

Hochschule Karlsruhe

Fakultät für Informatik und Wirtschaftsinformatik

Modulhandbuch

Studiengang Informatik (Master), SPO 8

Sommersemester 2025

Module Informatik (Master), SPO 8

Interaction Design	3
Maschinelles Lernen	5
Software-Architekturen	8
Smart Interaction	12
Data Science	15
Konzepte von Programmiersprachen	18
Wissenschaftstheorie und Ethik	22
Managementkompetenz	25
Projektbasiertes wissenschaftliches Arbeiten unter Anleitung 1	28
Game Design	29
Künstliche Intelligenz	37
Theorie effizienter Algorithmen	40
Spezielle Kapitel Medieninformatik	43
Spezielle Kapitel KI	45
Spezielle Kapitel Software-Engineering	48
Mobile und Verteilte Systeme	51
Projektbasiertes wissenschaftliches Arbeiten unter Anleitung 2	55
Hauptseminar	56
Abschlussarbeit mit Kolloquium	57

Modul Interaction Design	
EDV-Bezeichnung	INFM110MI
Verantwortlich	Prof. Thomas Hinz
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Die Studierenden erlernen die theoretischen Kenntnisse der Konzeption und Gestaltung von interaktiven Systemen und setzen sie bei der Entwicklung von Prototypen praktisch um.</p> <p>Die Studierenden sind kompetent, Designprinzipien und Designkonzepte zur Lösung von Gestaltungsproblemen bei der Entwicklung dieser Systeme anzuwenden. Sie beherrschen innovative Benutzungsschnittstellen und sind in der Lage, neue Formen der Interaktion zwischen Mensch und Maschine zu konzipieren und zu gestalten.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Interaction Design	
EDV-Bezeichnung	INFM111MI
Dozent/in	Prof. Thomas Hinz
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>Die Studierenden erwerben die Fähigkeit, zukunftsweisende interaktive Systeme unter Berücksichtigung gesellschaftlicher, technologischer und sozialer Aspekte zu konzipieren und zu gestalten. Sie können Methoden des Design Futuring zur systematischen Entwicklung von Zukunftsszenarien anwenden, Komplexe soziotechnische Systeme und deren Wechselwirkungen analysieren, Partizipative Designprozesse zur Entwicklung nachhaltiger Zukunftslösungen gestalten, Prototypen und erlebbare Zukunftsszenarien entwickeln, sowie die gesellschaftlichen Implikationen technologischer Innovationen kritisch reflektieren.</p>

Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungsskript - B. Groß, E. Mandir: Zukünfte gestalten: Spekulation. Kritik. Innovation. Mit »Design Futuring« Zukunftsszenarien strategisch erkunden, entwerfen und verhandeln.. 2022, ISBN-13: 978-3874399586 - M. Leube: The Future Designer: Anthropology Meets Innovation in Search of Sustainable Design (Routledge Earthscan), 2024, ISBN-13 : 978-1032735535 - A. Dunne, F. Raby: Speculative Everything: Design, Fiction, and Social Dreaming, 2013, ISBN-13:978-0262019842 - T. Pyczak: Tell me!: Wie Sie mit Storytelling überzeugen. Zahlreiche Praxisbeispiele für alle, die erfolgreich sein wollen in Beruf, PR und Marketing., 2023, ISBN-13:978-3836291545 - R. Klanten, L. Feireiss, "A Touch of Code: Interactive Installations and Experiences", 2011, ISBN-13 978-3899553314 - J. Sauter, S. Jaschko, J. Ängeslevä, ART+COM: "Medien, Räume und Installationen", 2011, ISBN-13 978-3899553864 - H.-D. Hellige, "Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung", 2008, ISBN-13 978-3899425642 - J. Schenk, G. Rigole, "Mensch-Maschine-Kommunikation: Grundlagen von sprach- und bildbasierten Benutzerschnittstellen", 2010, ISBN-13 978-3642054563 - R. Dorau, "Emotionales Interaktionsdesign Gesten und Mimik interaktiver Systeme", 2011, ISBN-13 978-3642031007
Prüfungsleistungen	Mündl. Prüfung/Entwurf 20/1 Min./Semester (benotet)
Anmerkungen	Vorlesung, Teilnahme am seminaristischen Unterricht.
Lehrveranstaltung Interaction Design Übung	
EDV-Bezeichnung	INFM112MI
Dozent/in	Prof. Thomas Hinz
Umfang	4.0 ECTS-Punkte, 4.0 SWS 120 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	Die TeilnehmerInnen setzen die theoretischen Kenntnisse der Konzeption und Gestaltung interaktiver Systeme praktisch um. Sie designen und entwickeln Prototypen für interaktive Systeme. Die Studierenden experimentieren mit innovativen Formen der Mensch-Maschine-Schnittstelle und den Möglichkeiten der Interaktion im öffentlich genutzten Raum.
Empfohlene Literatur	- Vorlesungsskripte
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	<ul style="list-style-type: none"> - Teamarbeit - Experimente zu Design und Prototyping - Präsentation und Diskussion der Ergebnisse

Modul Maschinelles Lernen	
EDV-Bezeichnung	INFM110ML
Verantwortlich	Prof. Dr. Dennis Janka
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>In diesem Modul werden Methoden des Maschinellen Lernens behandelt. Studierende erlernen, sich Methoden des Maschinellen Lernens anzueignen und bzgl. ihrer Eignung für konkrete Aufgabenstellungen zu evaluieren.</p> <p>Die vermittelten Inhalte und Kompetenzen des Moduls werden im Modul Künstliche Intelligenz als bekannt bzw. vorhanden vorausgesetzt.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Maschinelles Lernen	
EDV-Bezeichnung	INFM111ML
Dozenten	Prof. Dr. Dennis Janka Prof. Dr. Christine Preisach
Umfang	4.0 ECTS-Punkte, 4.0 SWS 120 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

Inhalt	<p>Lehrmethode der Vorlesung: Seminaristischer Unterricht.</p> <p>Inhalte Vorlesungsteil Prof. Janka Einführung in Konzepte des überwachten Lernens anhand der linearen und logistischen Regression:</p> <ul style="list-style-type: none"> - Regressions- und Klassifizierungsprobleme - Modellfunktionen - Verlustfunktionen und Training von Modellen - Evaluation und Verbesserung von Modellen - Regularisierung <p>Aufbauend: Einführung in das Thema Deep Learning</p> <ul style="list-style-type: none"> - Künstliche Neuronale Netze - Training von neuronalen Netzen (Stochastischer Gradientenabstieg und Backpropagation) - Spezielle Typen von neuronalen Netzen, z.B.: Faltungsnetze, Autoencoder, generative Modelle. <p>Inhalte Vorlesungsteil Prof. Preisach Vorbereitung:</p> <ul style="list-style-type: none"> - Einordnung der Begriffe Künstliche Intelligenz, Maschinelles Lernen, Mustererkennung, Data Mining, Big Data etc. - Distanzen, Metriken und Ähnlichkeiten (Minkowski, Cosinus-Ähnlichkeit, Mahalanobis, Dynamic Time Warping etc.) - Einführung in die Datenvorverarbeitung, eine ausführliche Behandlung des Themas findet im Modul Data Science statt. <p>Transformationen</p> <ul style="list-style-type: none"> - PCA, nichtlineare PCA, in diesen Zusammenhang: Kernel-Trick und Support Vektor Maschinen - SVD - Matrixfaktorisierung, z.B. Latente Faktoren, Faktorisierung für Recommender Engines <p>Methoden des unüberwachten Lernens, eine Auswahl aus</p> <ul style="list-style-type: none"> - Cluster-Algorithmen: K-Means, PAM, CLARA, CLARANS, DBSCAN, C-Means, EM, SOM - Assoziationsanalyse, z.B. Apriori, FP-Growth - Graphbasierte Ansätze zur Bildung von Clustern, z.B. Normalized Cut - Ranking Algorithmen, z.B. Google Page Rank
Empfohlene Literatur	<ul style="list-style-type: none"> - Folien und Tafelanschrieb - Bernhard Schölkopf ; Alexander J. Smola, "Learning with Kernels : support vector machines, regularization, optimization, and beyond", Cambridge, Mass. [u.a.], MIT Press, 2002 - Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning". MIT Press, 2016.
Prüfungsleistungen	Klausur/mündl. Prüfung 120/20 Min. (benotet)
Anmerkungen	
Lehrveranstaltung Maschinelles Lernen Übung	
EDV-Bezeichnung	INFM112ML

Dozenten	Prof. Dr. Dennis Janka Prof. Dr. Christine Preisach
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	<p>In der Übung werden die Inhalte beider Vorlesungen „Maschinelles Lernen“ (Prof. Preisach und Prof. Janka) durch praktische Übungen vertieft, mit dem Ziel grundlegende Machine Learning Algorithmen selbstständig implementieren zu können. Die Aufgaben werden in Form von Python Jupyter Notebooks gestellt und bearbeitet.</p> <p>Inhalte der Übung:</p> <ul style="list-style-type: none"> - Grundlagen der Python-Programmierung im Umfeld maschinelles Lernen - Explorative Datenanalyse, Regression, Cross-Validation - Dimensionsreduktion, Hauptkomponentenanalyse - Empfehlungsdienste, kollaboratives Filtern, Ähnlichkeitsmaße - Implementierung simpler Neuronaler Netze und Backpropagation, teils mit PyTorch - Clustering mittels des K-Means Algorithmus, Distanzmaße
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (benotet)
Anmerkungen	Die Übung wird wahlweise auf dem eigenen Laptop oder auf PCs des Labors für Maschinelles Lernen durchgeführt.

Modul Software-Architekturen	
EDV-Bezeichnung	INFM110SE
Verantwortlich	Prof. Dr. Thomas Fuchß
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, die Architektur großer Software-Systeme selbstständig zu entwerfen, zu gestalten und kritisch zu bewerten. Sie können Software in Komponenten und Konnektoren zerlegen, geeignete Frameworks auswählen, deren Einsatz bewerten und Funktionalitäten sowie Abläufe logisch organisieren. Dabei planen und gestalten sie auch parallele und nebenläufige Abläufe und integrieren diese in Architekturen für verschiedene Anwendungsfelder. Darüber hinaus haben die Studierenden ihre technischen, sozialen und persönlichen Kompetenzen sowie ihre Kommunikationsfähigkeit und ihr Selbstmanagement gezielt weiterentwickelt und gelernt, diese Fähigkeiten im Kontext großer Softwareprojekte effektiv einzusetzen.
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Software-Architekturen	
EDV-Bezeichnung	INFM111SE
Dozent/in	Prof. Dr. Thomas Fuchß
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

<p>Inhalt</p>	<p>Die Lehrveranstaltung "Software-Architekturen" vermittelt vertiefte Kenntnisse und Fähigkeiten zur Entwicklung und Analyse moderner Software-Architekturen.</p> <p>Im ersten Teil der Vorlesung werden relevante Prozessmodelle wiederholt und darauf aufbauend Ansätze für eine agile Architekturentwicklung erarbeitet. Studierende lernen verschiedene View-Modelle kennen, vergleichen diese und vertiefen ihre Kenntnisse in Modellierungstechniken für Komponenten und Konnektoren. Bereits erlernte Entwurfsmethoden werden erweitert, insbesondere im Hinblick auf die Nutzung von Szenarien zur Beschreibung und Bewertung nicht-funktionaler Anforderungen.</p> <p>Im zweiten Teil der Vorlesung stehen elementare Architekturmuster im Fokus, darunter Schichten-Architektur, hexagonale Architektur, Onion-Architektur, Blackboard, Pipes-and-Filters und Event-Driven Architecture. Studierende erlernen die Strukturen, Eigenschaften und Anwendungsmöglichkeiten dieser Muster und verstehen ihre Rolle in typischen Middleware-Konzepten. Die Vorlesung zeigt, wie solche Architekturvorlagen Ansatzpunkte für die Organisation flexibler und evolutionärer Systeme bieten.</p> <p>Der dritte Teil der Vorlesung widmet sich der Architektur auf Modulebene. Anhand praxisnaher Szenarien und Beispiele wird der Einsatz von Patterns für den Business Layer thematisiert und im jeweiligen Kontext analysiert. Ziel ist es, den Studierenden ein fundiertes Verständnis für die Gestaltung und Bewertung modularer Software-Architekturen zu vermitteln.</p>
<p>Empfohlene Literatur</p>	<ul style="list-style-type: none"> - Avgeriou, P; et. al (editors): Relating Software Requirements and Architectures. Springer, 2011. - Clements, P.; Bass, L. and Kazman, R.: Software Architecture in Practice, 2. ed. Addison-Wesley, 2003. - Fowler, M.: Patterns of Enterprise Application Architecture. Addison-Wesley, 2003. - Goll, J. und Dausmann, M.: Architektur- und Entwurfsmuster der Softwaretechnik. Springer Vieweg, 2013. - Gorton, Ian: Essential Software Architecture, 2. ed. Springer, 2011. - Larman, Craig: Applying UML and Patterns : An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3. ed. Prentice Hall, 2004. - Lilienthal, Carola: Sustainable software architecture: analyze and reduce technical debt. dpunkt.verlag, 2019. - Buschmann, Frank: A System of Patterns (POSA V.1). John Wiley & Sons. 1996 - Schmidt, Douglas C.: Patterns for Concurrent and Networked Objects (POSA V.2). John Wiley & Sons, 2000. - Sommerville, Ian: Software Engineering, 9. Auflage. Pearson Studium, 2012. - Vogel, O.; Arnold, I.; Chughtai, A. and Kehrer, T.: Software Architecture: A Comprehensive Framework and Guide for Practitioners. Springer, 2011. - Vogel, O.; et. al: Software-Architektur: Grundlagen – Konzepte – Praxis, 2. Auflage. Spektrum, 2009.
<p>Prüfungsleistungen</p>	<p>Klausur/mündl. Prüfung 60/20 Min. (benotet)</p>

