

Modulhandbuch

»Wahlpflichtfächer«

Bachelorstudiengänge



Entwurf - Voraussichtlicher Veröffentlichungstermin: Oktober 2024

Die Modulbeschreibungen dienen der inhaltlichen Orientierung in Ihrem Studium.

Rechtlich verbindlich ist nur die jeweils geltende Studien- und Prüfungsordnung.

Inhaltsverzeichnis

1 Wahlpflichtfächer Bachelor - Angebot	4
1.1 Aktuelles Semester: Wintersemester 2024/25	4
1.2 Vergangenes Semester: Sommersemester 2024	6
2 Wahlpflichtfächer Bachelor - Übersicht	8
2.1 3D-Druck	8
2.2 ABAP-Grundlagen	12
2.3 Advanced Security Testing	16
2.4 Agile Entwicklung eines Klick-Dummy-Spiels	20
2.5 Agile Softwareentwicklung (Scrum)	24
2.6 Agile Webanwendungen mit Python	28
2.7 Betriebliche Informationssysteme	30
2.8 Chancen- & Risikomanagement in Digitalisierten Wertschöpfungsnetzen	32
2.9 Compiler	36
2.10 Computer Games Development	40
2.11 Corporate Entrepreneurship	42
2.12 Datenkommunikation	46
2.13 Datenkommunikation im Fahrzeug	48
2.14 Digitale Innovationen	52
2.15 Digital Business Leadership Skills	56
2.16 Digital Transformation in Organizations	60
2.17 Einführung in die maschinelle Sprachverarbeitung	64
2.18 Einführung in die Robotik	68
2.19 Elektronische Handelssysteme	70
2.20 Embedded Linux	72
2.21 Flugrobotik	74
2.22 Formula Student Driverless	76
2.23 Fullstack-Webentwicklung	80
2.24 Grundlagen DevOps	84
2.25 Hard- und Software für das Internet der Dinge	86
2.26 Hochschul Innovationsprojekt	88
2.27 Industrielle Bildverarbeitung	92
2.28 Industrielle Informationsverarbeitung	96
2.29 Informatik und Umwelt	100
2.30 Integrierte Geschäftsprozesse mit SAP ERP	104
2.31 Interaction Engineering	106
2.32 Interaktive Computergrafik	110
2.33 IT-Consulting	114
2.34 IT-Forensik	118
2.35 IT-Sicherheit	122
2.36 IT Sourcing and Cloud Transformation	126
2.37 Klassische Projekttechniken modernisiert	128

2.38 Konzepte der Datenbanktechnologie	132
2.39 Künstliche Intelligenz in sicherheitskritischen Anwendungen	136
2.40 Lean IT & Enterprise Architecture	140
2.41 Linux LPIC	144
2.42 Methoden der KI	148
2.43 Mobile Robotik	152
2.44 Mustererkennung und maschinelles Lernen	154
2.45 Network Penetration Testing	156
2.46 Neuronale Netze und Deep Learning	158
2.47 NoSQL	162
2.48 Objektorientierte Programmierung mit Python	164
2.49 Open-Source Softwareentwicklung	168
2.50 Praktische Robotik mit Matlab	172
2.51 Process Intelligence	176
2.52 Programmieren mit Datenbanken	178
2.53 Programmieren mit Python	180
2.54 Programmierung von Web-Anwendungen	184
2.55 Project Jupyter	188
2.56 Projekt - Forschung und Transfer	192
2.57 RFID und NFC Technik	196
2.58 Schwaben Innovation Masterclass	198
2.59 Service Learning Projekt	206
2.60 Softwaretechnik für datenintensive Systeme	210
2.61 Software-Projektmanagement	212
2.62 Startitup - Entrepreneurial Thinking and Business Design	214
2.63 Systemnahe Programmierung	218
2.64 Visual Thinking for Business	222

1 Wahlpflichtfächer Bachelor - Angebot

Diese Listen beinhalten nur die Wahlpflichtfächer, die an der Fakultät für Informatik angeboten werden.

Alle weiteren Fächer entnehmen Sie bitte den verantwortlichen Fakultäten.

1.1 Aktuelles Semester: Wintersemester 2024/25

Die nachfolgende Liste führt alle für Bachelor geeigneten Wahlpflichtfächer auf, die im WS2023/24 angeboten werden und nach Anmeldeschluss stattfinden.

Module	Creditpoints	Wochenstunden
ABAP-Grundlagen ¹	8 CP	6 SWS
Agile Softwareentwicklung (Scrum)	5 CP	4 SWS
Agile Webanwendungen mit Python	5 CP	4 SWS
Corporate Entrepreneurship	5 CP	4 SWS
Datenkommunikation im Fahrzeug	5 CP	4 SWS
Digital Business Leadership Skills	8 CP	6 SWS
Elektronische Handelssysteme	5 CP	4 SWS
Flugrobotik	5 CP	4 SWS
Hard- und Software für das Internet der Dinge	5 CP	4 SWS
Hochschul Innovationsprojekt	5 CP	4 SWS
Integrierte Geschäftsprozesse mit SAP ERP	5 CP	4 SWS
Interaction Engineering ²	5 CP	4 CH
IT-Forensik	5 CP	4 SWS
IT-Sicherheit	5 CP	4 SWS
Klassische Projekttechniken modernisiert ⁵	5 CP	4 SWS
Konzepte der Datenbanktechnologie	5 CP	4 SWS
Künstliche Intelligenz in sicherheitskritischen Anwendungen	5 CP	4 SWS
Mobile Robotik	5 CP	2 SWS
Mustererkennung und maschinelles Lernen	5 CP	4 SWS
Process Intelligence	5 CP	4 SWS
Programmieren mit Datenbanken	5 CP	4 SWS
Projekt - Forschung und Transfer	10 CP	8 SWS
Service Learning Projekt	5 CP	4 SWS

Software-Projektmanagement	5 CP	4 SWS
Systemnahe Programmierung ⁴	6 CP	5 SWS

(1) WPF nur für IN, TI und IA. Für WI, IIS handelt es sich um ein Pflichtfach (WI: Programmieren 3, IIS: Programming of Information Systems).

(2) WPF nur für IN, TI.

(3) WPF nur für WI, IIS, TI und IA. Für IN handelt es sich um ein Pflichtfach (Programmieren 3).

(4) WPF nur für TI. Für IN handelt es sich um ein Pflichtfach.

(5) WPF nur für IN, WI, IIS und TI. Nicht für IA.

