof. Dr. Zhen Ru Dai	Sprache	deutsch
Voraussetzungen	PR1, PR2, SE1, (SE2)	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen den Stoff des Curriculums Certified Tester, Basislevel</li> <li>• beherrschen die Grundtechniken des statischen und dynamischen Testens in Theorie und Praxis</li> <li>• können Anforderungen systematisch in Tests umsetzen, Testergebnisse interpretieren</li> <li>• beherrschen programmiernahe Werkzeuge im Umfeld des Testens (Coverage-Analyse, modellbasierte Testfallerzeugung)</li> <li>• sind in der Lage das Zertifikat "Certified Tester Foundation Level" zu erwerben</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Elementare Begriffe des Testens</li> <li>• Grundkonzepte des Testens im Softwarelebenszyklus</li> <li>• Grundlegende Techniken des statischen Tests</li> <li>• Grundlegende Techniken des dynamischen Tests</li> <li>• Grundkonzepte des Testmanagement</li> <li>• Einführung in die Auswahl und Verwendung von Testwerkzeugen</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Overhead-/Rechnerpräsentationen, freiwillige Übungsaufgaben, Kleingruppenarbeit Praktikum: Beispielhafte Anwendung von Testtechniken in Teams von 3-4 Personen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• A. Spillner, T. Linz: Basiswissen Softwaretest – Aus- und Weiterbildung zum Certified Tester – Foundation Level</li> <li>• Material des ISTQB für Hochschullehrer</li> <li>• Online-Material des ISTQB</li> </ul>		

Modulbezeichnung	<b>Consulting</b>	Kürzel	WP-C WPP-C
Lehrveranstaltung(en)	Vorlesung: Consulting Übung: Consulting	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Übung, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Thomas Thiel-Clemen	SWS	2+2
Dozenten	Prof. Dr. Thomas Thiel-Clemen	Sprache	deutsch
Voraussetzungen	GW, BWL, SEA, IS	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen die Lösung komplexer, praxisrelevanter Fallstudien</li> <li>• können Kunden gewinnen und die Kundenbindung verstärken</li> <li>• können Unternehmensstrategien erarbeiten und präsentieren</li> <li>• können Workshops &amp; Visionsbildung moderieren</li> <li>• können einen Business Plan erstellen</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Arbeit mit Fallstudien</li> <li>• Kundenprofile erstellen</li> <li>• Lösungsportfolios definieren</li> <li>• Problemanalyse und Lösungsdesign</li> <li>• Account Management und Vertrieb</li> <li>• Organisationsentwicklung und Management-of-Change</li> <li>• (IT-)Unternehmensstrategien und Geschäftsmodelle</li> <li>• Business-to-IT-Alignment</li> <li>• Kennzahlen im Consulting</li> </ul>		
Lehr- und Lernformen	Vorlesung: Tafel, Präsentation, Fallstudien, freiwillige Übungsaufgaben, evtl. Tutorium Übung: selbstständiges Lösen der Aufgaben		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführte Übung Übung: erfolgreiche Bearbeitung aller Aufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• W. Ellet: <i>Das Fallstudien-Handbuch der Harvard Business School Press</i>, Haupt, 2008</li> <li>• H. Hungenberg: <i>Strategisches Management in Unternehmen – Ziele Prozesse Verfahren</i>, Gabler, 2008</li> </ul>		