Anmerkungen	Seminaristischer Unterricht, Übungsblätter und Präsentationen
Lehrveranstaltung Parallele Programmierung Labor	
EDV-Bezeichnung	INFM112SE
Dozent/in	Dipl.-Ing. Christian Meder
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Labor
Lehrsprache	deutsch
Inhalt	<p>Das "Parallele Programmierung Labor" vermittelt praxisorientierte Kenntnisse zur nebenläufigen Programmierung und deren Anwendung in verschiedenen modernen Technologien und Systemen.</p> <p>Im ersten Teil der Veranstaltung werden die Grundlagen der nebenläufigen Programmierung wiederholt, vertieft und durch praktische Übungen mit der Programmiersprache Java und dem Java Development Kit (JDK) angewendet. Dabei erlernen die Studierenden, wie sie nebenläufige Prozesse effizient gestalten und typische Herausforderungen wie Race Conditions oder Deadlocks bewältigen können.</p> <p>Im zweiten Teil der Veranstaltung wird das Wissen durch kompakte Einführungen in aktuelle Anwendungsfelder der nebenläufigen Programmierung erweitert. Dabei werden verschiedene Technologien und Plattformen beleuchtet, darunter:</p> <ul style="list-style-type: none"> - Nebenläufigkeit in Betriebssystemen, z. B. unter Linux - Nebenläufigkeit auf mobilen Plattformen, z. B. Android - Map/Reduce und das Hadoop-Ökosystem - Nebenläufigkeit in Webanwendungen - Nebenläufigkeit in modernen Programmiersprachen wie Go und Rust - Reactive Extensions (Rx) und deren Einsatzgebiete - Cloud-, Cluster- und verteilte Systeme - Microservice-Architekturen - Deep Learning und Nebenläufigkeit <p>Durch die Kombination aus theoretischer Einführung und praktischer Anwendung werden die Studierenden befähigt, nebenläufige Programmierung in unterschiedlichen Kontexten zu verstehen, anzuwenden und weiterzuentwickeln.</p>
Empfohlene Literatur	<ul style="list-style-type: none"> - Brian Goetz, "Java Concurrency in Practice" - Online-Tutorials <p>Die Literatur des zweiten Teils wird in jedem Semester neu zusammengestellt.</p>
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	Die erfolgreiche Teilnahme am Labor wird durch die eigenständige Bearbeitung der Übungen dokumentiert und begutachtet. Im zweiten Teil des Labors werden diverse aktuelle Quellen und Präsentationen vorgestellt und diskutiert.
Lehrveranstaltung Software-Architekturen Labor	
EDV-Bezeichnung	INFM113SE
Dozent/in	Prof. Dr. Carsten Sinz

Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Labor
Lehrsprache	deutsch
Inhalt	In diesem Labor werden die Methoden und Kenntnisse der Vorlesung Software-Architekturen praktisch erprobt und umgesetzt. Dabei werden anhand eines größeren Software-Projekts der gesamte Entwicklungsprozess vom Entwurf über Realisierung bis hin zu Test und Qualitätssicherung im Team bearbeitet. Dabei sollen Entwurfstechniken und Architekturmuster der Vorlesung eingesetzt sowie Software-Entwicklungs-Frameworks bewertet und ausgewählt werden. Die Implementierung in einer vom Team gewählten Programmiersprache und deren Qualitätssicherung ist ebenfalls Teil des Projekts und erlaubt auch eine kritische Bewertung der eingesetzten Techniken. Schließlich soll durch die Erstellung eines Entwickler- und Benutzer-"Handbuchs" die Dokumentation größerer Projekte erlernt werden.
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	

Modul Smart Interaction	
EDV-Bezeichnung	INFM120MI
Verantwortlich	Prof. Dr. Matthias Wölfel
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Lernergebnissen</p> <p>Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden in der Lage, die Grundlagen der Sprach- und Gestenerkennung sowie die Erfassung und Erkennung von Umgebungsobjekten zu verstehen. Sie können die Möglichkeiten und Grenzen der Nutzung solcher Technologien in der Interaktion und Informationspräsentation erfassen und deren Anwendung in verschiedenen Bereichen wie Smartphones, interaktiven Räumen, Spielekonsolen oder der Mensch-Roboter-Kooperation analysieren. Die Studierenden können komplexe wahrnehmungsbasierte Interaktionssysteme konzipieren, implementieren und evaluieren, um die Interaktion zwischen Menschen und Maschine zu verbessern. Weiterhin sind sie in der Lage, innovative Lösungen zu entwickeln, indem sie die Funktionsweise und Anwendung moderner Sensorik, wie beispielsweise dem Kinect-Sensor, in die Systemgestaltung integrieren. Darüber hinaus können die Studierenden aktuelle Entwicklungen und Anwendungsfelder im Bereich der smarten Interaktion diskutieren und die theoretischen Hintergründe der zugrunde liegenden Erkennungstechnologien anwenden. Sie verstehen die Prinzipien der Nutzung von Sensorik und Interaktionsmodellen, um Lösungen zu entwickeln, die sowohl technisch umsetzbar als auch praxisrelevant sind.</p> <p>Lehr-/ Lernmethode</p> <p>Die Veranstaltung nutzt verschiedene Lehrmethoden und Ressourcen, um den Studierenden ein tiefgreifendes Verständnis für die Themenbereiche zu vermitteln. Dazu gehören Vorlesungen, praktische Übungen, Anwendungsbeispiele mit verschiedener Sensorik (z.B. dem Kinect Sensor) sowie Diskussionen über aktuelle Entwicklungen und Anwendungen im Bereich der smarten Interaktion.</p> <p>Studien- und Prüfungsleistungen</p> <p>Die Modulprüfung besteht aus a) einer mündlichen Prüfung, in der die Studierenden unterschiedliche Theorien und praktische Beispiele der Smart Interaction ohne Hilfsmittel abrufen und erklären sollen und b) einem praktischen Teil, in der die Studierenden nachweisen müssen, dass sie die Theorien in einer eigenen Anwendung realisieren können.</p>

Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Smart Interaction	
EDV-Bezeichnung	INFM121MI
Dozent/in	Prof. Dr. Matthias Wölfel
Umfang	3.0 ECTS-Punkte, 3.0 SWS 90 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>Lernergebnissen</p> <p>Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden in der Lage, die Grundlagen der Sprach- und Gestenerkennung sowie die Erfassung und Erkennung von Umgebungsobjekten zu verstehen. Sie können die Möglichkeiten und Grenzen der Nutzung solcher Technologien in der Interaktion und Informationspräsentation erfassen und deren Anwendung in verschiedenen Bereichen wie Smartphones, interaktiven Räumen, Spielekonsolen oder der Mensch-Roboter-Kooperation analysieren. Die Studierenden können komplexe wahrnehmungsbasierte Interaktionssysteme konzipieren, implementieren und evaluieren, um die Interaktion zwischen Menschen und Maschine zu verbessern. Weiterhin sind sie in der Lage, innovative Lösungen zu entwickeln, indem sie die Funktionsweise und Anwendung moderner Sensorik, wie beispielsweise dem Kinect-Sensor, in die Systemgestaltung integrieren. Darüber hinaus können die Studierenden aktuelle Entwicklungen und Anwendungsfelder im Bereich der smarten Interaktion diskutieren und die theoretischen Hintergründe der zugrunde liegenden Erkennungstechnologien anwenden. Sie verstehen die Prinzipien der Nutzung von Sensorik und Interaktionsmodellen, um Lösungen zu entwickeln, die sowohl technisch umsetzbar als auch praxisrelevant sind.</p> <p>Lehr-/ Lernmethode</p> <p>Die Veranstaltung nutzt verschiedene Lehrmethoden und Ressourcen, um den Studierenden ein tiefgreifendes Verständnis für die Themenbereiche zu vermitteln. Dazu gehören Vorlesungen, praktische Übungen, Anwendungsbeispiele mit verschiedener Sensorik (z.B. dem Kinect Sensor) sowie Diskussionen über aktuelle Entwicklungen und Anwendungen im Bereich der smarten Interaktion.</p> <p>Studien- und Prüfungsleistungen</p> <p>Die Modulprüfung besteht aus a) einer mündlichen Prüfung, in der die Studierenden unterschiedliche Theorien und praktische Beispiel der Smart Interaction ohne Hilfsmittel abrufen und erklären sollen und b) einem praktischen Teil, in der die Studierenden nachweisen müssen, das sie die Theorien in einer eigenen Anwendung realisieren können.</p>

Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungsfolien - Lawrence R. Rabiner, Biing-Hwang Juang Juang, "Fundamentals of Speech Recognition", 1993, ISBN 0-13-015157-2 - Nitin Indurkha (Editor), Fred J. Demerau (Editor), "Handbook of Natural Language Processing", Second Edition, Chapman & Hall/CRC Machine Learning & Pattern Recognition, February 22, 2010, ISBN-10 1420085921, ISBN-13 978-1420085921 - Roberto Cipolla (Editor), Alex Pentland (Editor), "Computer Vision for Human-Machine Interaction", Massachusetts Institute of Technology Online Publication, Cambridge Books Online Date: July 2010, Online ISBN 9780511569937 - Marcus Tonnis, "Augmented Reality: Einblicke in die Erweiterte Realität" (Informatik im Fokus), Springer 2010 - Alan B. Craig, "Understanding Augmented Reality - Concepts and Applications", Morgan Kaufman 2013 ISBN 978-0-240-82408-6
Prüfungsleistungen	Mündl. Prüfung/Entwurf 20/1 Min./Semester (benotet)
Anmerkungen	Vorlesung, Teilnahme am seminaristischen Unterricht.
Lehrveranstaltung Smart Interaction Übung	
EDV-Bezeichnung	INFM122MI
Dozent/in	Prof. Dr. Matthias Wölfel
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	Die Vorlesungsinhalte werden anhand der Konzeption und der Umsetzung eines im Rahmen der Vorlesung gemeinsam definierten Aufgabe vertieft.
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	

Modul Data Science	
EDV-Bezeichnung	INFM120ML
Verantwortlich	Prof. Dr. Reimar Hofmann
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Am Ende dieses Moduls haben die Studierenden theoretische und mathematische Grundlagen des Maschinellen Lernens und der Datenanalyse verstanden. Sie können dadurch die Eignung verschiedener Verfahren für konkrete Situationen beurteilen, und Phänomene, die sie beobachten, schlüssig interpretieren, sowie daraus bei Bedarf Verbesserungsideen für die gewählten Ansätze ableiten.</p> <p>Die vermittelten Inhalte und Kompetenzen des Moduls sind für die Teilnahme am Modul Künstliche Intelligenz von Vorteil.</p>
Prüfungsleistungen	Klausur/mündl. Prüfung 120/20 Min. (benotet)
Lehrveranstaltung Data Science	
EDV-Bezeichnung	INFM121ML.a
Dozent/in	Prof. Dr. Reimar Hofmann
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<ul style="list-style-type: none"> - Skalentypen (nominal, ordinal, intervall, ratio), Konvertierung von Skalentypen (one-hot-coding etc.), Normierung - Explorative Datenanalyse, Unterscheidung direkter von indirekten Zusammenhängen, Datenvisualisierung - Statistische Grundlagen maschinellen Lernens, Maximum Likelihood-Ansatz, Bias und Varianz (Overfitting) als Fehlerquellen von Lernergebnissen. - Kostenfunktionen für numerische Regression und Klassifikation - Kriterien für Datenqualität, Umgang mit Qualitätsmängeln (z.B. fehlende Werte, Ausreißer) - Umgang mit komplexeren Datentypen (record-Daten, heterogene Daten, bag of words), Datentransformationen, Feature Engineering - Dimensionsreduktion: heuristisch, manuell etc. - Umfangsreduktion: Sampling etc.
Empfohlene Literatur	
Prüfungsleistungen	Modulprüfung
Anmerkungen	
Lehrveranstaltung Optimierung	

EDV-Bezeichnung	INFM121ML.b
Dozent/in	Prof. Dr.-Ing. Astrid Laubenheimer
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>Ziel: Vermittlung von theoretischen und praktischen Kenntnissen über Optimierungsmethoden/-kalküle im Kontext des Maschinellen Lernens (Minimierung der Fehlerfunktion)</p> <p>Inhalte:</p> <p>Einführung, Motivation und Modellierung</p> <p>(Wiederholung) Mehrdimensionale Analysis (u.a. Gradienten einführen und verstehen, Taylorentwicklung, konvexe Funktionen, spezielle Ableitungen)</p> <p>Numerische Verfahren zur Lösung von Gleichungssystemen:</p> <ul style="list-style-type: none"> - Direkte Ansätze (Gauß-Jordan, LU, Cholesky, QR) - Iterative Ansätze (z.B. Jacobi-Verfahren) - Konvergenzgeschwindigkeit - Lineare Gleichungssysteme mit speziellen (z.B. dünn besetzten) Matrizen <p>Gradientenabstieg:</p> <ul style="list-style-type: none"> - Line-search, Schrittweitestrategien - Newton-Verfahren (für quadratische Funktionen), Levenberg-Marquardt, Gauß-Newton, Quasi-Newton - Stochastischer Gradientenabstieg (Momentum, Nesterov Momentum) - Adaptive Lernraten (AdaGrad, RMSProp, Adam) <p>Kleinste Quadrate (Least Squares - LS):</p> <ul style="list-style-type: none"> - Formulierung - Lineares LS, nichtlineares LS, alternierendes LS - Varianten LS - Vektorielle Residuen - Robustes LS <p>Restringierte Optimierung:</p> <ul style="list-style-type: none"> - Lagrange-Multiplikatoren - SVM <p>Auf Wunsch: Spezielle Kapitel</p>

Empfohlene Literatur	<ul style="list-style-type: none"> - M. P. Deisenroth, A. A. Faisal, C. S. Ong. Mathematics for Machine Learning. Cambridge University Press. 2020. - I. Goodfellow, Y. Bengio, A. Courville. Deep Learning. MIT Press. 2016. - J. Nocedal, S. J. Wright. Numerical Optimization. Springer. 2006. - W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery. Numerical Recipes in C. Cambridge University Press. 2007.
Prüfungsleistungen	Modulprüfung
Anmerkungen	
Lehrveranstaltung Data Science und Optimierung Übung	
EDV-Bezeichnung	INFM122ML
Dozent/in	Prof. Dr.-Ing. Astrid Laubenheimer
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	Übungen und praktische Optimierungsaufgaben begleitend zur Vorlesung Optimierung.
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (benotet)
Anmerkungen	Die Übung wird mit Python durchgeführt, wahlweise auf dem eigenen Laptop oder auf PCs des Labors für Maschinelles Lernen. Erste Erfahrungen im Umgang mit Python werden vorausgesetzt. Die Übung findet mit Schwerpunkt in der zweiten Vorlesungshälfte statt.