Blockveranstaltungen

Module	Creditpoints	Wochenstunden
3D-Druck	5 CP	4 SWS
Computer Games Development	5 CP	4 CH
Linux LPIC	5 CP	4 SWS
Praktische Robotik mit Matlab	7.5 CP	6 SWS
Schwaben Innovation Masterclass ⁵	10 CP	8 SWS
Visual Thinking for Business	5 CP	4 CH

(5) WPF zweisemestrig

1.2 Vergangenes Semester: Sommersemester 2024

Module	Creditpoints	Wochenstunden
Agile Entwicklung eines Klick-Dummy-Spiels	5 CP	4 SWS
Agile Softwareentwicklung (Scrum)	5 CP	4 SWS
Agile Webanwendungen mit Python	5 CP	4 SWS
Betriebliche Informationssysteme	5 CP	4 SWS
DB-Anwendungen ¹	3 CP	3 SWS
Datenkommunikation ²	5 CP	4 SWS
Digital Business Leadership Skills	7.5 CP	6 SWS
E-Commerce	7.5 CP	6 SWS
Einführung in die maschinelle Sprachverarbeitung	5 CP	4 SWS
Einführung in die Robotik	5 CP	4 SWS
Embedded Linux	7.5 CP	6 SWS
Formula Student Driverless	5 CP	4 SWS
Fullstack-Webentwicklung	8 CP	6 SWS
Grundlagen DevOps	5 CP	4 SWS
Hochschul Innovationsprojekt	5 CP	4 SWS
Industrielle Bildverarbeitung	5 CP	4 SWS
Industrielle Informationsverarbeitung	5 CP	4 SWS
Interaktive Computergrafik	7.5 CP	6 SWS
IT-Sicherheit	7.5 CP	6 SWS
IT Sourcing and Cloud Transformation	5 CP	4 CH
Klassische Projekttechniken modernisiert	5 CP	4 SWS
Methoden der KI	7.5 CP	6 SWS
Network Penetration Testing	5 CP	4 SWS
Neuronale Netze und Deep Learning	5 CP	4 SWS
NoSQL	5 CP	4 SWS
Objektorientierte Programmierung mit Python	5 CP	4 SWS
Open-Source Softwareentwicklung	5 CP	4 SWS
Programmierung von Web-Anwendungen	5 CP	4 SWS
Projekt - Forschung und Transfer	10 CP	8 SWS
Service Learning Projekt	5 CP	4 SWS

Startitup - Entrepreneurial Thinking and Business Design	5 CP	4 SWS
--	------	-------

(1) WPF nur für IN, IIS. Pflichtfach für WI.

(1) WPF nur für WI, IIS. Pflichtfach für IN, TI.

Blockveranstaltungen

Module	Creditpoints	Wochenstunden
Advanced Security Testing	5 CP	4 SWS
Chancen- & Risikomanagement in Digitalisierten Wertschöpfungsnetzen	2.5 CP	2 SWS
Computer Games Development	5 CP	4 CH
Linux LPIC	2.5 CP	2 SWS
Linux LPIC Advanced	2.5 CP	2 SWS
Suchmaschinenoptimierung (SEO)	2.5 CP	2 SWS
Visual Thinking for Business	5 CP	4 CH

2 Wahlpflichtfächer Bachelor - Übersicht

2.1 3D-Druck

Informationen über das Modul

Name / engl.	3D-Druck / 3D printing
Kürzel	3DDR4.WP
Verantwortlicher	Prof. Dr. Jürgen Scholz
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	3D-Druck (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum die Gruppenarbeit und das Selbststudium.
Voraussetzungen	Kenntnisse Programmierung
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Klausur, 60 Minuten, keine Hilfsmittel, 25%• Studienarbeit, 10-25 Seiten, mit Ergebnispräsentation, 5-10 Minuten, 75%
Prüfungsnummer	IN -, - TI - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Abstimmung des Vorlesungsinhalts mit Vorkenntnissen / Interessen der Teilnehmer

- Einstieg
 - Was ist 3D-Druck?
 - Anwendungsgebiete
- Arten von Druckern
 - Welche gibt es und wie funktionieren sie (FDM, SLA, DLP, SLS, MSLA, BJ, MJet, EBM)
 - Für was benutzt man welche Technologie (auch Kosten für Arten angeben)
 - Industriedrucker vs Consumerdrucker
- FDM Drucker
 - Was für Unterschiede gibt es (bewegliche Achsen, delta, Direct Drive, Multi nozzle, kammer, Druckbett,
 - Nozzle Arten, etc.)
 - welche Hersteller gibt es, was sind Unterschiede
- Materialien
 - Welche Materialien gibt es und was sind Besonderheiten
- Slicer
 - Warum braucht man das
 - Wie funktioniert der (nur kurz ansprechen)
 - Welche Einstellungen machen was
 - Beispiele (verschiedene Slicer, Beispielobjekte)
 - Typische Probleme besprechen
- Nutzung von Druckern (FDM)
 - Objekt erstellen (herunterladen)
 - Slicer bedienen
 - Druck starten
 - Drucker richtig einstellen
 - Drucker warten
- Verschiedene Firmware
 - Marlin
 - Reprap

- Klipper
- Zukunftstechnologien/neue Ansätze
 - aktuelle Entwicklungen zur Beschleunigung des Druckens
 - Fließband Druckbett
 - 4/5 achsen 3D-Drucker
 - Variable Size Nozzle
 - Nozzle Extruder
 - ...
- Praxisteil
 - Zusammenbau eines 3D Druckers in Gruppenarbeit - alternativ Druck von 3D Objekten
 - Projekte an und mit 3D Druckern

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden verstehen die Funktionsweise eines (FDM) 3D Druckers. Sie lernen diese einzurichten, zu justieren und bedienen. Weiterhin lernen sie einfache 3D Objekte zu erstellen und auszudrucken.

Nach erfolgreichem Abschluss dieses Moduls sind sie berechtigt, die 3D-Drucker der Fakultät zu benutzen.

Literaturliste

Wird in der ersten Präsenzveranstaltung bekannt gegeben.

2.2 ABAP-Grundlagen

Informationen über das Modul

Name / engl.	ABAP-Grundlagen / Fundamentals of ABAP
Kürzel	ABAPGL6.WP
Verantwortlicher	Prof. Dr. Stefan Bensch Dipl.-Wirt.-Inf. (FH) Christian Herdin, M.Sc.
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	ABAP-Grundlagen (4 SWS) zugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum das Selbststudium.
Voraussetzungen	Softwareentwicklung und Programmierung 1 und 2
Verwendbarkeit	WPF nur für die Bachelorstudiengänge Informatik, Technische Informatik und Interaktive Medien. Für Wirtschaftsinformatik und International Information Systems handelt es sich um ein Pflichtfach (WI: Programmieren 3, IIS: Programming of Information Systems)
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Elektronische Prüfung, 90 Minuten, Hilfsmittel: Skript, SAP Software, Office-Anwendungen zur Text- und Datenverarbeitung
Prüfungsnummer	IN 3970339, 2970830 TI 3976559, 2976662
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Grundlagen der Programmierung betrieblicher Anwendungssysteme:

- Technische Basis
- Grundlagen in ABAP
- Klassisches Reporting, klassische Ereignisse zur Seitengestaltung und interaktives Reporting
- Datentypen (Variablen und Konstanten) und Programmierstrukturen
- Entscheidungen
- Wiederholungen
- Felder und Zeichenketten
- Funktionen
- Komplexe Datentypen

Fortgeschrittene Programmierung

- Objektorientiertes Reporting mit ABAP Objects
- Referenzen und Speichermanagement
- Events
- Interfaces
- Vererbung
- Fehlerbehandlung
- Globale Klassen
- Weiterführende Programmiertechniken

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Schlüsselwörter der Programmiersprache ABAP und deren Funktion beschreiben.
- Quellcode verstehen.
- Vorgegebene Algorithmen selbstständig und effizient implementieren.
- Einfache Algorithmen selbst entwickeln.
- Anforderungen selbstständig implementieren.