Modulbezeichnung	<b>Das Softwareprojekt</b>	Kürzel	WP-SP WPP-SP
Lehrveranstaltung(en)	Vorlesung: Das Softwareprojekt Praktikum: Das Softwareprojekt	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Stefan Sarstedt	SWS	2+2
Dozenten	Prof. Dr. Stefan Sarstedt	Sprache	deutsch
Voraussetzungen	Software-Engineering 1 (SE1, SEP1), Datenbanken (DB, DBP), Programmieren I+II (PR1, PRP1, PR2, PRP2)	Häufigkeit	semesterweise zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• lernen Software-Engineering- und Architekturthemen anhand eines praxisnahen Beispiels kennen</li> <li>• können im Team ein System von der Analyse bis hin zur Auslieferung erstellen</li> <li>• können Qualitätssicherungsmaßnahmen an einem realitätsnahen Projekt anwenden</li> <li>• beherrschen ein aktuelles Vorgehensmodell sowie ausgewählte Entwicklungswerkzeuge</li> <li>• können einzelne Qualitätsaspekte der Applikation beurteilen</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Entwicklung eines realitätsnahen Informationssystems</li> <li>• Spezifikation</li> <li>• Architekturdesign und -bewertung</li> <li>• Aktuelles Vorgehensmodell</li> <li>• Umgang mit Legacy-Quellcode</li> <li>• Moderne Referenzarchitektur für Informationssysteme</li> <li>• Automatisierte Architekturprüfung und Codeanalyse-Techniken, weitere Qualitätssicherungsmaßnahmen</li> <li>• Aktuelle und leistungsfähige Entwicklungswerkzeuge</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Rechnerpräsentationen, freiwillige Übungsaufgaben, Referat/Hausarbeit, Gruppenarbeit Praktikum: Aufgabenbearbeitung in Gruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Grundlagenbücher: s. Bibliothek</li> <li>• Eric Evans: Domain-Driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley Longman, 2003.</li> <li>• Craig Larman: Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice Hall, 2004.</li> <li>• Robert C. Martin: Agile Software Development, Principles, Patterns, and Practices. Prentice Hall International, 2011.</li> <li>• Robert C. Martin: Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall International, 2008.</li> <li>• Johannes Siedersleben. Moderne Software-Architektur. Umsichtig planen, robust bauen mit Quasar. dpunkt, Heidelberg, 2004.</li> <li>• Gernot Starke. Effektive Software-Architekturen. Ein praktischer Leitfaden. Carl Hanser Verlag GmbH &amp; Co. KG, 2011.</li> <li>• eigene Skripte des Dozenten</li> </ul>		

Modulbezeichnung	<b>Datenbankdesign</b>	Kürzel	WP-DD WPP-DD
Lehrveranstaltung(en)	Vorlesung: Datenbankdesign Praktikum: Datenbankdesign	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Wolfgang Gerken	SWS	2+2
Dozenten	Prof. Dr. Wolfgang Gerken	Sprache	deutsch
Voraussetzungen	Datenbanken (AI) bzw. Informationssysteme I (WI)	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• können die Funktionsweise von Datenbankmanagementsystemen (DBMS) beschreiben und Konsequenzen für Datenbank-Anwendungsentwicklung ableiten</li> <li>• können Problemstellungen des Datenbankbetriebs und der Datenbankadministration fundiert bearbeiten</li> <li>• beherrschen den Prozess des Entwurfs von Datenbankanwendungen vom Datenbankentwurf bis zum Betrieb des datenbankgestützten Anwendungssystem</li> <li>• können aktuelle Entwicklungen im Bereich der DBMS bewerten und nutzen</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Architektur von Datenbankmanagementsystemen                         <ul style="list-style-type: none"> <li>- Schichtenarchitekturen von DBMS</li> <li>Interne Funktionsweise von Datenbanksystemen</li> <li>- Auswirkungen von Datenbankarchitekturen auf die Leistungsfähigkeit von Datenbankanwendungen</li> <li>- Transaktionen und Datenbankverwaltung</li> <li>- Transaktionskonzept inklusive Transaktionsmodelle verstehen</li> <li>- Protokolle zur Transaktionsausführung</li> <li>- Fragestellungen der Datenbankadministration (Datenbanksicherung, Systemeinstellungen und physische Optimierungen)</li> </ul> </li> <li>• Entwurf von Datenbankanwendungen                         <ul style="list-style-type: none"> <li>- Externes Datenbankdesign inklusive Sichtenentwurf und Rollenmodellierung</li> <li>- Konzeptuelles Datenbankdesign inklusive Realisierung von Anwendungsfunktionalität</li> <li>- Internes Datenbankdesign (Festlegung von Speicherungs- und Zugriffsstrukturen)</li> <li>- Umsetzung dieser Entwurfsaufgaben in einem exemplarischen kommerziellen DBMS</li> </ul> </li> <li>• Objektrelationale Datenbankmodelle</li> <li>• Aktuelle Entwicklungen im Bereich von Datenbanken</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Rechnerpräsentationen, freiwillige Übungsaufgaben, Referat/Hausarbeit, Gruppenarbeit Praktikum: Aufgabenbearbeitung in Zweiergruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Gunter Saake, Andreas Heuer und Kai-Uwe Sattler: Datenbanken: Konzepte und Sprachen, mitp 2011</li> <li>• Gunter Saake, Andreas Heuer und Kai-Uwe Sattler: Datenbanken: Implementierungstechniken. mitp 2011</li> <li>• eigene Vorlesungsunterlagen</li> </ul>		