Modul Konzepte von Programmiersprachen	
EDV-Bezeichnung	INFM120SE
Verantwortlich	Prof. Dr. Martin Sulzmann
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine

Kompetenzen	<p>Programmiersprachen sind das grundlegendste Werkzeug von Programmierern. Dieser Kurs behandelt die Grundlagen der Theorie der Programmiersprachen, ihre Anwendungen auf heute weit verbreitete Sprachen und Trends auf diesem Gebiet, die wahrscheinlich dazu beitragen werden, die Programmiersprachen von morgen zu definieren. Nach diesem Kurs verfügen die Studierenden über ein Verständnis für ein breites Spektrum an Programmiersprachenkonzepten und sind in der Lage, sich schnell an neue Programmiersprachen anzupassen. Eine Mischung aus schriftlichen und praktischen Übungen hilft den Studierenden, verschiedene Programmierstile in mehreren Programmiersprachen zu verstehen.</p> <p>Inhalt.</p> <p>Verschiedene Programmierstile mit Schwerpunkt auf funktionaler Programmierung:</p> <p>Datentypen und Mustervergleich Funktionen höherer Ordnung polymorphe Typen</p> <p>Verschiedene Arten von Polymorphismus:</p> <p>Subtypisierung (strukturell und nominal) Generics Überladung</p> <p>Programmkorrektheit:</p> <p>statische Verifizierungsmethoden Testmethoden wie QuickCheck</p> <p>Es gibt keine Abschlussprüfung. Die Bewertung basiert auf Quizzes, Midterms und mehreren kleineren Projekten, die während des Semesters durchgeführt werden.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Programmierparadigmen	
EDV-Bezeichnung	INFM121SE
Dozent/in	Prof. Dr. Martin Sulzmann
Umfang	3.0 ECTS-Punkte, 3.0 SWS 90 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Vorlesung

Lehrsprache	deutsch
Inhalt	<p>Programmiersprachen sind das grundlegendste Werkzeug von Programmierern. Dieser Kurs behandelt die Grundlagen der Theorie der Programmiersprachen, ihre Anwendungen auf heute weit verbreitete Sprachen und Trends auf diesem Gebiet, die wahrscheinlich dazu beitragen werden, die Programmiersprachen von morgen zu definieren. Nach diesem Kurs verfügen die Studierenden über ein Verständnis für ein breites Spektrum an Programmiersprachenkonzepten und sind in der Lage, sich schnell an neue Programmiersprachen anzupassen. Eine Mischung aus schriftlichen und praktischen Übungen hilft den Studierenden, verschiedene Programmierstile in mehreren Programmiersprachen zu verstehen.</p> <p>Inhalt:</p> <p>Verschiedene Programmierstile mit Schwerpunkt auf funktionaler Programmierung:</p> <ul style="list-style-type: none"> - Datentypen und Mustervergleich - Funktionen höherer Ordnung - polymorphe Typen <p>Verschiedene Arten von Polymorphismus:</p> <ul style="list-style-type: none"> - Subtypisierung (strukturell und nominal) - Generics - Überladung <p>Programmkorrektheit:</p> <ul style="list-style-type: none"> - statische Verifizierungsmethoden - Testmethoden wie QuickCheck <p>Es gibt keine Abschlussprüfung. Die Bewertung basiert auf Quizzes, Midterms und mehreren kleineren Projekten, die während des Semesters durchgeführt werden.</p>
Empfohlene Literatur	<ul style="list-style-type: none"> - Foliensammlung - Tafelmitschrift - Ausgearbeitete Beispiele und Übungen - Textbuch: Concepts in Programming Languages von John C. Mitchell
Prüfungsleistungen	Übung 1 Semester (benotet)
Anmerkungen	Seminaristischer Unterricht mit theoretischen und praktischen Übungen.
Lehrveranstaltung Optimierung von Programmen Übung	
EDV-Bezeichnung	INFM122E
Dozent/in	Prof. Dr. Christian Pape
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch

Inhalt	<p>Die Studierenden lernen, komplexe Algorithmen mit einer systemnahen prozeduralen oder objekt-orientierten Programmiersprache wie C oder C++ korrekt zu implementieren.</p> <p>Sie optimieren diese Algorithmen hinsichtlich der Ausführungsgeschwindigkeit mit verschiedenen Paradigmen wie Hybridisierung oder Techniken zur Cache-Optimierung.</p> <p>Die Teilnehmer der Veranstaltung messen die Ausführungszeiten und überprüfen empirisch die asymptotische Laufzeit der implementierten Verfahren. Dazu erstellen sie verschiedener Testszenarien, um die Algorithmen darauf basierend zu vergleichen und zu beurteilen.</p> <p>Die Studierenden dokumentieren und diskutieren die Ergebnisse. Sie analysieren ihre Optimierungen und erklären die Resultate.</p> <p>Die zu behandelnden Informatikprobleme variieren. Typische Inhalte sind:</p> <ul style="list-style-type: none"> - Raytracing - Schnelle Multiplikation von Polynomen und Zahlen - Grafisch-Geometrische-Algorithmen - Graphenalgorithmen
Empfohlene Literatur	Die Übungsaufgaben und Zusatzmaterialien werden elektronisch (HTML, PDF, Programmgerüste) zur Verfügung gestellt.
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	<p>Vorkenntnisse:</p> <p>Vertiefte praktische Kenntnisse einer prozeduralen oder objekt-orientierten Programmiersprache wie C, C++, C# oder Java. Implementierung und Testen grundlegender Algorithmen aus dem Bereich der Such-, Graphen- und Sortierprobleme.</p> <p>Format: Praktische Arbeit, Übungen im Labor mit geringem Vorlesungsanteil.</p> <p>Die Ergebnisse werden in den praktischen Übungen von den Studenten vorgestellt und mit dem Dozenten diskutiert. Die Ergebnisse sind von den Studenten in einem Bericht zu dokumentieren.</p> <p>Betreuung:</p> <p>Individuelle Betreuung durch den Dozenten während der Präsenzzeit im Rechnerlabor. Außerhalb der Präsenzzeit via E-Mail oder während der Sprechzeiten des Dozenten.</p>

Modul Wissenschaftstheorie und Ethik	
EDV-Bezeichnung	INFM130
Verantwortlich	Prof. Dr. Thomas Morgenstern
Umfang	5.0 ECTS-Punkte, 3.0 SWS
Einordnung	2. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierenden in der Lage über die theoretischen und methodischen Grundlagen der Wissenschaft Informatik und die ethische Tragweite ihres Berufsfelds Informatik reflektiert nachzudenken, sie einzuordnen und zu beurteilen. Daraus gewinnen sie Orientierung für ihre wissenschaftliche und alltäglich-berufliche Praxis. Sie kennen unterschiedliche Konzepte und Methoden der Wissenschaft und deren Ziele und können sie anwenden und sich mit ethischen Fragen der Informatik auseinandersetzen. Aus dieser Reflektion heraus können sie eigene Maßstäbe für ihr Urteilen und verantwortliches Handeln als angehende Informatiker entwickeln.
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Wissenschaftstheorie für Informatik	
EDV-Bezeichnung	INFM131
Dozent/in	Prof. Dr. Thomas Morgenstern
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	Die Wissenschaftstheorie befasst sich mit methodisch gewonnenen, in Theorien formulierten wissenschaftlichen Wissen. Sie untersucht die Bildung von wissenschaftlichen Begriffen und Strukturen von Theorien und die dabei eingesetzten Methoden. Speziell klären wir Fragen wie: "Was ist Wissen, Wissenschaft, eine Theorie, eine Beschreibung, eine Erklärung, eine Begründung, eine Bestätigung, Zweifel?" "Welches sind die Hauptströmungen der Wissenschaftstheorie: Realismus, Empirismus, Naturalismus, Kritischer Rationalismus, Konstruktivismus, Sprachphilosophie, Strukturalismus, Holismus?" "Was ist Information, ein Computer, ein Programm?" "Wie kann man methodisch Wissen erlangen: begrifflich, theoretisch, logisch-deduktiv, empirisch, induktiv, simulativ, hermeneutisch, diskursiv?" "Welchen Bezug haben wissenschaftliche Theorien zur Realität und zur wirklichen Welt?"

Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungsskript und Mitschrift. - Stephan Kornmesser, Wilhelm Büttemeyer: "Wissenschaftstheorie", Springer Verlag. - Alan F. Chalmers: "Wege der Wissenschaft", Springer Verlag. - Wolfgang Stegmüller: "Probleme und Resultate der Wissenschaftstheorie und Analytischen Philosophie", Springer Verlag. - Klaus Mainzer: "Gehirn, Computer, Komplexität", "Computernetze und virtuelle Realität", "The Universe as Automaton", "Künstliche Intelligenz", "Wie berechenbar ist unsere Welt?", "Quantencomputer", "Philosophisches Handbuch Künstliche Intelligenz", "Grenzen der KI", "Zukunft durch nachhaltige Innovation", Springer Verlag. - William J. Rapaport: "Philosophy of Computer Science", Wiley. - Luciano Floridi: "The Philosophy of Information", Oxford University Press. <p>Abhängig von einem gewählten Schwerpunkt kann noch weitere Literatur empfohlen werden.</p>
Prüfungsleistungen	Klausur/mündl. Prüfung/Prakt. Arbeit 60/20/11 Min./Min./Semester (benotet)
Anmerkungen	Seminaristischer Unterricht mit Übungen, Referaten und Diskussionen. In unterschiedlichen Semestern können verschiedene Schwerpunkte gesetzt werden und auf aktuelle Themen eingegangen werden.
Lehrveranstaltung Ethik für Informatik	
EDV-Bezeichnung	INFM132
Dozent/in	Prof. Dr. phil. Ziad Mahayni
Umfang	2.0 ECTS-Punkte, 1.0 SWS 60 Stunden gesamt, davon 15 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>Informations- und Digitaltechnologien führen zu einem fundamentalen Wandel der Gesellschaft und der Rolle des Menschen. Damit einher gehen neue ethische Fragen von großer Reichweite wie etwa dem Umgang mit Künstlicher Intelligenz, der Zukunft menschlicher Arbeit im Kontext voranschreitender Automatisierung, der Überwachung und Manipulation menschlichen Verhaltens, der Schaffung neuer Lebensräume in virtuellen Welten oder der technischen Optimierung des Menschen (Trans-/Posthumanismus). In der Lehrveranstaltung werden diese ethischen Fragen herausgearbeitet, Antwortoptionen diskutiert und ethisch bewertet. Schließlich wird der Einfluss des Berufsfelds Informatik auf diese Entwicklungen reflektiert und ethische Leitlinien für die Entwicklung von Informationstechnologien diskutiert.</p>

Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungs-Skript und Mitschriebe. - Julian Nida-Rümelin, Nathalie Weidenfeld: Digitaler Humanismus. Eine Ethik für das Zeitalter der Künstlichen Intelligenz, Piper Verlag - Sarah Spiekermann: Digitale Ethik: Ein Wertesystem für das 21. Jahrhundert, Droemer Verlag - Cathy O'Neil: Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy, Penguin Random House - Michael Hauskeller, Mythologies of Transhumanism, Palgrave Macmillan - Leitlinien für eine Ethik der Künstlichen Intelligenz wie sie z.B. von der Europäischen Union, UNESCO oder auch von Unternehmen formuliert worden sind. <p>Abhängig von einem gewählten Schwerpunkt kann noch weitere Literatur empfohlen werden.</p>
Prüfungsleistungen	Klausur/mündl. Prüfung/Prakt. Arbeit 60/20/1 Min./Min./Semester (benotet)
Anmerkungen	Seminaristischer Unterricht mit Übungen, Referaten und Diskussionen. In unterschiedlichen Semestern können verschiedene Schwerpunkte gesetzt werden und auf aktuelle Themen eingegangen werden.