Literaturliste

Keller, Horst und Sascha Krüger. ABAP Objects: ABAP-Programmierung mit SAP Net-Weaver. 3. Aufl. Bonn: SAP PRESS, 2006.

Roth, Felix. ABAP Objects: Das neue umfassende Handbuch zu Konzepten, Sprachelementen und Werkzeugen in ABAP OO. 1. Aufl. Bonn: SAP PRESS, 2016.

Schwaiger, Roland. Schrödinger programmiert ABAP: Das etwas andere Fachbuch - Dein unterhaltsamer Einstieg in ABAP. 2. Aufl. Bonn: SAP PRESS, 2014.

„SAP ERP - SAP Help Portal“. Help Portal. SAP Help Portal SAP ERP. Zugriffen 8. März 2019. https://help.sap.com/viewer/p/SAP_ERP.

2.3 Advanced Security Testing

Informationen über das Modul

Name / engl.	Advanced Security Testing / Advanced Security Testing
Kürzel	AST4.WP
Verantwortlicher	Dr.-Ing. Matthias Niedermaier
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Advanced Security Testing (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Integrierte Vorlesung
Voraussetzungen	Kenntnisse in IT Sicherheit unabdingbar
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Klausur, 60 Minuten, keine Hilfsmittel, 50%• Präsentation, 20-30 Minuten, 50%
Prüfungsnummer	IN 3970372, 2970870 TI 3976568, 2976681 WI 3975790 IIS 9775100
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Standards bei Security Tests
- Berichterstellung
- Verwenden von Tools
- Auszug nicht komplett: Nessus, OpenVAS, Metasploit, binwalk, Firmwaremodification kit, ZAP, Checkstyle, CCP Check, burp suite
- Erstellung eigener Skripte um aktuelle IT-Sicherheitsaspekte zu beleuchten
- Vorgehen bei Softwaretests
- Vorgehen bei Produkttests / Hardwaretests
- Vorgehen beim Testen von IT Landschaften
- Aktueller Stand von Technik und Forschung in Bezug auf IT-Sicherheit wird vermittelt

Vorgehen

- Im Rahmen der Vorlesung werden folgende Themenbereiche der IT-Sicherheit beleuchtet:
 - Netzwerksicherheit
 - Hardwaretests
 - Softwaretestmethoden
- Es werden Schwachstellen und Schutzmaßnahmen praktisch an aktuellen Geräten und Software durchgeführt
- Die Studierenden müssen in Projektgruppen eine wissenschaftliche Fragestellung bearbeiten, hier werden Themenfelder vertieft und der Stand der Forschung aufgegriffen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Kenntnisse:

- In der Vorlesung soll mit praxisnahen Fragestellungen die Planung, das Vorgehen und der Abschluss von Security Tests besprochen werden. Um die Vorlesung möglichst nahe an der beruflichen Praxis zu halten, wird ein vielfältiges Spektrum an Tools/Werkzeugen verwendet.
- Es wird Wert auf eine möglichst breite Themenvielfalt in diesem Bereich gelegt. Das Aufspüren von Softwareschwachstellen im Source Code, Testen von ganzen Netzwerken sowie Hardwarenahe Fragestellungen gehören dazu.

Fertigkeiten:

- Durchführen von klassischen Security Produkttests
- Durchführen von Netzwerksicherheitstests
- Angriffe und Verteidigung auf Hardware
- Durchführen von Softwaretests

Kompetenzen:

- Die Studierenden können Penetrationstests u.a. mit Hilfe von Tools durchführen
- Sie können sich in neue Thematiken im Rahmen von Sicheren Architekturen einarbeiten
- Studierende sind in der Lage Produkte grundlegend auf ihr IT-Sicherheitsniveau zu prüfen

Literaturliste

HUANG, Andrew Bunnie. The Hardware Hacker: Adventures in Making and Breaking Hardware. 2017.

HUANG, Andrew. Hacking the Xbox: An Introduction to Reverse Engineering. 2002.

ERICKSON, Jon. Hacking: The Art of Exploitation. No Starch Press, 2008.

Script

2.4 Agile Entwicklung eines Klick-Dummy-Spiels

Informationen über das Modul

Name / engl.	Agile Entwicklung eines Klick-Dummy-Spiels / Agile development of a Click-Dummy Game
Kürzel	AEKDS4.WP
Verantwortlicher	Matthias Regner, M.Eng.
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Wintersemester angeboten. Außerplanmäßig manchmal im Sommersemester.
Lehrveranstaltungen	Agile Entwicklung eines Klick-Dummy-Spiels (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht mit Theorie- und Praxisteilen. Die Entwicklung des Klick-Dummys wird als Projektarbeit in Gruppen erfolgen.
Voraussetzungen	Grundkenntnisse der Softwareentwicklung
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Studienarbeit, 20-30 Seiten, 90%• Präsentation, 30 Minuten, 10%
Prüfungsnummer	IN 3970391, 2970889 TI 3976569, 2976711 WI 3975809 IIS 9775163
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Modul vermittelt den Teilnehmenden Wissen, um IT-Projekte nach agilen Projektmethoden zu planen, aufzusetzen und durchzuführen. Im Fokus liegt die praktische Anwendung der Scrum-Methode im Rahmen eines Gruppenprojekts. Jede Woche wird es kurze Theorieeinheiten geben, die das Projekt mit neuen agilen Elementen anreichern, bis zum Schluss ein vollwertiger Scrum-Prozess durchlaufen wird.

Optional wird es eine Exkursion nach München für einen Scrum-Minecraft Workshop geben.

Theorieeinheiten:

- Grundlagen für agile Konzepte und Scrum
- Entwicklung von Produktvisionen
- Schätztechniken in agilen Projekten
- Kanban & Scrumban
- Extreme Programming
- Qualitätsmanagement in agilen Projekten
- Skalierte Scrum Frameworks (SAFe, Less, Nexus, ...)
- Kennenlernen von Prototyping Tools
- Schrittweise Einführung neuer Scrum-Elemente

Gruppenprojekt:

- Benutzung eines Prototyping Tools (z.B. Figma, Adobe XD, ...)
- Anwendung von Scrum, um einen Klick-Dummy für ein digitales Spiel zu entwickeln
- Abhalten von regelmäßigen Reviews und Retrospektiven
- Planung eines Sprints mit Hilfe digitaler Tools
- ToDos in Form von User Stories im Product Backlog erfassen
- Führen eines Sprint Backlogs während der Entwicklung

Die Anwendung der Scrum Methode steht ganz klar im Vordergrund des Moduls. Der Klick-Dummy dient dabei nur als Anschauungsobjekt. Es ist nicht das Ziel, den besten Klick-Dummy zu entwickeln, sondern erste Erfahrungen mit Scrum-Projekten in der Praxis zu sammeln und zu reflektieren.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Den Nutzen agiler Methoden in Projekten zu beurteilen
- Agile Projektmethoden auszuwählen, ihren Einsatz zu planen und anzuwenden
- Die Scrum-Methode in Projekten praktisch anzuwenden
- Erfahrungen mit Scrum zu sammeln und zu reflektieren
- Methoden für die Aufwandsschätzung im agilen und nicht-agilen Setup auszuwählen und anzuwenden
- Sprintplanning und Backlog-Refinements durchzuführen
- Verschiedene Priorisierungsstrategien (Kosten/Nutzen, Bedürfnisse) beim Backlog Management anzuwenden und zu kombinieren
- Skalierte agile Frameworks zu nennen
- Die Funktionsweise und den Nutzen des Scaled Agile Frameworks zu erklären

Literaturliste

Literatur wird in der Veranstaltung bekannt gegeben.