Modulbezeichnung	<b>Enterprise Architekturen</b>	Kürzel	WP-EA WPP-EA
Lehrveranstaltung(en)	Vorlesung: Enterprise Architekturen Praktikum: Enterprise Architekturen	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 116 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Stefan Sarstedt	SWS	2+2
Dozenten	Prof. Dr. Stefan Sarstedt	Sprache	deutsch
Voraussetzungen	SEA1, SEA2	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• können ein großes Softwaresystem mit einer komplexen Fachlichkeit in Projektteams erstellen und weiterentwickeln.</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Komplexe, realistische fachliche Probleme aus einem bestimmten Anwendungsbereich, bspw. aus der Logistik</li> <li>• Komplexe nichtfunktionale Anforderungen (Performanz, Robustheit, etc.)</li> <li>• Organisation von Teamarbeit</li> <li>• Fortgeschrittene Architekturstile und Datenbanktechnologien</li> <li>• QS-Maßnahmen in großen Projekten</li> <li>• Testwerkzeuge</li> <li>• Prototyping</li> </ul>		
Lehr- und Lernformen	Vorlesung: Präsentation, Tafel, Demonstration am Rechner, Beispielaufgaben, freiwillige Übungsaufgaben, evtl. Tutorium Praktikum: Bearbeitung der Aufgaben in Teams, Begutachtung der Lösungen, Gesprächsführung		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Literaturhinweise werden je nach aktuellem Stand in der Vorlesung gegeben.</li> </ul>		

Modulbezeichnung	<b>Mensch-Maschine-Interaktion</b>	Kürzel	WP-MMI WPP-MMI
Lehrveranstaltung(en)	Vorlesung: Mensch-Maschine-Interaktion Praktikum: Mensch-Maschine-Interaktion	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 116 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Olaf Zukunft	SWS	3+1
Dozenten	Prof. Dr. Olaf Zukunft	Sprache	deutsch
Voraussetzungen	Programmieren 1 und 2, Algorithmen und Datenstrukturen, SEA1	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>verstehen die Relevanz einer ergonomisch gut gestalteten Benutzungsschnittstelle für die Qualität des Produktes</li> <li>erkennen den Einfluss der Gestaltung von Software auf die damit zu bearbeitenden Aufgaben</li> <li>kennen Anwendungsgebiete der Software-Ergonomie</li> <li>kennen den Nutzen eines benutzerzentrierten Entwicklungsprozesses</li> <li>verstehen die Software-Ergonomie als notwendigen Teil des Software-Entwicklungsprozesses und damit als Teil von Software-Qualität</li> <li>verstehen die Interdisziplinarität als notwendigen Aspekt der Entwicklung von Software</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>Ebenen der Mensch-Computer-Interaktion</li> <li>Zusammenhang der Gestaltung von Software mit der Gestaltung von Arbeit und Arbeitsabläufen</li> <li>Einsatzbereiche, zum Beispiel Anwendungs-Software, Websites und Embedded Systems</li> <li>Rollen von Beteiligten in der Software-Entwicklung, zum Beispiel Entwickler, Benutzer, Verantwortliche, Interessensvertreter</li> <li>Angebote der Software-Ergonomie zur Optimierung der Benutzungsschnittstelle Agentensysteme im Umfeld von Ubiquitous Computing</li> </ul>		
Lehr- und Lernformen	Vorlesung: Präsentation, Tafel, Demonstration am Rechner, Beispielaufgaben Praktikum: Bearbeitung der Aufgaben in Teams		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>GI-Fachgruppe Software-Ergonomie: Curriculum für ein Basismodul zur Mensch-Computer-Interaktion. 2006.</li> </ul>		