Modul Managementkompetenz	
EDV-Bezeichnung	INFM140
Verantwortlich	Prof. Dr. rer. pol. Mathias Philipp
Umfang	5.0 ECTS-Punkte, 4.0 SWS
Einordnung	1. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Das Modul Managementkompetenzen setzt sich aus den Teilen "IT Project Management" und "IT Entrepreneurship" zusammen.</p> <p>Die Studierenden werden in der Veranstaltung IT Project Management in die Lage versetzt, Entwicklungsprojekte im SWE eigenständig zu planen und gegebenenfalls durchzuführen. Die Studierenden werden zu diesem Zweck mit verschiedenen branchenspezifischen Anforderungen, Methoden, Vorgehensmodelle und Werkzeuge vertraut gemacht. Die Veranstaltung findet auf English statt. Ziel ist es, die Studierenden auf internationale IT-Projekte vorzubereiten.</p> <p>Die Studierenden erlangen im IT-Management die Kompetenz für die Führungsaufgaben im IT-Bereich. Anhand eines Lebenszyklusmodells von der Unternehmensgründung (IT Entrepreneurship) bis zum Management eines IT-Unternehmens werden die jeweiligen Methoden und Werkzeuge erlernt und an Fallbeispielen von dem Studierenden selbstständig angewendet. Sie erhalten damit die notwendigen Grundlagen, als eigenständiger IT-Unternehmer sowie auch als Führungskraft im IT-Bereich tätig zu sein.</p>
Prüfungsleistungen	Klausur 120 Min. (benotet)
Lehrveranstaltung IT Projektmanagement	
EDV-Bezeichnung	INFM141.a
Dozent/in	Prof. Dr. Uwe Haneke
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	englisch

Inhalt	<p>Die Studierenden sollen in der Veranstaltung IT Project Management in die Lage versetzt werden, IT Projekte eigenständig zu planen und gegebenenfalls durchzuführen. Zu diesem Zweck werden in der Veranstaltung unterschiedliche Konzepte und Werkzeuge vorgestellt und in Fallstudien eingesetzt.</p> <p>Die Veranstaltung findet auf English statt. Ziel ist es, die Studierenden auf internationale IT-Projekte vorzubereiten.</p> <p>Insbesondere werden die nachfolgenden Anforderungen an das Management von IT-Projekten beherrscht:</p> <ul style="list-style-type: none"> - Moderne Vorgehensmodelle im IT-Projektmanagement (agile Methoden) - Design Thinking - Frameworks für das Skalieren agiler IT-Projekte - Erstellung von Lasten- und Pflichtenheften - Planungsmethoden für IT-Projekte - Risikomanagement - Berichtswesen im IT-Projektmanagement - Spezielle Vorgehensmodelle im Bereich KI/ML
Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungsbegleitende Foliensätze - Übungsaufgaben - Ausgewählte Literatur
Prüfungsleistungen	Modulprüfung
Anmerkungen	Vorlesung 50%, Übungen 20%, Gruppenarbeit 30%
Lehrveranstaltung IT-Entrepreneurship	
EDV-Bezeichnung	INFM141.b
Dozent/in	Prof. Dr. rer. pol. Mathias Philipp
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

Inhalt	<p>Die Studierenden erwerben in dieser Veranstaltung die Kompetenz, als eigenständiger IT-Unternehmer als auch als Führungskraft im IT-Bereich tätig zu sein. Anhand eines Lebenszyklusmodells (von der Unternehmensgründung (IT-Entrepreneurship) bis zum Management eines IT-Unternehmens werden die jeweiligen Methoden und Werkzeuge erlernt und an Fallbeispielen von dem Studierenden selbstständig angewendet.</p> <p>Folgende Lebenszyklusphasen werden behandelt:</p> <ul style="list-style-type: none"> - Studierender und freiberuflicher Softwareentwickler - Angestellter freiberuflicher Softwareentwickler (Nebenerwerbsgründung) - Unternehmensgründung (IT-Entrepreneurship) - Management eines IT-Unternehmens <p>A. Freiberuflicher Softwareentwickler</p> <ol style="list-style-type: none"> 1. Rechnung, Angebot und Haftung 2. Freiberufler vs. Gewerblich 3. Finanzamt: Kleinunternehmertum, Umsatzsteuer, Steuererklärung 4. IT-Recht für Software-Entwickler: Vertragsrecht, Urheberrecht, ... <p>B. Wir werden IT-Unternehmer (IT-Entrepreneur)</p> <ol style="list-style-type: none"> 1. Entrepreneurship 2. Geschäftsidee, Geschäftsmodell und Businessplan 3. Rechtsformen und Finanzierungsinstrumente <p>C: Wie manage ich ein IT-Unternehmen? (IT Management)</p> <ol style="list-style-type: none"> 1. Überblick Modelle des IT-Managements 2. IT Governance, IT-Strategie, IT-Organisationsformen, Internationale Projektsteuerung 3. IT-Organisationsentwicklung 4. IT-Sicherheitsmanagement
Empfohlene Literatur	<ul style="list-style-type: none"> - Vorlesungsmaterial vollständig als PowerPoint-Folien verfügbar - Tafelaufschrieb bei interaktiver Erarbeitung von Kernproblemstellungen - Zahlreiche Multiple-Choice Fragen zu den einzelnen Lernmodulen
Prüfungsleistungen	Modulprüfung
Anmerkungen	Teilnahme am seminaristischen Unterricht.

Modul Projektbasiertes wissenschaftliches Arbeiten unter Anleitung 1	
EDV-Bezeichnung	INFM150
Verantwortlich	Prof. Dr. Thomas Fuchß
Umfang	5.0 ECTS-Punkte, 4.0 SWS
Einordnung	1. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, wissenschaftliche Themen oder praxisnahe Projekte im IT-Bereich selbstständig und im Team zu bearbeiten. Sie wenden ihre im Studium erworbenen Kenntnisse gezielt auf komplexe Aufgabenstellungen an, analysieren Probleme und entwickeln wissenschaftlich fundierte Lösungsansätze. Darüber hinaus planen und setzen sie Konzepte eigenständig um und können fortgeschrittene industrielle und wissenschaftliche Projekte eigenverantwortlich durchführen. Durch die intensive Zusammenarbeit im Team entwickeln die Studierenden ihre sozialen und organisatorischen Fähigkeiten weiter und sind in der Lage, Arbeitsergebnisse gemeinsam zu präsentieren.
Prüfungsleistungen	Mündliche Prüfung 20 Min. (benotet)
Lehrveranstaltung Projektarbeit 1	
EDV-Bezeichnung	INFM151
Dozent/in	Alle Professoren
Umfang	5.0 ECTS-Punkte, 4.0 SWS 150 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Praktische Arbeit
Lehrsprache	deutsch
Inhalt	Arbeitsgebiete können durch jeden Professor der Fakultät eröffnet werden, dies erfolgt durch Aushang zu Semesterbeginn. Forschungsprojekte bewegen sich an der vorderen Front der aktuellen Informatikforschung und können in Zusammenarbeit mit Forschungsinstitutionen durchgeführt werden. Anwendungsprojekte sind von besonderer Relevanz für die industrielle Praxis und können in Zusammenarbeit mit Industrieunternehmen durchgeführt werden.
Empfohlene Literatur	- Nach Maßgabe der Arbeitsgruppe
Prüfungsleistungen	Praktische Arbeit 1 Semester (benotet)
Anmerkungen	- Vorkenntnisse: Nach Maßgabe der Arbeitsgruppe - Format: Präsenzzeit mit Gruppendiskussion 30 %, selbständige Arbeit 70 %. - Betreuung: In der Regel wird wöchentlich zu einem festen Termin eine Sitzung der Arbeitsgruppe stattfinden.

Modul Game Design	
EDV-Bezeichnung	INFM210MI
Verantwortlich	Prof. Daniel Schwarz
Umfang	7.0 ECTS-Punkte, 5.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine

<p>Kompetenzen</p>	<p>Wissen:</p> <ul style="list-style-type: none"> - Wissen über den gesamten Konzeptions-, Gestaltungs- und Entwicklungsprozess von Computerspielen. - Kenntnis einer ganzheitlichen Game-Design-Methodologie, die genre- und inhaltsübergreifend für die Konzeption und Gestaltung von Computerspielen eingesetzt werden kann. - Kenntnis der verschiedenen Design-Disziplinen, die für die Herstellung der Erfahrungsräume in Computerspielen wichtig sind, sowie ihre Integration zu einer immersiven multimodalen und multimedialen Erfahrung, die beim Gestaltungsprozess von Spielen angestrebt wird. - Theoretisches Wissen über die Arten, Wesensmerkmale, Eigenschaften, Gestaltung und Wirkungen von Computerspielen als ein Medium, um Erfahrungsräume zu schaffen und Erfahrungen zu vermitteln. - Kenntnisse von System-Denken und System-Theorie. <p>Können:</p> <ul style="list-style-type: none"> - Anwendung einer Game-Design-Methodologie auf eine eigene Idee für ein Spiel - Erstellung eines Game Concept Documents mit allen notwendigen Entwicklungsschritten eines vollständig Spielkonzepts - Umsetzung des Game Concept Documents im Übungskurs als spielbaren Prototyp - Umsetzung von Feedback im iterativen Design-Ansatz bei der wechselseitigen Weiterentwicklung von Konzept und Prototyp <p>Haltung:</p> <ul style="list-style-type: none"> - Bewusstsein über die transdisziplinäre Natur von Game Design als integrierende Gestaltungsdisziplin und der dafür erforderlichen breiten Bildung in unterschiedlichen Bereichen der Künste, - (Geistes-)Wissenschaften und Technik. - Computerspiele werden als Lernumgebungen aufgefasst, in denen die Spielenden Lern-Erfahrungen machen, das Game Design ist somit in der Tradition von Pädagogik und Didaktik zu begreifen. - Systemische, ganzheitliche Sichtweise, die beim Game Design - Prozess für das world building notwendig sind. - System-Design als entscheidende Verständnis-Brücke zwischen den Game Designern und den Game Developern im Prozess der Spielentwicklung. <p>Lehr-/Lernmethode</p> <p>Das Wissen um Konzeption, Gestaltung und Entwicklung von Computerspielen wird in der Vorlesung anhand vieler transparenter Herstellungsbeispiele aus der Praxis professioneller Spiele-Produktionen anschaulich und detailliert vermittelt.</p> <p>Der Prozess zur Herstellung eines Computerspiels von der Ideen-Findung bis hin zur Veröffentlichung und Rezeption der Spieleproduktion ist Gliederung und Leitfaden der gesamten Vorlesung. Entlang der Präsentation des Entwicklungsprozesses eines Computerspiels entfaltet die Vorlesung an entsprechenden</p> <p>Gestaltungs- und Produktionsschritten auch wichtige Erkenntnisse aus der Theorie (Game Studies) und Praxis (aktuelle Entwicklungen in der</p>
--------------------	--

	<p>Computerspiel-Industrie).</p> <p>Damit der komplexe Herstellungsprozess eines Computerspiels nicht nur in seiner Gänze dargestellt, sondern auch angewandt und somit verstanden wird, vermittelt die Vorlesung eine etablierte Game-Design-Methodologie zur Erstellung eines Spielkonzepts in Form eines Game Concept Documents.</p> <p>Während der Vorlesung werden die einzelnen Entwicklungsschritte und Kapitel der Game-Design-Methodologie vorgestellt, mit Beispielen ergänzt und von den Studierenden dann direkt als (wöchentliche) Teilaufgabe der Prüfungsleistung auf die Entwicklung ihres eigenen Spielkonzepts direkt angewandt. Die Spielkonzepte der Studierenden werden besprochen, es gibt Feedback und iterative Verbesserungsmöglichkeiten. So sind am Ende der Vorlesung alle Entwicklungsschritte und Kapitel der Game-Design-Methodologie durchlaufen worden und das vermittelte Wissen wurde von den Studierenden direkt auf die eigene Spiel-Idee angewandt und konstruktiv kritisiert worden.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Game Design	
EDV-Bezeichnung	INFM211MI
Dozent/in	Prof. Daniel Schwarz
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