2.5 Agile Softwareentwicklung (Scrum)

Informationen über das Modul

Name / engl.	Agile Softwareentwicklung (Scrum) / Agile Software Development (Scrum)
Kürzel	SCRUM4.WP, SCRUMZ.WP
Verantwortlicher	Dipl.-Inf. (FH) Gregor Liebermann, M.Sc.
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig jedes Semester angeboten.
Lehrveranstaltungen	Agile Softwareentwicklung (Scrum) (2 SWS) Praktikum Agile Softwareentwicklung (Scrum) (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht
Voraussetzungen	Erste Erfahrungen in Programmierung und Anforderungsanalyse
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Zulassungsvoraussetzung zur Prüfung	Praktikum Agile Softwareentwicklung (Scrum)
Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970323, 2970791 TI 3976560, 2976565 WI 3975711 IIS 9775102
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Grundlagen:

- Klassische und agile Entwicklungsmethoden
- Agiles Manifest
- Iteratives Vorgehen

Scrum:

- Grundlagen und Motivation
- Anforderungsmanagement
- Rollen und Meetings
- Sprints und Vorgehen
- Releaseplanung

Das Team:

- Phasen der Teamentwicklung
- Persönlichkeitsprofile
- Kommunikation und Vier-Seiten-Modell
- Teambuilding

Scrum Tools und Praxis:

- Scrum in der Praxis und mögliche Probleme
- Continuous Integration
- Pair Programming
- CVS und SVN
- Bugtracking
- Review Tools
- Digital Taskboards

Weitere Agile Methoden:

- Extreme Programming
- Crystal
- FDD
- Exkurs: Kanban
- Exkurs: Design Thinking

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden kennen die Vor- und Nachteile agiler Entwicklungsmethoden im Vergleich zu klassischen Vorgehensmodellen und können einschätzen, welche Methodik für welches Projekt geeignet ist und welche nicht. Die Grundlagen von Scrum wurden praxisnah erlernt.

Gewichtung der Einzelleistung in der Gesamtnote

Zum Bestehen müssen beide Teilleistungen erfolgreich absolviert werden.

Literaturliste

Wird in der ersten Präsenzveranstaltung bekannt gegeben.

2.6 Agile Webanwendungen mit Python

Informationen über das Modul

Name / engl.	Agile Webanwendungen mit Python / Agile Web Applications with Python
Kürzel	PYTHON4.WP
Verantwortlicher	Dipl.-Inf. (FH) Dipl.-Designer (FH) Erich Seifert, MA
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird unregelmäßig bzw. auf Nachfrage angeboten.
Lehrveranstaltungen	Agile Webanwendungen mit Python (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht zur theoretischen Wissensvermittlung, praktische Umsetzung der Studienarbeit in Gruppen ergänzt die Vorlesung und fördert die Teamarbeit sowie das Selbststudium, der schriftliche Teil der Studienarbeit vermittelt die Fähigkeit zur Bewertung der gewonnenen Kenntnisse.
Voraussetzungen	Programmierung mit Python
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Studienarbeit, 10-25 Seiten, 80%• Präsentation, 10-25 Minuten, 20%
Prüfungsnummer	IN 3970329, 2970801 TI 3976562, 2976573 WI 3975721 IIS 9775104
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Agile Entwicklungsmethoden
- Test Driven Development
- Webtechnologien (HTML, CSS, JavaScript)
- Softwarearchitektur für Webanwendungen
- Einführung in verschiedene Python-Frameworks für die Webentwicklung

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden können verschiedene Frameworks zur Webentwicklung beurteilen und können sie passend zu eigenen Projekten auswählen. Agile Entwicklungstechniken im Web-Umfeld werden verstanden und können angewendet werden.

Literaturliste

Wird in der ersten Präsenzveranstaltung bekannt gegeben.

2.7 Betriebliche Informationssysteme

Informationen über das Modul

Name / engl.	Betriebliche Informationssysteme / Business Information Systems
Kürzel	BEINF4.WP
Verantwortlicher	Prof. Dr. Stefan Bensch
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Betriebliche Informationssysteme (2 SWS) Betriebliche Informationssysteme Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum das Selbststudium.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Projektarbeit
Prüfungsnummer	IN 3970386, 2970884 TI 3976571, 2976706 WI 3975804 IIS 9775164
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Einführung zum Thema Betriebliche Informationssysteme: Operative und analytische Systeme (Business Intelligence)
- Überblick zum Thema: Betriebliche Informationssysteme, Architekturen und Entwicklungen
- Methoden und Strukturierungsansätze zu Analyse, Design und Entwicklung
- Transfer auf praktische Anwendungsbeispiele: Entwicklung und Anwendung im Projekt.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Grundlegende Konzepte, Lösungen und Anwendungsbereiche Betrieblicher Informationssysteme zu kennen und zu verstehen
- Konzepte und Lösungen erfolgreich anzuwenden
- Praktische Übungen und Simulationen durchzuführen (mit SAP S/4HANA, SAP Analytics Cloud und SAP Datasphere)
- Aktuelles Wissen und den Stand der Forschung zu Betrieblichen Informationssystemen selbständig zu erarbeiten

Literaturliste

Aßmann, Dietz, Japing, Jensen, Kästner, Rose, Scivos: ABAP Objects: SAP Data Warehouse Cloud, Rheinwerk Publishing SAP PRESS, 2023

Baars, H., Kemper, HG.: Business Intelligence & Analytics – Grundlagen und praktische Anwendungen: Ansätze der IT-basierten Entscheidungsunterstützung. Springer Vieweg, Wiesbaden, 2021

Gluchowski, P., Chamoni, P. (Hrsg.): Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen, 5. Aufl., Springer Gabler, Berlin, 2016

Kemper, H.-G., Baars, H., Mehanna, W.: ABAP Objects: Business Intelligence – Grundlagen und praktische Anwendungen: Eine Einführung in die IT-basierte Managementunterstützung, 4. Aufl., Springer Vieweg, Wiesbaden, 2016

Laudon, Kenneth C., Jane P. Laudon und Detlef Schoder: Wirtschaftsinformatik: Eine Einführung. 3. Aufl. Hallbergmoos/Germany: Pearson Studium, 2015

Roth, Felix: ABAP Objects: Das neue umfassende Handbuch zu Konzepten, Sprachelementen und Werkzeugen in ABAP OO. 1. Aufl. Bonn: SAP PRESS, 2016