Modulbezeichnung	<b>Multi-Agenten-Simulation</b>	Kürzel	WP-MAS WPP-MAS
Lehrveranstaltung(en)	Vorlesung: Multi-Agenten-Simulation Praktikum: Multi-Agenten-Simulation	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 116 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Thomas Thiel-Clemen	SWS	2+2
Dozenten	Prof. Dr. Thomas Thiel-Clemen	Sprache	deutsch
Voraussetzungen	gute Programmierkenntnisse, Grundlagen Künstlicher Intelligenz und Verteilter Systeme	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• erhalten einen Einblick in interdisziplinäre Arbeitsweisen</li> <li>• kennen die einzelnen Phasen des Modellbildungs- und Simulationsprozesses und können diese auch umsetzen</li> <li>• können sich informatikfremde Fachthemen erarbeiten und Methoden der Informatik darauf anwenden</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Grundlagen und Anwendungsbereiche der Agenten-basierten Simulation</li> <li>• das Domänenmodell</li> <li>• Implementierungsaspekte</li> <li>• Visualisierung und Datenauswertung</li> <li>• Verifikation und Validierung</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Overhead-/Rechnerpräsentationen, freiwillige Übungsaufgaben, Kleingruppenarbeit Praktikum: Aufgabenbearbeitung in Kleingruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• S.F. Railsback, V. Grimm, Agent-Based and Individual-Based Modeling – A practical introduction, Princeton University Press 2012</li> <li>• H.-J. Bungartz, S. Zimmer, M. Buchholz, D. Pflüger, Modellbildung und Simulation - Eine anwendungsorientierte Einführung, Springer, 2009</li> <li>• A.M. Uhrmacher, D. Weyns (Eds.), Multi-Agent Systems - Simulation and Applications, CRC Press, 2009</li> <li>• Eigene Skripte der Dozenten</li> </ul>		

Modulbezeichnung	<b>Quantenrechner und Quantenkryptographie</b>	Kürzel	WP-QRK WPP-QRK
Lehrveranstaltung(en)	Vorlesung: Quantenrechner und Quantenkryptographie Praktikum: Quantenrechner und Quantenkryptographie	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Thomas Canzler	SWS	3+1
Dozenten	Prof. Dr. Thomas Canzler	Sprache	deutsch
Voraussetzungen	Vektor- und Matrizenrechnung	Häufigkeit	ca. jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• verstehen die für Quantenrechner wesentlichen Phänomene der Quantenmechanik</li> <li>• verstehen den Aufbau von Quantenrechnern und den Unterschied zu konventionellen Rechnern</li> <li>• können die Matrixform für Operationen herleiten, entsprechende Rechnungen durchführen und verstehen die Darstellung im Simulator</li> <li>• verstehen und simulieren Quantenalgorithmen</li> <li>• verstehen das Prinzip der Quantenkryptographie, kennen den Unterschied zur klassischen Kryptographie und simulieren entsprechende Protokolle</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Grundlagen (Superposition, Verschränkung, No Cloning Theorem, Dekohärenz)</li> <li>• Quantenrechnerstruktur (Reversibilität, QBits, QRegister, Quantengatter)</li> <li>• Vektor- und Matrixdarstellung von Quantenoperationen</li> <li>• Quantenalgorithmen (Deutsch-Alg., Shor-Alg., Teleportation)</li> <li>• Quantenkryptographie (Schlüsselerzeugung und –verteilung, BB84-Protokoll)</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Präsentationen und Erläuterungen am Tablet PC, Simulation von Beispielen (Vorführung sowie Studierende auf ihren Laptops) Praktikum: Übungen sowie Simulationen mit Simuquant (s.u.)		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Matthias Homeister: Quantum Computing verstehen; Vieweg: Wiesbaden 2013(3)</li> <li>• Dagmar Bruß: Quanteninformatik; Fischer: Frankfurt a.M. 2003</li> <li>• Simuquant: ein Simulator für Quantenschaltungen; <a href="http://sourceforge.net/projects/simuquant/">http://sourceforge.net/projects/simuquant/</a></li> </ul>		

Modulbezeichnung	<b>Rekonfigurierbare Petrinetze</b>	Kürzel	WP-recPN WPP-recPN
Lehrveranstaltung(en)	Vorlesung: Rekonfigurierbare Petrinetze Praktikum: Werkzeugentwicklung und Modellierung	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	24 Std. Vorlesung, 24 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Julia Padberg	SWS	2+2
Dozenten	Prof. Dr. Julia Padberg	Sprache	deutsch, englisch
Voraussetzungen	Graphtransformation, Petrinetze, Programmierkenntnisse in Java	Häufigkeit	jährlich zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen Modellierung nebenläufiger Systeme, die sich selbst dynamisch verändern</li> <li>• kennen Konzepte und Algorithmen rekonfigurierbarer Petrinetze</li> <li>• besitzen Kenntnisse über Layoutalgorithmen</li> <li>• können sich eigenständig anhand wissenschaftlicher Publikationen in neue Themen einarbeiten</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Elementare Begriffe, wie Petrinetze, Graphtransformation</li> <li>• Grundlagen der formalener Ersetzungssysteme</li> <li>• Grundkonzepte der rekonfigurierbare Petrinetze</li> <li>• Einführung in Layoutalgorithmen</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Präsentationen, Übungsaufgaben, studentische Referate, Gesprächs-/ Diskussionsführung, Gruppenarbeit Praktikum: Programmieren und/oder Modellieren in Zweiergruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Fundamentals of Algebraic Graph Transformation, Ehrig, H. et al, 2006, Springer-Verlag Berlin and Heidelberg</li> <li>• Petrinetze: Modellierungstechnik, Analysemethoden, Fallstudien, Reisig, W., 2010, Vieweg+Teubner Verlag</li> <li>• S.O. Krumke, H. Noltemeier: Graphentheoretische Konzepte und Algorithmen, 2009, Teubner</li> <li>• eigene Skripte der Dozenten</li> </ul>		