Inhalt	<p>Zu Beginn der Vorlesung wird eine Übersicht unterschiedlicher Spiele-Produktionen des Entwicklerstudios takomat GmbH von Professor Schwarz gegeben. Dadurch wird den Studierenden eine interne Sichtweise auf den Prozess der Akquise, Konzeption, der Gestaltung, Produktion und Entwicklung bis hin zur Veröffentlichung von Computerspielen anhand ausgewählter Beispiele aus erster Hand gezeigt. Nach diesen Einblicken in die Praxis der Produktion verlassen</p> <p>wir die „Hexenküche“ des speziellen Entwicklerstudios takomat und verallgemeinern die dort kennengelernten Design- und Wirkungsprinzipien von Computerspielen: Die Frage „Was ist ein Spiel“ und „Wie wirkt ein Spiel?“ werden mit den theoretischen Erkenntnissen der aktuellen Spielwissenschaft (deutsch: Ludologie / im angelsächsischen Raum: game studies) und ihrer bedeutendsten Wissenschaftler und Praktiker (Katie Salen und Eric Zimmerman, Jane McGonigal, Jesse Schell, Paul Gee) beantwortet. Die Spielwissenschaften sind seit ihrem Aufschwung in den späten 90er Jahren noch keinesfalls eine konsolidierte Forschungs-Disziplin und entsprechen in ihrer Transdisziplinarität immer noch dem „produktiven Chaos“, das der Spieleforscher Jesper Juuls ihnen 2005 attestierte. Der Game Designer, Produzent und Theoretiker Jesse Schell beschreibt in seinem Buch „The Art of Game Design“ dazu treffend: Die Disziplin des Game Designs hatte noch nicht wie die Chemie ihren Mendelejew, der mit der Einführung des Periodensystems der Quacksalberei der Alchimie ein Ende machte und die Chemie auf ein fundiertes und funktionales Theorie-Gebäude stellte. „Der Mensch ist nur da ganz Mensch, wo er spielt.“ (Friedrich Schiller) Da das Spiel den ganzen Menschen in seiner komplexen Ganzheit erfasst, berührt und bewegt (Huizinga, McLuhan), müssen wir also nach wie vor die HexenmeisterInnen aller involvierten Wissenschafts-, Technik- und Design-Disziplinen um den Hexenkessel versammeln, in dem die Magie des Mediums Computerspiel gebraut werden soll. Literatur-, Kultur- und Medienwissenschaften betreten den „magischen Kreis“ der Spiele ebenso wie die Psychologie oder Anthropologie, der große Reigen der unterschiedlichen Design-Disziplinen (Visual Design, Sound Design, 3D-Design, Animation, Interface Design, Interaction Design, System Design, Story Design, Character Design, Game Design), die Informatik und Kybernetik.</p> <p>Die verschiedenen Design-Disziplinen, die zur Gesamtwirkung eines Computerspiels integriert werden - werden aufgeführt und ihre Querverbindung, Abhängigkeiten und Integrationsmöglichkeiten verdeutlicht.</p> <p>Mit Hilfe dieser multi- und interdisziplinären Sichtweise auf Computerspiele zeigt die Vorlesung Design-Muster, Bausteine und Ziele von Game Design als spielgenre-unabhängige Konzepte auf. Anhand der eingangs gezeigten Beispiele von Computerspielproduktionen wird eine Game Design Methodologie präsentiert, die von den Studierenden in eigenständiger Arbeit zur Erstellung eines eigenen Game-Konzepts angewandt wird. Die Game Design Methodologie wurde in über 20 unterschiedlichster Spiele-Produktionen eingesetzt und begleitet von der Themenfindung bis hin zur Umsetzung der Kernerfahrung des Spiels in einem spielbaren Prototypen ein kapitelweises Vorgehen.</p>
--------	---

Empfohlene Literatur	<ul style="list-style-type: none"> - Salen, Katie, Zimmerman Eric, Rules of Play - Game Design Fundamentals, The MIT Press 2003 - Salen, Katie, Zimmermann Eric, The Game Design Reader - A Rules of Play Anthology, The MIT Press 2006 - Schell, Jesse, The Art of Game Design - A book of lenses, second edition, CRC Press, Taylor & Francis Group 2015 - McGonigal, Jane, Besser als die Wirklichkeit!: Warum wir von Computerspielen profitieren und wie sie die Welt verändern, Heyne Verlag 2011 - Hagner, Kerner, Thomä, Theorien des Computerspiels - zur Einführung, Junius Verlag GmbH, 2012 - Troy Duniway, Jeannie Novak, Game Development Essentials: Gameplay Mechanics, Delmar Cengage Learning, 2008
Prüfungsleistungen	Mündl. Prüfung/Entwurf 20/1 Min./Semester (benotet)
Anmerkungen	
Lehrveranstaltung Game Design Übung	
EDV-Bezeichnung	INFM212MI
Dozent/in	Prof. Daniel Schwarz
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch

<p>Inhalt</p>	<p>Wissen:</p> <ul style="list-style-type: none"> - Kenntnisse über Aufbau und Handhabung des Szenen-Editor von Unity. - Kenntnisse über Spiel-Design im Szenen-Editor von Unity. - Kenntnisse über C im Szenen-Editor von Unity und - Microsoft Visual Studio. - Kenntnisse über den Publishing-Prozess von Unity. <p>Können:</p> <ul style="list-style-type: none"> - Grundlagen der Spielentwicklung mit einer Game Engine („Unity“) - Skripting und Bauen des Spiels im Szenen-Editor von Unity. - Umsetzung des im Vorlesungs-Teil entwickelten Game Concept Documents als spielbaren Prototypen - Vollständiges Durcharbeiten des iterativen Game-Design-Prozesses mit seinen Phasen „Konzeption“, „Design“, „Development“, „User-Testing“, „Evaluation“ und „Feedback“ und daraus resultierendem „Re-Design“. - Durchführung von User-Testings und Evaluation des Feedbacks. - Umsetzung von User-Feedback im iterativen Design-Ansatz bei der wechselseitigen Weiterentwicklung von Konzept und Prototyp. <p>Haltung:</p> <ul style="list-style-type: none"> - Game Design ist „Übungen in Demut“: Entscheidende Kompetenz eines guten Game Designers ist - neben all den vielen Design- und Wissenschafts-Disziplinen, die zu einem guten Game Design beitragen - die Fähigkeit, zuzuhören. - Sprachlich präzises und reichhaltiges Ausdrucksvermögen ist das mächtigste Schwert des Game Designers, denn wie bei Wittgenstein gilt für die Gestaltung der Spielewelt: „Die Grenzen meiner Sprache sind die Grenzen meiner Welt.“ <p>Lehr-/Lernmethode:</p> <p>Aufeinander aufbauende und sich nacheinander erweiternde Übungen befähigen die Studierenden, sich zuerst mit der Game Engine vertraut zu machen und führen schließlich über Modifikation und Erweiterungen zur Umsetzung eines spielbaren Prototypen des im Vorlesungskurs selbst entwickelten Spielkonzeptes. Durch die Gestaltung und Inhalte der Übungen werden wichtige Stationen der Spielentwicklung aus der Praxis von Spieleentwickler-Studios nachgestellt, angewandt, durchgeführt und so erlernt. Alle Spiel-Prototypen, die in den Übungen schrittweise und aufeinander aufbauend entwickelt werden, werden in der Gruppe der Kursteilnehmenden präsentiert und durch User-Testing innerhalb der Gruppe evaluiert, das Feedback wird von allen gegeben und gemeinsam analysiert und Verbesserungen und Änderungen für die nächste Iteration des Prototypen festgelegt. Durch diese Lernmethode wird der gesamte iterative Design-Zyklus der Spieleentwicklung durchlaufen, erprobt und trägt zur Verbesserung der Spielerfahrung des Prototypen bei.</p> <p>Inhalt</p> <p>Im Übungsteil von „Game Design und Development“ gibt es insgesamt fünf Übungen, die aufeinander aufbauen, sich weiterentwickeln und das vollständige „Nachspielen“ eines realen Game-Design- und Development-Prozesses</p>
---------------	---

durchlaufen. In der ersten Übung wird zunächst ein einfaches Tutorial in der Game Engine „Unity“ nachprogrammiert, um sich mit der Entwicklungs-Umgebung und dem Szenen-Editor der Spieltechnologie Unity vertraut zu machen. Das Wissen um die vier wesentlichen Kern-Elemente eines Spiels aus dem Vorlesungskurs wird dann in der darauffolgenden Übung 2 genutzt, um eine Modifikation („Mod“) des nachprogrammierten Tutorials zu machen, in dem genau eines dieser Elemente verändert wird. Die Studierenden schauen dann, wie diese Änderung die Spielerfahrung beeinflusst, und ob die Änderung ausreicht, um das Spiel zu einem anderen Spiel werden zu lassen. Nach einem User Testing des „Mods“ wird dieser nochmals iterativ dem Feedback entsprechend geändert und wiederum präsentiert (3. Übung). Zu diesem Zeitpunkt ist die parallel zu den Übungen ablaufende Entwicklung der eigenen Spiele-Idee und Konzeption im Vorlesungs-Kurs bereits soweit fortgeschritten, daß nun in Übung 4 versucht werden soll, die eigene Spiele-Idee in dem „Mod“ umzusetzen. Die Studierenden sollen dabei bewußt versuchen, die Idee in den bisherigen „Mod“ einzuarbeiten, bzw. sie dem „Mod“ überzustülpen. Das entspricht der realen Vorgehen von Spieleentwickler-Studios, bei der Neuentwicklung von Spielen bisherige Spiele und Technologie-Entwicklungen aus der Schublade zu holen und zu versuchen, sie weiter zu nutzen und so zu ändern, daß sie das neue Spielkonzept umsetzen können. Diese Übung zeigt den Studierenden entweder, dass sie auf den bisherigen „Mod“ weiter zu ihrem Prototypen ausbauen können, oder sie können diese Art der Spielumsetzung bereits ausschließen und sich mit dieser Erfahrung zu einer anderen Umsetzung ihrer Spiel-Idee inspirieren lassen. Die letzte Übung 5 wird dann die verbleibende Zeit zur Weiterentwicklung des Prototypen der eigenen Spiel-Idee nutzen. In wöchentlichem Turnus werden dann bei gemeinsamer Präsentation und Evaluation die Prototypen im iterativen Design-Zyklus der Spiele-Entwicklung bearbeitet, bis der Prototyp dann am Ende des Kurses als wesentliche Prüfungsleistung in Form einer publizierten spielbaren Software und einem „Playthrough-Video“ abgegeben wird.

Prüfung
Das im Vorlesungsteil entwickelte eigene und vollständige Spielkonzept wird im Übungskurs als spielbarer Prototyp umgesetzt.

Empfohlene Literatur	<ul style="list-style-type: none"> - Salen, Katie, Zimmerman Eric, Rules of Play – Game Design Fundamentals, The MIT Press 2003 - Salen, Katie, Zimmermann Eric, The Game Design Reader – A Rules of Play Anthology, The MIT Press 2006 - Schell, Jesse, The Art of Game Design – A book of lenses, second edition, CRC Press, Tayer & Francis Group 2015 - Adams, Earnest, Dormans, Joris, Game mechanics: Advanced Game Design, - McGonigal, Jane, Besser als die Wirklichkeit!: Warum wir von Computerspielen profitieren und wie sie die Welt verändern, Heyne Verlag 2011 - Hagner, Kerner, Thomä, Theorien des Computerspiels - zur Einführung, Junius Verlag GmbH, 2012 - Troy Dunniway, Jeannie Novak, Game Development Essentials: Gameplay Mechanics, Delmar Cengage Learning, 2008
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	

Modul Künstliche Intelligenz	
EDV-Bezeichnung	INFM210ML
Verantwortlich	Prof. Dr. Patrick Baier
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	Maschinelles Lernen
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Die Studierende lernen den aktuellen Stand der Forschung im Bereich "Künstliche Intelligenz" in Theorie und Praxis kennen.</p> <p>Die Studierende haben einen Überblick über aktuelle Methoden im Bereich "Deep Learning" und kennen relevante Anwendungsbereiche in der Praxis.</p> <p>Die Inhalte des Moduls Maschinelles Lernen (Methoden und praktische Kenntnisse) werden als bekannt vorausgesetzt. Die Inhalte des Moduls Data Science sind von für den praktischen Teil des Moduls Künstliche Intelligenz von Vorteil.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Künstliche Intelligenz	
EDV-Bezeichnung	INFM211ML
Dozenten	Prof. Dr. Patrick Baier Prof. Dr. Jannik Strötgen
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

<p>Inhalt</p>	<p>In dieser Vorlesung werden aktuelle Entwicklungen und aktuelle Forschungsergebnisse aus dem Bereich der Künstlichen Intelligenz, insbesondere des Deep Learnings, abgedeckt.</p> <p>Nach der Teilnahme an der Vorlesung sind die Studierenden in der Lage, die Grundlagen von neuronalen Netzen einzuschätzen und aktuelle Forschungsmethoden aus den Bereichen Computer Vision und Natural Language Processing zu beurteilen.</p> <p>Die Studierenden können grundlegende Architekturen wie Convolutional Neural Networks und Recurrent Neural Networks bewerten und aktuelle Architekturen wie Transformer-Modelle analysieren. Weiterhin sind Studierende in der Lage für die Anwendungsbereiche Computer Vision und Natural Language Processing entsprechende Architekturen zu bewerten sowie Fachliteratur wie aktuelle Publikationen zu verstehen und Teilaspekte herauszustellen. Zusätzlich sind die Studierenden in der Lage aktuelle Methoden aus einzelnen Teilbereichen wie dem Continual Learning zu verstehen.</p> <p>Die Vorlesung dient als Grundlage für die Wissensvermittlung für das Gesamtmodul „Künstliche Intelligenz“ und als theoretisches Pendant zur praktischen Veranstaltung „AI Lab“. Praktische Übungen sind also nicht Teil der Vorlesung. Für die tiefgehende Auseinandersetzung mit einzelnen Themen sind Gruppenpräsentationen möglich.</p> <p>Inhaltlich werden in der Vorlesung folgende Themen abgedeckt:</p> <ul style="list-style-type: none"> - Wiederholung der Grundlagen von Neuronalen Netzen und des Deep Learnings - Verschiedene Architekturen von CNNs - Anwendungsbeispiele aus dem Bereich Computer Vision wie Object Detection und Instance Segmentation - Modelle für das Verarbeiten sequentieller Daten wie RNNs und (Bi-)LSTMs - Anwendungsbeispiele aus dem Bereich Natural Language Processing wie Machine Translation und Information Extraction - Language Models und Word Embeddings - Attention Mechanismus und Transformer-Modelle - Große Sprachmodelle LLMs - Grundlegende anwendungsübergreifende Methoden wie Transfer Learning - Einführungen in Spezialgebiete wie Explainable AI, Continual Learning (z.B. zur Verbesserung von LLMs) oder Diffusion Models <p>In Form einer Klausur wird überprüft, inwieweit die Studierenden die grundlegenden Konzepte und Architekturen des Deep Learnings verstanden haben sowie aktuelle Forschungsergebnisse bewerten können. Das Beantworten der Fragen erfordert teils eigene Formulierungen und teils das Ausführen von eigenen Berechnungen mit Beispielen aus den Anwendungsdomänen. Zusätzlich haben Studierende die Möglichkeit mit einer freiwilligen Gruppenpräsentation eines Spezialthemas zu demonstrieren, dass sie einzelne Themen detailliert bewerten und erklären können. Eine</p>
---------------	--