2.8 Chancen- & Risikomanagement in Digitalisierten Wertschöpfungsnetzen

Informationen über das Modul

Name / engl.	Chancen- & Risikomanagement in Digitalisierten Wertschöpfungsnetzen / Chance and Risk Management in Digitized Value Networks
Kürzel	CHRIMG4.WP
Verantwortlicher	Prof. Dr. Björn Steven Häckel
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester; regelmäßig sowohl im Wintersemester als auch im Sommersemester
Lehrveranstaltungen	Chancen- & Risikomanagement in Digitalisierten Wertschöpfungsnetzen (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Vorlesung mit integrierten Übungen und Diskussion praxisnaher Fallstudien.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Klausur, 60 Minuten, Hilfsmittel: nicht-prog. Taschenrechner, 50%• Studienarbeit, 5-10 Seiten, 25%• Präsentation, 10-15 Minuten, 25%
Prüfungsnummer	IN -, - TI -, - WI - IIS -

Inhalte des Moduls

Getrieben durch Entwicklungen wie die zunehmende Globalisierung oder die fortschreitende Digitalisierung im Zuge der Industrie 4.0 und disruptiven Technologien wie KI unterliegen industrielle Wertschöpfungsnetze einem stetigen Wandel. Aus diesen Entwicklungen resultieren für die beteiligten Unternehmen zum einen vielversprechende Chancen wie etwa die Erschließung neuer Märkte, die Entwicklung neuer, datenbasierter Produkte und Dienstleistungen oder die flexiblere Fertigung von Produkten. Andererseits stellen die Entwicklungen aufgrund der komplexen und oftmals intransparenten Abhängigkeitsbeziehungen in vernetzten Wertschöpfungssystemen auch erhebliche Risiken für Unternehmen dar. Um den Studierenden einen Überblick über die daraus resultierenden Herausforderungen für die Unternehmenssteuerung zu geben, beschäftigt sich die Veranstaltung unter anderem mit den folgenden Fragen:

- Was sind zentrale Charakteristika von Wertschöpfungsnetzen?
- Welche Faktoren tragen zur Transformation von Wertschöpfungsnetzen bei?
- Welche Chancen und Risiken entstehen aus der zunehmenden Vernetzung?
- Wie können Wertschöpfungsnetze modelliert und analysiert werden?
- Welchen Einfluss hat KI auf Wertschöpfungsnetze?

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Die grundlegenden Elemente von und Einflussfaktoren auf digitalisierte Wertschöpfungsnetze zu erläutern.
- Die Chancen und Risiken in digitalisierten Wertschöpfungsnetzen zu charakterisieren.
- Zentrale Herausforderungen im Rahmen der digitalen Transformation von Geschäftsmodellen in Wertschöpfungssystemen zu skizzieren.
- Die Implikationen der Digitalisierung auf unternehmerische Geschäftsmodelle und die Struktur von Wertschöpfungssystemen zu beurteilen.
- Die Abhängigkeitsstrukturen in komplexen Wertschöpfungssystemen mit ausgewählten Methoden zu analysieren.
- Ausgewählte Verfahren (insbesondere Zentralitätsmaße) zur Bestimmung der Kritikalität einzelner Akteure in Wertschöpfungsnetzen anzuwenden.

Literaturliste

- Broy, M., (2011):** Integrierte Forschungsagenda Cyber-Physical Systems – Vortrag Embedded Systems, Symposium 2011.
- Fraunhofer IAO (2013):** Produktionsarbeit der Zukunft – Industrie 4.0.
- Sydow, J. (1992):** Strategische Netzwerke. Evolution und Organisation, Wiesbaden.
- Fleisch E., Weinberger M., Wortmann F. (2015):** Geschäftsmodelle im Internet der Dinge. Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung 67:444 - 465.
- Schneider, S. und Spieth, P. (2013):** Business Model Innovation - Towards an Integrated Future Research Agenda. International Journal of Innovation Management, Vol. 17, No. 1.
- Osterwalder, A. und Pigneur, Y. (2011):** Business Model Generation; Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt, New York.
- Brandes, U. und Erlebach, T. (Hrsg.) (2005):** Network Analysis – Methodological Foundations, Band 3418 der Reihe Lecture Notes in Computer Science. Springer, 2005.
- Fridgen, G. und Garizy, Tira Zare (2015):** Supply Chain Network Risk Analysis“, Veranstaltungsbeitrag, 23rd European Conference on Information Systems (ECIS) , 26.-29.05.2015, Münster, Germany.

2.9 Compiler

Informationen über das Modul

Name / engl.	Compiler / Compiler
Kürzel	COM4.WP
Verantwortlicher	Prof. Dr. Rolf Winter
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird unregelmäßig bzw. auf Nachfrage angeboten.
Lehrveranstaltungen	Compiler (3 SWS) zugehöriges Praktikum (1 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum die Gruppenarbeit und das Selbststudium.
Voraussetzungen	Solide Kenntnisse einer höheren Programmiersprache wie JAVA oder C / C++
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: 1 DIN-A4-Seite handgeschrieben
Prüfungsnummer	IN 3970320, 2970776 TI 3976573, 2976515 WI 3975696 IIS 9775107
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Wie oft wird ein mehr oder weniger kleiner Scanner oder Parser benötigt? Häufig wird abenteuerlich auf selbst "erfundene" Scanner und Parser gesetzt. Um hier rechtzeitig den Absprung von ein paar Zeilen Code zu erleichtern, ist die Kenntnis des Aufbaus und der Wirkungsweise von Compilern wichtig.

In dieser Vorlesung wird die Funktionsweise und der von Parsern über Scanner bis zu Compilern erarbeitet. Hierbei wird der sinnvolle Einsatz von Werkzeugen basierend auf den theoretischen Grundlagen beschrieben.

Zunächst werden die theoretischen Grundlagen des Compilerbaus - die formalen Sprachen und die Automaten - erarbeitet. Hier wird ein Schwerpunkt auf CH-2 und CH-3 Sprachen gesetzt, die für Compiler besonders relevant sind. Aufbauend auf der Theorie wird dann die praktische Realisierung des Übersetzerbaus besprochen. Der Weg führt zur Konstruktion von Programmen zur lexikalischen und syntaktischen Analyse. Deren konkrete Realisierung wird an Hand allgemein verwendeter Programme veranschaulicht. Hierbei wird ein Compiler mit Hilfe gängiger Werkzeuge erstellt.

- Einführung
- Sprachentheorie: Grundlagen der Sprachanalyse
- Lexikalische Analyse
- lex / flex
- Die Syntaxanalyse
- Semantische Analyse
- Der Compiler-Generator YACC / BISON
- Zwischencodeerzeugung
- Code-Optimierung

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- zu erkennen wo die Einsatzgebiete von Scannern und Parsern liegen.
- die Funktion und Wirkungsweise von Scannern und Parsern darzustellen.
- einen Compiler bestehend aus Scanner und Parser aufbauend auf der Theorie der Formalen Sprachen für eine gegebenen Aufgabenstellung zu entwerfen und zu erstellen.

Literaturliste

A.V. Aho, R. Sethi, J.D. Ullmann: Compilerbau. Band 1 und 2, Addison-Wesley 1999

A.V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman Compilers: Principles, Techniques, and Tools. Addison-Wesley, 2007.