Modulbezeichnung	<b>Robot Vision</b>	Kürzel	WP-RV WPP-RV
Lehrveranstaltung(en)	Vorlesung: Robot Vision Praktikum: Robot Vision	Semester	AI: 4-6 TI/WI: 5-6
Arbeitsaufwand	36 Std. Vorlesung, 12 Std. Praktikum, 132 Std. Eigenarbeit/Selbststudium	CP	6
Modulverantwortliche(r)	Prof. Dr. Andreas Meisel	SWS	3+1
Dozenten	Prof. Dr. Andreas Meisel	Sprache	deutsch
Voraussetzungen	Modul Mathematische Grundlagen, lineare Algebra	Häufigkeit	semesterweise zur Wahl
Lernziele und Kompetenzen	Die Studierenden <ul style="list-style-type: none"> <li>• verstehen die Grundstruktur von Bildverarbeitungsanwendungen</li> <li>• können Bildeigenschaften beurteilen und gezielt verändern</li> <li>• verstehen die Grundlagen der Bildfilterung im Orts- und Frequenzbereich und können eigene Filter realisieren oder an eigene Problemstellungen anpassen</li> <li>• können Verfahren zur Merkmalsextraktion realisieren bzw. an eigene Problemstellungen anpassen</li> <li>• verstehen die Grundlagen der Mustererkennung und können einfache Klassifikatoren entwickeln</li> <li>• können sich eigenständig in neue Themen der Bildverarbeitung einarbeiten</li> </ul>		
Inhalte	<ul style="list-style-type: none"> <li>• Verfahren zur Bildverbesserung und -aufbereitung, wie z.B. Histogrammausgleich</li> <li>• automat. Schwellwertverfahren, geometrische Bildtransformationen (affine Transformation, Bildverzerrung)</li> <li>• logische und arithmetische Bildoperationen, Rangordnungsverfahren</li> <li>• Faltungsbasierte Filteroperationen, wie Laplace-Operator, Sobel-Operator, LoG-Operator, Canny-Operator</li> <li>• anwendungsorientierte Einführung in die Diskrete Fourier Transform. sowie die Filterung im Frequenzbereich</li> <li>• ausgewählte Verfahren der Merkmalsextraktion, z.B. Konturextraktion, Connected-component-Labeling, Hough-Transformation, Hu-Momente</li> <li>• Einführung in Klassifikationsverfahren, wie Nearest-Neighbour, radiale Basisfunktionen, Multilayer-Perzeptron und Kohonen-Netz</li> </ul>		
Lehr- und Lernformen	Vorlesung: Seminaristischer Unterricht, Tafelarbeit, Overhead-/Rechnerpräsentationen, freiwillige Übungsaufgaben, Applets und Beispielprogramme zur Demonstration Praktikum: Programmieren in Zweiergruppen		
Studien- und Prüfungsleistungen	Vorlesung: nach Festlegung als benotete Klausur, benotete mündliche Prüfung oder benotetes Referat Prüfungsvorleistung (PVL): erfolgreich durchgeführtes Praktikum Praktikum: erfolgreiche Bearbeitung der Praktikumsaufgaben (PVL)		
Literatur	<ul style="list-style-type: none"> <li>• Gonzalez und Woods: Digital Image Processing, Pearson Education</li> <li>• Bernd Jähne: Digitale Bildverarbeitung, Springer-Verlag</li> <li>• Andreas Zell: Simulation Neuronaler Netze, Oldenbourg Verlag</li> <li>• Simon Haykin: Neural Networks and Learning Machines, Pearson Education</li> <li>• Duda, Hart, Stork: Pattern Classification, Wiley Interscience Publication</li> <li>• eigene Skripte und Folien der Dozenten</li> </ul>		