	Präsentation, die durch fundierte Analyse und verständliche Darstellung überzeugt, führt zu einer Verbesserung der Klausurnote um einen Notenschritt.
Empfohlene Literatur	
Prüfungsleistungen	Klausur/mündl. Prüfung 60/20 Min. (benotet)
Anmerkungen	
Lehrveranstaltung Künstliche Intelligenz Übung	
EDV-Bezeichnung	INFM212ML
Dozent/in	Dr. Patrick Baier
Umfang	4.0 ECTS-Punkte, 4.0 SWS 120 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	Die Lehrveranstaltung beinhaltet die praktische Umsetzung der in der KI-Vorlesung erworbenen Kenntnisse. Dies geschieht exemplarisch am Beispiel dreier Themenfelder: * Computer Vision * Natural Language Processing * Reinforcement Learning Dabei werden unterschiedliche Problemstellungen innerhalb dieser drei Anwendungsfelder behandelt und Lösungen in Form von Vorhersagemodellen entworfen. Zur praktischen Umsetzung werden diese dann mit Hilfe von Python und PyTorch implementiert.
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (benotet)
Anmerkungen	Zur erfolgreichen Teilnahme am Labor sind Grundkenntnisse in Python erforderlich. Die Anzahl der Teilnehmer ist begrenzt.

Modul Theorie effizienter Algorithmen	
EDV-Bezeichnung	INFM210SE
Verantwortlich	Prof. Dr. Heiko Körner
Umfang	7.0 ECTS-Punkte, 5.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Die Studierenden werden befähigt, effiziente Algorithmen in Theorie und Praxis zu entwerfen. Sie können die Korrektheit von diversen graphentheoretischen Problemen mit exakten logischen Schlüssen nachweisen. Sie sind in der Lage, Laufzeiten von Verfahren zu analysieren und passende Analysetechniken einzusetzen. Sie können zudem Modellierungs- und Simulationsverfahren für die computergestützte Auslegung von Prozessabläufen selbstständig implementieren sowie diverse Iterationsverfahren exemplarisch auf modernen Hochleistungsrechnern parallelisieren.
Prüfungsleistungen	Klausur/mündl. Prüfung 120/20 Min. (benotet)
Lehrveranstaltung Graphenalgorithmen	
EDV-Bezeichnung	INFM211SE.a
Dozent/in	Prof. Dr. Heiko Körner
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	Die Lehrveranstaltung vermittelt grundlegende Algorithmen auf Graphen und zeigt auf, wie man deren Korrektheiten und Zeitkomplexitäten analysiert. Nach einer kurzen Einführung in die Graphentheorie werden zunächst Durchmusterungsmethoden wie die Breiten- und Tiefensuche vorgestellt. Weitere Algorithmen befassen sich mit der Erkennung von starken Zusammenhangskomponenten, topologischen Sortierungen sowie der Berechnung von kürzesten Wegen. Effiziente Tests auf die Kreisfreiheit von Graphen werden ebenfalls besprochen. Die Vorlesung befähigt die Teilnehmer, selbstständig weiterführende Algorithmen zu erarbeiten, beweisbar sicher anzuwenden und ihren Nutzen einzuschätzen.
Empfohlene Literatur	- Tafelanschrieb - Skript - Musterlösungen für alle Übungsaufgaben - T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein: Introduction to Algorithms, 4th edition. MIT Press, 2022.
Prüfungsleistungen	Modulprüfung

Anmerkungen	Die Lehrveranstaltung findet als Vorlesung statt. Begleitende Übungen vertiefen die vermittelten Gebiete.
Lehrveranstaltung SAT Solving	
EDV-Bezeichnung	INFM211SE.a
Dozent/in	Prof. Dr. Carsten Sinz
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>SAT-Solving ist eines der wichtigsten allgemeinen Verfahren zur Lösung schwerer (oft NP-vollständiger) kombinatorischer Probleme. Diese treten in der Praxis in einer Vielzahl von Anwendungen auf, z.B. hier:</p> <ul style="list-style-type: none"> - Planungs- und Scheduling-Probleme in Lieferketten - Konfiguration komplexer, variantenreicher Produkte, z.B. PKWs, LKWs, Flugzeuge - Prüfung (Verifikation) von Hardware- und Software - Erstellung von Spielplänen, z.B. in der Bundesliga <p>Dieses Modul vermittelt Studierenden die theoretischen und schwerpunktmäßig praktischen Aspekte des SAT-Solvings. Behandelt werden:</p> <ul style="list-style-type: none"> - Grundlagen, historische Entwicklung - Codierungen, z.B. cardinality constraints - Phasenübergänge bei Zufallsproblemen - Lokale Suche (GSAT, WalkSAT, ..., ProbSAT) - Resolution, Davis-Putnam-Algorithmus, DPLL-Algorithmus, Look-Ahead-Algorithmus - Effiziente Implementierungen, Datenstrukturen - Heuristiken im DPLL-Algorithmus - CDCL-Algorithmus, Klausellernen, Implikationsgraphen - Restarts und Heuristiken im CDCL-Algorithmus - Preprocessing, Inprocessing - Generierung von Beweisen und deren Prüfung - Paralleles SAT-Solving (Guiding Paths, Portfolios, Cube-and-Conquer) - Fortgeschrittene Anwendungen: Bounded Model Checking, Planen, Satisfiability-Modulo-theories <p>Auch die Einbindung von industriellen Anwendern ist vorgesehen.</p>
Empfohlene Literatur	
Prüfungsleistungen	Modulprüfung
Anmerkungen	
Lehrveranstaltung SAT Solving Übung	
EDV-Bezeichnung	INFM212SE

Dozent/in	Prof. Dr. Carsten Sinz
Umfang	2.0 ECTS-Punkte, 1.0 SWS 60 Stunden gesamt, davon 15 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	In dieser Übung werden Verfahren der Vorlesung Practical SAT Solving anhand von Fragestellungen aus der Praxis erprobt und SAT-Solver zur Lösung von kombinatorischen Problemen eingesetzt.
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	

Modul Spezielle Kapitel Medieninformatik	
EDV-Bezeichnung	INFM220MI
Verantwortlich	Prof. Dr. Matthias Wölfel
Umfang	7.0 ECTS-Punkte, 5.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Das Modul "Spezielle Kapitel Medieninformatik" behandelt aktuelle Entwicklungen und spezielle Themen im Bereich der Medieninformatik. Die Inhalte des Moduls können von Semester zu Semester variieren, um auf die neuesten Trends und Entwicklungen in der Medieninformatik einzugehen. Das Modul nutzt verschiedene Lehrmethoden und Ressourcen, um den Studierenden ein umfassendes Verständnis für die aktuellen Themen und Entwicklungen in der Medieninformatik zu vermitteln. Dazu gehören Vorlesungen, Diskussionen über aktuelle Forschungsergebnisse und Anwendungen im Bereich der Medieninformatik.</p> <p>Das Modul zielt darauf ab, den Studierenden einen Einblick in aktuelle Entwicklungen und spezielle Themen der Medieninformatik zu geben und sie mit den neuesten Technologien, Methoden und Anwendungen vertraut zu machen. Durch die Auseinandersetzung mit aktuellen Themen sollen die Studierenden in die Lage versetzt werden, relevante Trends zu erkennen, innovative Lösungen zu entwickeln und kritisch zu reflektieren. Am Ende des Moduls sollen die Studierenden ein fundiertes Verständnis für die aktuellen Herausforderungen und Möglichkeiten in der Medieninformatik haben und in der Lage sein, ihr Wissen in praktischen Anwendungen umzusetzen.</p>
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Spezielle Kapitel Medieninformatik	
EDV-Bezeichnung	INFM221MI
Dozenten	Dr. Tim Schlippe Prof. Thomas Hinz Prof. Dr. Matthias Wölfel Prof. Daniel Schwarz
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	siehe allgemeine Modulbeschreibung, nicht die Veranstaltung
Empfohlene Literatur	
Prüfungsleistungen	Mündl. Prüfung/Entwurf 20/1 Min./Semester (benotet)
Anmerkungen	
Lehrveranstaltung Spezielle Kapitel Medieninformatik Übung	

EDV-Bezeichnung	INFM222MI
Dozenten	Prof. Daniel Schwarz Prof. Thomas Hinz Prof. Dr. Matthias Wölfel
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch
Inhalt	siehe allgemeine Modulbeschreibung, nicht die Veranstaltung
Empfohlene Literatur	
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	

Modul Spezielle Kapitel KI	
EDV-Bezeichnung	INFM220ML
Verantwortlich	Prof. Dr. Patrick Baier
Umfang	7.0 ECTS-Punkte, 5.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Das Modul "Spezielle Kapitel KI" behandelt aktuelle Entwicklungen und spezielle Themen im Bereich des Maschinellen Lernens und der Künstlichen Intelligenz. Die Inhalte des Moduls können von Semester zu Semester variieren, um auf die neuesten Trends und Entwicklungen in dem Bereich einzugehen.
Prüfungsleistungen	Klausur/mündl. Prüfung 120/20 Min. (benotet)
Lehrveranstaltung Spezielle Kapitel KI 1	
EDV-Bezeichnung	INFM221ML
Dozent/in	Dr. Patrick Baier
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	Eine Übersicht über aktuelle Veranstaltungen in diesem Bereich finden Sie hier: https://intranet.hka-iwi.de/info/compulsoryoptionalsubjects/INFM
Empfohlene Literatur	
Prüfungsleistungen	Klausur/mündl. Prüfung 60/20 Min. (benotet)
Anmerkungen	
Lehrveranstaltung Explainable AI	
EDV-Bezeichnung	INFM221ML.a
Dozent/in	Prof. Dr. Jannik Strötgen
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

<p>Inhalt</p>	<p>Diese Vorlesung befasst sich mit einem Teilgebiet der Künstlichen Intelligenz, Explainable AI (XAI). XAI-Methoden erfüllen eine kritische Funktion in der modernen KI-Landschaft, indem sie die Brücke zwischen menschlichen Nutzern und komplexen KI-Systemen schlagen.</p> <p>Nach der Teilnahme an der Vorlesung sind die Studierenden in der Lage, die vielfältigen Ziele von XAI zu beschreiben und mit der korrekten XAI-Terminologie zahlreiche Methoden aus dem Bereich der erklärbaren künstlichen Intelligenz einzuordnen und zu bewerten. Weiterhin sind Studierende in der Lage für dedizierte XAI-Methoden die zugrundeliegenden Ideen, Architekturen und Implikationen zu bewerten sowie Fachliteratur wie aktuelle Publikationen zu verstehen und Teilaspekte herauszustellen. Auch sind die Studierenden in der Lage durch praktische Übungen Methoden anzuwenden und Libraries einzusetzen.</p> <p>Inhaltlich werden in dieser Vorlesung folgende Themen abgedeckt. Dabei werden Methoden mit unterschiedlichem (i) Scope (local vs. global), (ii) Result (z.B. feature relevance, surrogate models), (iii) Functioning (z.B. perturbations, examples), (iv) Format (z.B. textual, visual) und (v) Stage (post-hoc vs. ante-hoc) abgedeckt:</p> <ul style="list-style-type: none"> - Motivation und Ziele von XAI - XAI-Terminologie und XAI Taxonomie - Methoden aus dem Bereich der Feature Importance Explanations (u.a. feature attribution vs. feature selection) - Methoden aus dem Bereich Removal-based Explanations (z.B. LIME, RISE) - Shapley Values (und SHAP) - Methoden aus dem Bereich Concept-based Explanations (z.B. Concept Bottleneck Models) - Methoden zur Interpretation neuronaler Netze (z.B. Attention-basierte Methoden) - Potentiellen Gefahren von XAI-Methoden etwa bezüglich "Fairwashing" und "Manipulation". <p>Am Ende der Vorlesung wird mittels einer mündlichen Prüfung überprüft, inwieweit die Studierenden die grundlegenden XAI-Konzepte verstanden haben sowie aktuelle Forschungsergebnisse bewerten können. Das Beantworten der Fragen erfordert sowohl die gelernte Terminologie richtig zu gebrauchen, als auch unterschiedliche Methoden analysieren bzw. teilweise auch evaluieren zu können.</p> <p>Neben den durch Gruppenarbeit nachvollzogenen und durch Kurzpräsentationen vorgestellten einzelnen Methoden, haben Studierende zusätzlich die Möglichkeit mit einer freiwilligen, größeren Gruppenpräsentation eines Spezialthemas zu demonstrieren, dass sie einzelne Themen detailliert bewerten und erklären können. Eine Präsentation, die durch fundierte Analyse und verständliche Darstellung überzeugt, führt zu einer Verbesserung der Prüfungsnote um einen Notenschritt.</p>
---------------	--