A.W. Appel modern compiler implementation in java, Cambridge University Press 2004
Download:
<http://eden.dei.uc.pt/~amilcar/pdf/CompilerInJava.pdf>

B. Bauer, H. Höllerer: Übersetzung objektorientierter Programmiersprachen: Konzepte, Abstrakte Maschinen Und Praktikum: Java-Compiler; Springer; 4. Auflage; 2013

S.D. Bergmann Compiler Design: Theory, Tools, and Examples; free download: <http://elvis.rowan.edu/~bergmann/books/cd/java/CompilerDesignBook.pdf> (Computer Science Department, Rowan University), 2016

H. Herold: Linux-Unix-Profitools. Addison-Wesley 1999

D. Grune, K. van Ree, H.E. Bal, C.J.H. Jacobs, K. Langendoen: Springer; 2. Auflage 2012

R.H. Güting, M. Erwig: Übersetzerbau; Springer 1999

A. Kunert: LR(k)-Analyse für Pragmatiker; Humboldt-Universität zu Berlin; Institut für Informatik / ZE Rechenzentrum (CMS) (Dissertation) 2011

Levine, J. R., Mason, T., Brown, D.: lex & yacc; O'Reilly & Associates 1995

A.J.D. Reiss. Compiler Construction using Java, JavaCC, and Yacc; Wiley, 2012.

F.J. Schmitt: Praxis des Compilerbaus; C. Hanser 1992

Wagenknecht C, Hielscher M.: Formale Sprachen, abstrakte Automaten und Compiler, Lehr- und Arbeitsbuch für Grundstudium und Fortbildung, Vieweg Teubner 2009
über Springer Link als download verfügbar!

2.10 Computer Games Development

Information about the module

engl. Name	Computer Games Development
Code	CGDEV4.WP
Coordinator	Philip McClenaghan
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course during the semester break. (February/March) and (August/September)
Courses	Computer Games Development (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical classes and workshops
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 10-30 minutes, 40%• Written assignment, 8-25 pages, 60%
Examination number	IN 3970322, 2970788 TI 3976563, 2976562 WI 3975708 IIS 9775108
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The aim of this course is to provide students with an understanding of computer game theory and design. This is not a technical course. Conceptual design and critical analysis exercises allow students to explore a range of relevant topics in order to gain the ability to look at computer games objectively and from an informed standpoint. Students present their work (in English) both verbally and in written form through presentations and analysis documents.

Qualification aims for the module learning objectives/skills

On completion of this module, the student will be able to demonstrate:

- An appreciation of the computer games industry
- An understanding of computer games design and the ability to critically evaluate computer games
- An understanding of design implementation
- The ability to create a pre-production games proposal document
- The ability to articulate course related ideas and concepts in English, both verbally and in written form

Reading list

Sylvester, T. (2013) Designing Games: A Guide to Engineering Experiences. O'Reilly

Gamasutra Website (<http://www.gamasutra.com/>)

2.11 Corporate Entrepreneurship

Informationen über das Modul

Name / engl.	Corporate Entrepreneurship / Corporate Entrepreneurship
Kürzel	CES4.WP
Verantwortlicher	Prof. Dr. Christoph Buck
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Corporate Entrepreneurship (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht zur theoretischen Vermittlung von Kenntnissen und Kompetenzen in Verbindung mit einer interaktiven Anwendung und Reflexion des Erlernten im Sinne eines erfahrungsbasierten Lernens.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Zwischenpräsentation, 15 Minuten, 30%• Abschlusspräsentation, 25 Minuten, 35%• Schriftliche Ausarbeitung der Abschlusspräsentation, ca. 6-8 Seiten, 35%
Ergänzende Hinweise zur Prüfungsform	Die Präsentationen sind Gruppen-Präsentationen. Studienarbeit: in Gruppenarbeit sollen konkrete Lösungsvorschläge für praxisnahe Problemstellungen erarbeitet und präsentiert werden.

Prüfungsnummer	IN 3970396, 2970894 TI 3976564, 2976716 WI 3975814 IIS 9775174
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Entwickeln, Bewerten und Umsetzen von Innovationen in (Groß-)Unternehmen besteht aus verschiedenen erlernbaren Fähigkeiten.

In diesem Kurs lernen Studierende:

- die Grundlagen von Corporate Entrepreneurship;
- die Besonderheiten, Bedarfe und Herangehensweisen von Corporate Entrepreneurship;
- Strategien, Werkzeuge und Methoden für Entrepreneurship innerhalb von Unternehmen und wenden diese im Rahmen von praxisnahen Problemstellungen an
- Chancen, Risiken und Herausforderungen von Corporate Entrepreneurship.

Dazu werden die Studierenden entlang der Veranstaltungen von einem Industriepartner begleitet.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Innerhalb von Organisationen Möglichkeiten für Corporate Entrepreneurship identifizieren und bewerten zu können
- Innovative Lösungen zu entwickeln und eine Strategie für deren Umsetzung in einem Unternehmen zu erstellen
- Geschäftsmodelle zu erstellen, die in Bezug auf Kosten, Nutzen, Risiken und Chancen im Corporate-Kontext eingebettet werden können.

Literaturliste

Osterwalder, A., Pigneur, Y. (2010): Business Model Generation Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt am Main, 2010.

Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A. (2014): Value Proposition Design.

Nambisan, S., Lyytinen, K., Majchrzak, A., Song, M. (2017): Digital Innovation Management: Reinventing Innovation Management Research in a Digital World. Management Information Systems Quarterly, 41 (1), 223–238.

Kohli, R., Melville, N.P. (2018): Digital innovation A review and synthesis. Information Systems Journal, 29 (1), 200–223.

Christensen, C. M. (2011): The innovator's dilemma: Warum etablierte Unternehmen den Wettbewerb um bahnbrechende Innovationen verlieren. Vahlen.

Kraus, R., Kreitenweis, T., & Jeraj, B. (2022): Intrapreneurship. Springer.

2.12 Datenkommunikation

Informationen über das Modul

Name / engl.	Datenkommunikation / Fundamentals of Data Communications
Kürzel	DAKO4.WP
Verantwortlicher	Prof. Dr. Rolf Winter
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Datenkommunikation (3 SWS) zugehöriges Praktikum (1 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	WPF nur für Bachelorstudiengänge: Wirtschaftsinformatik, International Information Systems und Interaktive Medien. Für Informatik (Bachelor) und Technische Informatik (Bachelor) handelt es sich um ein Pflichtfach.
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: Taschenrechner
Prüfungsnummer	WI 3975755 IIS 9775111
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Funktionsweise und Aufbau der Internet-Architektur und seiner Prinzipien und Protokolle insbesondere:

- Protokolle der Anwendungsschicht (wie HTTP und DNS)
- Transport-Protokolle (wie TCP und UDP)
- Routing-Protokolle (link state und distance vector)
- Protokolle der Sicherungsschicht (z.B. Ethernet)
- Arbeitsweise von Kernkomponenten des Internets (Switches, CDNs, NAT, uvm.)
- Aspekte der Netzsicherheit (z.B. Paketfilter)
- Schlüsselprinzipien des Internets (Zuverlässige Datenübertragung, Staukontrolle etc.)
- Umgang mit Standardwerkzeugen (Software) im Bereich Netzwerke
- Netzeinrichtung, Wartung und Fehlerdiagnose

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden kennen die Schlüsselprotokolle des Internets und können deren Aufgaben und Funktionsweise im Detail erklären. Sie wissen welche Funktionen der Internet-Architektur wie und wo im Netz implementiert sind. Auch die komplexen Zusammenhänge zwischen Protokollen und Mechanismen im Internet können Studierende beschreiben.