Empfohlene Literatur	- Vorlesungsfolien - Molnar, C. (2024). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (2nd ed.). christophm.github.io/interpretable-ml-book/
Prüfungsleistungen	Modulprüfung
Anmerkungen	
Lehrveranstaltung Spezielle Kapitel KI 2	
EDV-Bezeichnung	INFM222ML
Dozent/in	Dr. Patrick Baier
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	
Empfohlene Literatur	
Prüfungsleistungen	Klausur/mündl. Prüfung 60/20 Min. (benotet)
Anmerkungen	
Lehrveranstaltung Summer School Advanced Topics in AI	
EDV-Bezeichnung	INFM223ML
Dozent/in	Dr. Patrick Baier
Umfang	4.0 ECTS-Punkte, 3.0 SWS 120 Stunden gesamt, davon 45 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	
Empfohlene Literatur	
Prüfungsleistungen	Modulprüfung
Anmerkungen	
Lehrveranstaltung Winter School Advanced Topics in AI	
EDV-Bezeichnung	INFM224ML
Dozent/in	Dr. Patrick Baier
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	
Empfohlene Literatur	
Prüfungsleistungen	Modulprüfung
Anmerkungen	

Modul Spezielle Kapitel Software-Engineering	
EDV-Bezeichnung	INFM220SE
Verantwortlich	Prof. Dr. Dirk Hoffmann
Umfang	7.0 ECTS-Punkte, 5.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Das Modul "Spezielle Kapitel Software-Engineering" behandelt aktuelle Entwicklungen und spezielle Themen im Bereich des Software-Engineerings. Die Inhalte des Moduls können von Semester zu Semester variieren, um auf die neuesten Trends und Entwicklungen im Software-Engineering einzugehen.
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Codierungstheorie	
EDV-Bezeichnung	INFM221SE
Dozent/in	Prof. Dr. Dirk Hoffmann
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	Diese Vorlesung vermittelt das Grundwissen über Codes und Codierungen und die gängigen Algorithmen aus dem Gebiet der Quellen-, Kanal- und Leitungscodierung. Im Einzelnen werden Themen aus den folgenden Bereichen behandelt: Informations- und Codierungstheorie, Datenkompression, Fehlererkennende und -korrigierende Codes, Grenzen der Datenübertragung.
Empfohlene Literatur	- PowerPoint-Folien - Tafelmitschrift - Übungsblätter
Prüfungsleistungen	Klausur/mündl. Prüfung 60/20 Min./Semester (benotet)
Anmerkungen	Vorlesungsteilnahme
Lehrveranstaltung Modellierung und Simulation	
EDV-Bezeichnung	INFM222SE
Dozent/in	Prof. Dr. Britta Nestler
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

Inhalt	<p>Die Vorlesung gibt eine Einführung in Modellierungs- und Simulationsmethoden. Themen der Vorlesung und Übungen sind:</p> <ul style="list-style-type: none"> - Numerische Lösung von Nullstellenproblemen - Numerische Lösung linearer / nichtlinearer Gleichungssysteme - Approximationsverfahren: Taylorentwicklung, Polynominterpolation, Splines - Ausgleichsrechnung - Numerische Integration und Differentiation, Diskretisierungsverfahren, finite Differenzen - Anfangswertprobleme, dynamische Systeme, numerische Lösung gewöhnlicher Differentialgleichungen - Raum-Zeit-Probleme, Numerische Verfahren zur Lösung partieller Differentialgleichungen; Anwendung: Stoff- und Wärmetransport - Parallele Algorithmen und Standards zum verteilten Rechnen auf Hochleistungsrechnern
Empfohlene Literatur	<p>Die Inhalte der Vorlesung werden über Latex-Folien vermittelt. Die Folien werden den Studierenden vorlesungsbegleitend als PDF ins ILIAS hochgeladen. Ergänzend werden regelmäßig Beispiele und Anwendungen in vorlesungsintegrierten Rechenübungen besprochen. Die Aufgaben und Lösungen werden ebenfalls elektronisch bereitgestellt. Während der Veranstaltung werden ca. 6 Übungsblätter ausgeteilt, deren Lösung in darauffolgenden Terminen ausführlich vorgestellt wird. Zu der Veranstaltung gehört ein begleitendes Computerpraktikum, in dem numerische Algorithmen zu Interpolations- und Approximationsverfahren in kleinen Beispielprogrammen umgesetzt und am Rechner auf konkrete Probleme angewendet wird. Zum weiteren Selbststudium werden folgende Lehrbücher empfohlen:</p> <ul style="list-style-type: none"> - Scientific Computing, G. H. Golub and J.M. Ortega, B.G.Teubner Stuttgart 1996, ISBN 0-12-289255-0. - Numerische Mathematik, M. Knorrenschild, Fachbuchverlag Leipzig, Carl Hanser Verlag, ISBN 978-3-446-42228-5.
Prüfungsleistungen	Klausur/mündl. Prüfung 60/20 Min./Semester (benotet)
Anmerkungen	Seminaristischer Unterricht und Übungen
Lehrveranstaltung Modellierung und Simulation Übung	
EDV-Bezeichnung	INFM222SE
Dozent/in	Prof. Dr. Britta Nestler
Umfang	2.0 ECTS-Punkte, 1.0 SWS 60 Stunden gesamt, davon 15 Stunden Kontaktstudium.
Art/Modus	Übung
Lehrsprache	deutsch

Inhalt	<p>In dem begleitenden Rechnerpraktikum werden die Inhalte der Vorlesung "Modellierung und Simulation" vertieft, indem numerische Algorithmen zur Interpolation diskreter Datenmengen und zur Approximation von Lösungen für kontinuierliche Probleme in der Programmiersprache C/C++ implementiert werden. Zunächst werden die Iterationsverfahren in kleinen Beispielprogrammen umgesetzt. Diese werden auf konkrete Fragestellungen angewendet und die Lösungen diskutiert bzw. graphisch dargestellt. Im Anschluss werden ausgewählte numerische Methoden hinsichtlich Laufzeit analysiert und Konzepte der Parallelisierung eingesetzt, um die Iterationen parallel auszuführen oder durch Gebietszerlegung auf mehrere Prozessoren zu verteilen.</p> <p>Themen der Rechnerübung zur Vorlesung "Modellierung und Simulation" sind:</p> <ul style="list-style-type: none"> - Umsetzung der numerischen Algorithmen zur Lösung von Nullstellenproblemen, linearen / nichtlinearen Gleichungssystemen, Interpolationsverfahren (Polynominterpolation, Splines, Taylorreihen), Ausgleichsrechnung, Numerische Integration und Differentiation, dynamische Systeme, partielle Differentialgleichungen - Anwenden auf konkrete Fragestellungen - Rechenzeit- bzw. Speicheroptimierung der implementierten Programme durch Konzepte der Parallelisierung und des verteilten Rechnens auf Hochleistungsclustern
Empfohlene Literatur	<p>Für die praktischen Übungen am Rechner werden Aufgabenblätter erstellt und als PDF im ILIAS System bereitgestellt. Die Aufgaben werden zu Beginn der Veranstaltung besprochen, die Ziele erklärt und Lösungswege skizziert. Als Unterstützung werden den Studierenden Programmrümpfe zur Verfügung gestellt, in die die jeweiligen Algorithmen in C/C++ umgesetzt werden sollten. Nach Fertigstellung und Anwenden der Programme erfolgt eine Abnahme und eine ausführliche Besprechung der implementierten Lösung. Zum Vertiefen der in der Vorlesung erarbeiteten numerischen Verfahren wird auf das Lehrbuch:</p> <ul style="list-style-type: none"> - Numerische Mathematik, M. Knorrenschild, Fachbuchverlag Leipzig, Carl Hanser Verlag, ISBN 978-3-446-42228-5. <p>verwiesen. Als Unterstützung bei der Implementierung der Verfahren in C/C++ wird der Klassiker für Beispielprogramme in C empfohlen:</p> <ul style="list-style-type: none"> - Numerical Recipes in C book set: Numerical recipes . The art of scientific computing. Cambridge University Press; ISBN-10: 0521431085, ISSN-13: 978-0521431088
Prüfungsleistungen	Übung 1 Semester (nicht benotet)
Anmerkungen	Praktische Übungen am Rechner

Modul Mobile und Verteilte Systeme	
EDV-Bezeichnung	INFM230SE
Verantwortlich	Prof. Dr. Oliver Waldhorst
Umfang	7.0 ECTS-Punkte, 6.0 SWS
Einordnung	Alle Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	<p>Das Modul "Mobile und Verteilte Systeme" vermittelt fundiertes Wissen und praxisrelevante Fähigkeiten im Bereich mobiler und verteilter Systeme. Studierende entwickeln ein tiefgehendes Verständnis der zugrunde liegenden Technologien und Architekturen sowie die Kompetenz, moderne Lösungen für komplexe, reale Problemstellungen zu entwerfen, zu bewerten und anzuwenden.</p> <ul style="list-style-type: none"> - In der Vorlesung Mobile Systeme erwerben die Studierenden die Fähigkeit, mobile Kommunikationssysteme zu analysieren, deren Herausforderungen zu identifizieren und passende Technologien wie MANETs, DTNs oder mobiles TCP anzuwenden. - Die Vorlesung Verteilte Systeme befähigt die Studierenden, aktuelle Technologien wie Cloud Computing, IoT und DLT zu verstehen, zu evaluieren und in einem globalen Kontext anzuwenden. - Das Labor Verteilte Systeme bietet praktische Erfahrungen in der Entwicklung verteilter Systeme, insbesondere der Migration monolithischer Architekturen hin zu Microservices. Hierbei nutzen Studierende moderne Technologien wie Kubernetes, Docker und Istio. <p>Nach Abschluss des Moduls können Studierende technologische Konzepte mobiler und verteilter Systeme kritisch bewerten, deren Integration gestalten und innovative Lösungen im beruflichen und forschungsorientierten Kontext umsetzen.</p>
Prüfungsleistungen	Klausur/mündl. Prüfung 120/20 Min. (benotet)
Lehrveranstaltung Mobile Systeme	
EDV-Bezeichnung	INFM231.a
Dozent/in	Prof. Dr. Oliver Waldhorst
Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch

Inhalt	<p>In der Vorlesung „Mobile Systeme“ erwerben die Studierenden ein grundlegendes Verständnis der Herausforderungen und Technologien im Bereich mobiler Kommunikationssysteme. Nach Abschluss der Vorlesung können sie grundlegende Prinzipien drahtloser und mobiler Kommunikation erläutern und spezifische Technologien wie Mobilitätsmodelle, mobile Ad-hoc-Netze (MANETs), verzögerungstolerante Netze (DTNs) und mobiles TCP analysieren und bewerten. Darüber hinaus sind sie in der Lage, Problemstellungen im Bereich mobiler Systeme zu identifizieren und geeignete Lösungsansätze in praktischen Szenarien anzuwenden.</p> <p>Die Vorlesung behandelt die folgenden Inhalte:</p> <ul style="list-style-type: none"> • Grundlagen mobiler Systeme: Herausforderungen durch Mobilität, drahtlose Kommunikation und Netzwerke. • Mobilitätsmodelle: Simulation und Analyse individueller und gruppenbasierter Mobilität. • Mobile Ad-Hoc-Netze (MANETs): Selbstorganisierende Netzwerke, Routing-Protokolle und Anwendungsfälle. • Verzögerungstolerante Netze (DTNs): Kommunikation bei intermittierender Konnektivität und „Store-Carry-Forward“-Mechanismen. • Mobiles TCP: Anpassung und Optimierung des Transmission Control Protocols für mobile und drahtlose Netzwerke. <p>Die Vorlesung wird im Flipped Classroom-Format unterrichtet. Die Studierenden bereiten sich eigenständig mithilfe von Vorlesungsfolien und Erklärvideos auf die Live-Termine vor. In den Präsenzveranstaltungen werden Inhalte durch Fallstudien und Übungen vertieft. Zur Selbstüberprüfung stehen Online-Tests zur Verfügung, die den Studierenden Feedback und die Möglichkeit zur Vertiefung des Gelernten bieten. Die Prüfungsleistung besteht aus einer 60-minütigen Klausur, die Teil der Modulklausur „Mobile und Verteilte Systeme“ ist. Der Arbeitsaufwand beträgt insgesamt 60 Stunden, die sich auf 20 Stunden Präsenzzeit in den Live-Terminen, 20 Stunden asynchrones Lernen mit Vorlesungsfolien und Videos sowie 20 Stunden für Prüfungsvorbereitung und Nachbereitung verteilen.</p>
Empfohlene Literatur	<ul style="list-style-type: none"> - Foliensammlung und Erklärvideos im ILIAS-System - James Kurose, Keith Ross: Computer Networking - A Top-Down Approach, 8. Auflage, Pearson, 2021 (Kapitel 7) - Martin Sauter, Grundkurs Mobile Kommunikationssysteme, 8. Auflage, 2022 (als E-Book über die KIT-Bibliothek verfügbar) - Weitere Hinweise in ILIAS und in der Vorlesung
Prüfungsleistungen	Modulprüfung
Anmerkungen	Seminaristischer Unterricht, Übungsblätter
Lehrveranstaltung Verteilte Systeme	
EDV-Bezeichnung	INFM231SE
Dozent/in	Prof. Dr. Christian Zirpins