Darüber hinaus können die Studierenden ihr erlerntes Wissen auch praktisch bei der Entwicklung von vernetzten Anwendungen oder der Einrichtung und Wartung von Netzen einsetzen. Das Praktikum befähigt Studierende mit Standardwerkzeugen Anwendungen und Netze zu analysieren und einzurichten.

Literaturliste

Kurose, J.F./ Ross, K.W.: Computernetzwerke, 6. Auflage, Pearson Studium, 3/2014, ca. 900 Seiten, ISBN 978-3-8689-4237-8

2.13 Datenkommunikation im Fahrzeug

Informationen über das Modul

Name / engl.	Datenkommunikation im Fahrzeug / Data communication in the vehicle
Kürzel	DAKOFZ4.WP
Verantwortlicher	Prof. Dr.-Ing. Thomas Kirchmeier
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Vorlesung (2 SWS) Interaktives Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse.
Voraussetzungen	Kenntnisse über eine der folgenden Programmiersprachen: Python, C++, Java, oder andere, solange Protobuf unterstützt wird.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970369, 2970867 TI 3976576, 2976678 WI 3975787 IIS 9775112
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Veranstaltung veranschaulicht anhand von praktischen Beispielen den grundlegenden Aufbau und Funktionsweise eines Fahrzeuges aus Sicht der Datenübertragung. Dabei werden nicht nur die Übertragungstechnologien und -protokolle betrachtet, sondern vor allem die Rahmenbedingungen und strategischen Überlegungen zum Interface-Design. Das verteilte System Fahrzeug steht damit im Vordergrund. Zur besseren Veranschaulichungen werden einzelne Funktionen programmiert, die anschließend mittels Google Protocol Buffer miteinander interagieren. Zusammengefasst adressiert die Veranstaltung die folgenden Themenbereiche:

- Historische Entwicklung und Rahmenbedingungen in der Fahrzeugentwicklung
- Fahrzeugarchitektur
- Grundlegende Kommunikationssysteme im Fahrzeug, vom Feldbus zur IP-Kommunikation
- Middleware (SOMEIP und ServiceDiscover, protobuf, etc.)
- Bedatung und Schnittstellenmodellierung
- Trace- und Fehleranalyse
- Funktionale Sicherheit und der Umgang mit unsicheren Kommunikationskanälen
- Zeitsynchronisation und Security in der Fahrzeugkommunikation

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage,

- die Rahmenbedingungen der Softwareentwicklung im Automobilbereich herauszustellen.
- die Hintergründe und den Aufbau der Fahrzeugsystemarchitektur zu beurteilen.
- unterschiedliche Kommunikationssysteme im Fahrzeug zu planen.
- das SOMEIP-Protokoll und ServiceDiscovery zu beurteilen.
- können Schnittstellen mittels Protobuf erstellen und generieren.
- Einflüsse von Safety und Security auf die Datenkommunikation zu adaptieren.
- den Mechanismus der Zeitsynchronisation über ein asynchrones Netzwerk zu modifizieren.

Literaturliste

Matheus, K.; Königseder, T. Automotive Ethernet, Cambridge University Press; Auflage: 2 (13. Juli 2017), ISBN: 978-1107183223.

Völker, L. COMMUNICATION PROTOCOLS FOR ETHERNET IN THE VEHICLE. IQCP Congress, 09 –11 DECEMBER 2013, STUTTGART MARRIOTT HOTEL SINDELFINGEN, <https://www.iqpc.com/media/9048/29408.pdf>

Kirchmeier, T. Design and Qualification of Automotive Ethernet - An OEM Perspective. Automotive Ethernet Congress. Munich, Germany: 4-5 February 2015.

Kirchmeier, T. Automotive Ethernet: How to handle the difference between the standard and its implementation. IEEE Ethernet & IP @ Automotive Technology Day. Paris, France: 20-21 September 2016.

Völker, L. SOME/IP SERVICE DISCOVERY, Vector Symposium 2014-05-27, http://some-ip.com/papers/2014-05-27-DrLarsVoelker-The_need_for_Service_Discovery_in_the_vehicle.pdf

Overview of additional publications to SOMEIP and Service Discovery: <http://some-ip.com/papers.shtml>

2.14 Digitale Innovationen

Informationen über das Modul

Name / engl.	Digitale Innovationen / Digital Innovation
Kürzel	DIGINN4.WP
Verantwortlicher	Prof. Dr. Christoph Buck
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester Das Modul findet im WS2024/25 nicht statt.
Lehrveranstaltungen	Digitale Innovationen (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht zur theoretischen Vermittlung von Kenntnissen und Kompetenzen und begleitende Übungen mit Praxisbeispielen zur interaktiven Anwendung und Reflexion des Erlernten im Sinne eines erfahrungsbasierten Lernens.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Papervorstellung, 15 Minuten, 10%• Präsentation A, 20 Minuten, 20%• Präsentation B, 20 Minuten, 20%• Abschlusspräsentation, 30 Minuten, 50%
Prüfungsnummer	IN 3970382, 2970880 TI 3976580, 2976696 WI 3975800 IIS 9775116

Inhalte des Moduls

- Einführung zum Thema “Digitale Innovationen”
- Überblick zum Thema digitale Innovation (in Abgrenzung zu traditioneller Innovation) und digitale Technologien als Wegbereiter neuer Geschäftsmodelle
- Methoden und Strukturierungsansätze zur Analyse, zum Design und zur Neuentwicklung (digitaler) Innovationen und Wertschöpfungsmechanismen
- Überblick über spannende traditionelle und neue, digitale Innovationen und Geschäftsmodelle
- Transfer auf praktische Anwendungsbeispiele regionaler Unternehmen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul „Digitale Innovationen“ sind die Studierenden in der Lage

- Definitionen und Strukturierungsansätzen zu digitaler Innovation wiederzugeben
- Charakteristika digitaler Innovationen zu beschreiben sowie Unterschiede zu anderen Innovationstypen zu nennen
- Einflüsse der Digitalisierung auf traditionelle Geschäftsmodelle und Innovationen abzuleiten
- Frameworks, Theorien und Innovationsmethoden und -tools (z.B. Value Proposition Design) anzuwenden, um selbst digitale Innovationen zu entwickeln und zu analysieren
- Realweltbeispiele digitaler Innovationen aus unterschiedlichen Perspektiven zu beurteilen und zu vergleichen

Außerdem erlernen Studierende Softskills, wie z.B. Teamarbeit, Strukturierungs- und Präsentationsfähigkeiten.

Literaturliste

Osterwalder, A., Pigneur, Y. (2010) Business Model Generation Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt am Main, 2010.

Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A. (2014) Value Proposition Design.

Nambisan, S., Lyytinen, K., Majchrzak, A., Song, M. (2017) Digital Innovation Management: Reinventing Innovation Management Research in a Digital World. Management Information Systems Quarterly, 41 (1), 223–238.

Kohli, R., Melville, N.P. (2018): Digital innovation A review and synthesis. Information Systems Journal, 29 (1), 200–223.