Umfang	2.0 ECTS-Punkte, 2.0 SWS 60 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Vorlesung
Lehrsprache	deutsch
Inhalt	<p>Aufbauend auf einem vorausgesetzten Verständnis grundlegender Prinzipien und Paradigmen verteilter Systeme behandelt die Master-Vorlesung Fallstudien aktueller Anwendungsgebiete. Die Auswahl behandelte Inhalte variiert. Es werden zum einen praktisch bedeutsame (industrierelevante) Bereiche berücksichtigt. Zum anderen werden aktuelle Trends aus Forschung und Entwicklung aufgegriffen. Die aktuelle Vorlesung fokussiert den Themenkomplex des Internet Computing.</p> <p>In der Vorlesung erlangen die Studierenden ein umfassendes Verständnis der Infrastruktur und der Technologien, die dem heutigen Internet-Computing zugrunde liegen. Sie entwickeln ein solides Verständnis für verteilte Systemarchitekturen und Webtechnologien, die für die Orientierung in der komplexen Landschaft globaler IT-Umgebungen unerlässlich sind. Durch die Betrachtung aktueller Paradigmen wie Cloud Computing und das Internet der Dinge (IoT) werden die Studierenden in die Lage versetzt, diese Technologien in verschiedenen fachlichen Kontexten zu bewerten und zu nutzen. Darüber hinaus werden die Studierenden durch die Behandlung neu aufkommender Technologien wie Distributed-Ledger-Technologien (DLT) und Fog Computing darauf vorbereitet, innovative Beiträge im Bereich des Internet-Computing zu entwickeln. Der Kurs zielt darauf ab, ihre analytischen Fähigkeiten zu verbessern und sie in die Lage zu versetzen, die Integration und das Potenzial internetbasierter Technologien bei der Gestaltung individueller, organisatorischer und gesellschaftlicher Praktiken kritisch zu bewerten. Darüber hinaus erweitern die Studierenden durch die Beschäftigung mit einer Vielzahl von Lernressourcen, einschließlich Beispielen, weiterführender Lektüre und Verständnisfragen im begleitenden Lehrbuch, ihre Forschungs- und Recherchefähigkeiten.</p>
Empfohlene Literatur	<p>Primäre Begleitliteratur:</p> <p>- Sunyaev, Ali. 2020. Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-34957-8.</p> <p>Während des Kurses werden eine umfangreiche Bibliographie und spezifische Empfehlungen für weiterführende Literatur vorgestellt.</p>
Prüfungsleistungen	Modulprüfung
Anmerkungen	Eigenständige Arbeitsanteile betreffen Vor- und Nacharbeit der Vorlesungsinhalte und Klausurvorbereitung.
Lehrveranstaltung Verteilte Systeme Labor	
EDV-Bezeichnung	INFM232SE
Dozent/in	Prof. Dr. Christian Zirpins
Umfang	3.0 ECTS-Punkte, 2.0 SWS 90 Stunden gesamt, davon 30 Stunden Kontaktstudium.
Art/Modus	Labor

Lehrsprache	deutsch
Inhalt	<p>Das Labor vermittelt praktische Einblicke in den Aufbau verteilter Informationssysteme. Es werden aktuelle Paradigmen aufgegriffen und erweiterte Prinzipien im Kontext realistischer Anwendungsfälle behandelt. Die konkreten Aufgabenstellungen orientieren sich an aktuellen Themen der industriellen Forschung und Entwicklung. Sie variieren daher von Semester zu Semester. Die praktische Umsetzung erfolgt mit modernen industrierelevanten Plattformen und Frameworks.</p> <p>Aktuell umfasst das Labor ein Projekt zur Migration eines monolithischen Informationssystems nach dem Microservice-Architekturstil. Dabei werden Technologien wie UML und Domain Driven Design, REST-basierte Microservices mit Spring/Spring-Boot, eine Microservice-Plattform mit Docker und Kubernetes sowie Service Meshes auf Basis von Istio eingesetzt.</p> <p>Durch Teilnahme an dem Labor werden die Studenten praktische Erfahrungen im Design, der Entwicklung und dem Einsatz von verteilten Informationssystemen sammeln, insbesondere durch die Umwandlung von monolithischen Architekturen in Microservices. Sie werden mit einer Reihe von modernen, branchenüblichen Tools und Technologien vertraut gemacht, darunter UML für die Modellierung, Domain-Driven Design für die Strukturierung von Systemen, Spring und Spring Boot für die Erstellung REST-basierter Microservices sowie Docker und Kubernetes für die Containerisierung und Orchestrierung. Durch den Einsatz von Istio zur Verwaltung von Service-Meshes werden die Studierenden zudem in die Lage versetzt, die Kommunikation und den Betrieb von Microservices in komplexen Systemen zu optimieren. Diese praktischen Erfahrungen bereiten die Studierenden auf aktuelle und zukünftige Herausforderungen in der industriellen Forschung und Entwicklung vor.</p>
Empfohlene Literatur	<ul style="list-style-type: none"> - S. Newman, "Microservices - Konzeption und Design", mitp, 2015 - E. Wolf, Das Microservices-Praxisbuch: Grundlagen, Konzepte und Rezepte, dpunkt, 2018, - B. Rumpe , Modellierung mit UML, Xpert.press, 2011 - V. Vernon, Domain-Driven Design kompakt, dpunkt, 2017 - E. Wolf, 2016, Microservices - Grundlagen flexibler Softwarearchitekturen, dpunkt, 2016 - E. Wolf, H. Prinz, Service Mesh – The New Infrastructure for Microservices, innoQ, 2020, http://leanpub.com/service-mesh-primer <p>Weitere Literatur wird in jedem Semester passend zur Aufgabenstellung vorgestellt. Hierzu zählen auch Online Tutorials basierend auf einer Auswahl aktueller Frameworks und Bibliotheken.</p>
Prüfungsleistungen	Laborarbeit 1 Semester (nicht benotet)
Anmerkungen	Es werden Grundkenntnisse in den Bereichen web- und komponentenbasierter verteilter Systeme sowie Web- und Datenbankprogrammierung in Java vorausgesetzt. Die Veranstaltung beinhaltet 50 % betreute Präsenzzeit (2 SWS) sowie 50% selbständige Arbeit. Der Leistungsnachweis erfolgt durch Präsentation und Verteidigung der Lösung.

Modul Projektbasiertes wissenschaftliches Arbeiten unter Anleitung 2	
EDV-Bezeichnung	INFM240
Verantwortlich	Prof. Dr. Thomas Fuchß
Umfang	5.0 ECTS-Punkte, 4.0 SWS
Einordnung	2. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, Techniken und Methoden des wissenschaftlichen und projektbasierten Arbeitens selbständig und reflektiert anzuwenden. Sie können ihre erworbenen Kenntnisse eigenständig und im Team auf komplexere und umfangreichere Aufgabenstellungen übertragen. Darüber hinaus haben sie gelernt, Entscheidungen kritisch zu hinterfragen, alternative Ansätze zu evaluieren sowie eigene Entscheidungen fundiert zu bewerten und zu rechtfertigen. Sie sind in der Lage, größere industrielle sowie wissenschaftliche IT-Projekte zu planen, durchzuführen und erfolgreich abzuschließen. Im Team übernehmen die Studierenden Verantwortung, arbeiten effektiv zusammen und tragen gemeinsam zum Projekterfolg bei.
Prüfungsleistungen	Mündliche Prüfung 20 Min. (benotet)
Lehrveranstaltung Projektarbeit 2	
EDV-Bezeichnung	INFM241
Dozent/in	Alle Professoren
Umfang	5.0 ECTS-Punkte, 4.0 SWS 150 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Praktische Arbeit
Lehrsprache	deutsch
Inhalt	Arbeitsgebiete können durch jeden Professor der Fakultät eröffnet werden, dies erfolgt durch Aushang zu Semesterbeginn. Forschungsprojekte bewegen sich an der vorderen Front der aktuellen Informatikforschung und können in Zusammenarbeit mit Forschungsinstitutionen durchgeführt werden. Anwendungsprojekte sind von besonderer Relevanz für die industrielle Praxis und können in Zusammenarbeit mit Industrieunternehmen durchgeführt werden.
Empfohlene Literatur	- Nach Maßgabe der Arbeitsgruppe
Prüfungsleistungen	Praktische Arbeit 1 Semester (benotet)
Anmerkungen	- Vorkenntnisse: Nach Maßgabe der Arbeitsgruppe - Format: Präsenzzeit mit Gruppendiskussion 30 %, selbständige Arbeit 70 %. - Betreuung: In der Regel wird wöchentlich zu einem festen Termin eine Sitzung der Arbeitsgruppe stattfinden.

Modul Hauptseminar	
EDV-Bezeichnung	INFM250
Verantwortlich	Prof. Dr. rer. pol. Mathias Philipp
Umfang	5.0 ECTS-Punkte, 4.0 SWS
Einordnung	2. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	
Prüfungsleistungen	Mündliche Prüfung 20 Min. (benotet)
Lehrveranstaltung Seminararbeit mit Präsentation	
EDV-Bezeichnung	INFM251
Dozent/in	Alle Professoren
Umfang	5.0 ECTS-Punkte, 4.0 SWS 150 Stunden gesamt, davon 60 Stunden Kontaktstudium.
Art/Modus	Abschlussarbeit
Lehrsprache	deutsch
Inhalt	
Empfohlene Literatur	
Prüfungsleistungen	Referat 1 Semester (benotet)
Anmerkungen	

Modul Abschlussarbeit mit Kolloquium	
EDV-Bezeichnung	INFM310
Verantwortlich	Prof. Dr. Heiko Körner
Umfang	30.0 ECTS-Punkte, 0.0 SWS
Einordnung	3. Semester
Inhaltliche Voraussetzungen	keine
Voraussetzungen nach SPO	keine
Kompetenzen	Die Master Thesis ist die Abschlussarbeit des Informatikstudiums. Die Teilnahme an diesem Modul befähigt die Studierenden, ein wissenschaftliches oder anwendungsnahes Thema weitgehend selbständig zu bearbeiten. Sie sind in der Lage, in Zusammenarbeit mit einem Industrieunternehmen oder einer Forschungsinstitution die Aufgabenstellung sorgfältig zu strukturieren, die erforderlichen Ressourcen zusammenzustellen und anschließend die Aufgabe anhand eines eigens abgeleiteten Zeitplans zu lösen. Sie können ihre Ergebnisse sowohl schriftlich dokumentieren als auch in einem Kolloquium vor einem Fachpublikum wissenschaftlich ansprechend präsentieren und verteidigen.
Prüfungsleistungen	Einzelprüfungen
Lehrveranstaltung Abschlussarbeit	
EDV-Bezeichnung	INFM311
Dozent/in	Alle Professoren
Umfang	29.0 ECTS-Punkte, 0.0 SWS 870 Stunden gesamt, davon 0 Stunden Kontaktstudium.
Art/Modus	Abschlussarbeit
Lehrsprache	deutsch
Inhalt	In der Abschlussarbeit bearbeiten die Studierenden in einem vorgegebenen Zeitraum eine praxisnahe Problemstellung oder eine Forschungsaufgabe selbstständig mit wissenschaftlichen Methoden und Erkenntnissen des Fachs. Sie strukturieren dazu die Aufgabenstellung, prüfen Abhängigkeiten, stellen die erforderlichen Ressourcen zusammen und bearbeiten das Problem an Hand eines Zeitplans. Die schriftliche Thesis fasst die Ergebnisse didaktisch sinnvoll aufbereitet zusammen und genügt wissenschaftlichen Standards.
Empfohlene Literatur	Passend zur Aufgabenstellung nach Absprache
Prüfungsleistungen	Master-Thesis 6 Monate (benotet)
Anmerkungen	
Lehrveranstaltung Verteidigung der Abschlussarbeit	
EDV-Bezeichnung	INFM312
Dozent/in	Alle Professoren
Umfang	1.0 ECTS-Punkte, 0.0 SWS 30 Stunden gesamt, davon 0 Stunden Kontaktstudium.

Art/Modus	Kolloquium
Lehrsprache	deutsch
Inhalt	Die Abschlussprüfung erfolgt über alle Informatik-relevanten Themen des Masterstudiengangs. Die Studierenden weisen nach, dass sie fachübergreifende Zusammenhänge verstanden haben und diese anwenden können. Sie präsentieren dazu die wichtigsten Ergebnisse ihrer Masterthesis didaktisch sinnvoll aufbereitet vor einem Fachpublikum. Außerdem stellen sie sich Fragen aus vielfältigen Bereichen der Informatik, die im Zusammenhang mit ihrer Abschlussarbeit stehen. Mit der Abschlussprüfung weisen sie nach, dass sie die Kompetenz zur selbstständigen Bearbeitung von neuartigen Problemstellungen aus der Informatik besitzen.
Empfohlene Literatur	
Prüfungsleistungen	Mündliche Prüfung 30 Min. (nicht benotet)
Anmerkungen	