Weitere Literatur gemäß gesonderter Angabe.

2.15 Digital Business Leadership Skills

Informationen über das Modul

Name / engl.	Digital Business Leadership Skills / Digital Business Leadership Skills
Kürzel	DIBUS6.WP
Verantwortlicher	Prof. Dr. Norbert Gerth
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Digital Business Leadership Skills (6 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Gastvorträge, Best Practices, Eigenarbeit, Präsentationen
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Präsentation, 30 Minuten, 70%• Studienarbeit, 6-18 Seiten, 30%
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Hintergrund:

- Die Digitalisierung als neuer Megatrend stellt in ihrer Radikalität und Geschwindigkeit alle Branchen vor große Herausforderungen (Stichwort ‚Disruption‘).
- Dabei geht es nicht nur um den Einbezug neuer Schlüsseltechnologien.
- Vielmehr verändern sich gerade grundlegende Herangehensweisen und Ansätze, angefangen im Bereich Forschung und Entwicklung (agiles, kundenzentriertes Innovationsmanagement) reichen diese über das Personalmanagement (Teamführung und Motivation) bis hin zur Art und Weise, wie Unternehmen zukünftig mit ihren Kunden interagieren.
- All dies stellt Unternehmen vor große Herausforderungen.

Welche neuen Ansätze hier zu beachten sind, ist Schwerpunkt dieser Veranstaltung. Die Studenten sind aufgefordert, sich die praxisnahen Inhalte im Rahmen von Studienarbeiten selbst zu erarbeiten. Anschließend werden die Ergebnisse im Kreis aller Teilnehmer vorgestellt und diskutiert.

- Unternehmen im Digitalen Wandel
- Chancen der Disruption für Startup-Gründer
- Digitale Schlüsseltechnologien und ihre Business-Potenziale
- Neue Organisationskonzepte etablierter Unternehmen (Digital Units) und Change Management
- Was etablierte Unternehmen von Startups lernen können?
- Agile Unternehmensführung, Leadership Communication & Team Productivity
- Chancen und Risiken einer Startup-Industry-Cooperation
- Methoden kundenzentrierter Produktentwicklung (u.a. Design Thinking; Lean Startup)
- Innovation-Selling, Acceleration und Growth Hacking
- Digital Marketing und E-Commerce

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Studierende des Kurses sollten durch ihre Teilnahme ...

- die Relevanz der Digitalisierung für Unternehmen verstehen
- die Chancen der Disruption für Startup-Gründer erkennen
- die Business-Potenziale ausgewählter Digitaler Schlüsseltechnologien besser einschätzen lernen
- Einblicke erhalten in neuere Management- und Organisationskonzepte des DIG Zeitalters
- wichtige Methoden einer kundenzentrierten Produktentwicklung kennen lernen
- die Herausforderungen der Vermarktung von Digitalen Innovationen erkennen
- Hinweise erhalten zu möglichen Lösungsansätzen im Rahmen des Digital Marketing und E-Commerce

In diesem Sinne wird in diesem Seminar besonderes Augenmerk auf die Diskussion aktueller und praxisrelevanter Fragestellungen gelegt.

Literaturliste

Die jeweils themenrelevante Literatur ist von den Teilnehmern eigenständig zu recherchieren.

2.16 Digital Transformation in Organizations

Information about the module

engl. Name	Digital Transformation in Organizations
Code	DTO4.WP
Coordinator	Prof. Dr. Jens Lauterbach
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Digital Transformation in Organizations (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical group work and case studies, industry talks
Prerequisites	Students should have acquired basic skills in informatics or business information systems to understand core concepts/fundamentals behind business organizations and digital technologies. Bachelor (5th semester) or master in business information systems or computer science is recommended.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 50%• Written assignment, 10-15 pages, 50%
Examination number	IN 3970377, 2970875 TI 3976579, 2976686 WI 3975795 IIS 9775115
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Digitalization is one of the megatrends of our time. We live in a time where digital technologies and their applications make astonishing progress. Cars become driverless, computers beat humans in chess and Jeopardy and 3D-printers create houses. In the first part of this course the terms digitalization and digital transformation will be defined and the foundations are laid. Specifically, the following topics will be covered:

- Digital transformation – why it is one of the biggest buzzwords but also megatrends of our time
- Digitalization and digital transformation: Definition and delimitation
- A framework for organizations, individuals, and digital technology
- Historical evolution of industry and (digital) technologies
- Key digital technologies of our time
- Influence of digital technologies on organizations

Many organizations are confronting the question of how to design and manage the digital transformation. Based on phase-models of innovation adoption, the generic transformation process will be explained. Along this process, specific tasks and challenges that an organization needs to design and manage will be introduced. Specifically, the following topics will be covered:

- Stage models for digital transformation in organizations
- Key design aspects for digital transformations
- Methods and instruments to design, manage and facilitate digital transformations

Overall, this course is aimed at giving students the opportunity to learn and practice important aspects of digital transformations in organizations, one of the most pressing topics of our time for businesses around the globe. Group work with (research) papers and case studies will be used to complement the concepts and examples from the lecture. In industry talks, practitioners will share their own experiences from digital transformation management.

Qualification aims for the module learning objectives/skills

Students that aim at learning the design and management aspects of digitalization in organizations will create and deepen their knowledge. Students will be prepared for working in digital transformation projects in business organizations. After successful participation, students particularly will:

- Understand the term and the reasons for accelerated digital transformation in organizations
- Understand the technological and conceptual foundations of digital transformation
- Remember the historical evolution of industries and (digital) technologies
- Understand the influence of digital technologies on organizations
- Understand the typical phases and tasks in digital transformations
- Analyze and evaluate design and management problems in digital transformations
- Apply methods and instruments to create solutions for real world problems in the context of digital transformation projects

Reading list

Literature recommendations will be provided in the lecture

2.17 Einführung in die maschinelle Sprachverarbeitung

Informationen über das Modul

Name / engl.	Einführung in die maschinelle Sprachverarbeitung / Introduction to Natural Language Processing
Kürzel	EMSV4.WP
Verantwortlicher	Dr. Phil. Alessandra Zarcone
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Einführung in die maschinelle Sprachverarbeitung (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützen und fördern die Übungen das Selbststudium.
Voraussetzungen	Grundlagen der Informatik wie sie im Grundstudium vermittelt werden.
Verwendbarkeit	Wahlpflichtfach für Masterstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: Taschenrechner
Ergänzende Hinweise zur Prüfungsform	für Interaktive Medien <ul style="list-style-type: none">• Klausur, 60 Minuten, Hilfsmittel: Taschenrechner, 70%• semesterbegleitende Arbeit, 30%, alternativ:<ul style="list-style-type: none">- Präsentation (15-30 Minuten)- Studienarbeit (6-10 Seiten)
Prüfungsnummer	IN 3970378, 2970876 TI 3976583, 2976687 WI 3975796 IIS 9775119

Inhalte des Moduls

- Morphologische und syntaktische Einheiten, Modellierung von Strukturen und Regeln
- N-Gram-Sprachmodelle
- Text-Klassifikatoren: naive-Bayes und logistische Regression
- Wörter als Vektoren und Embeddings
- Neuronale Sprachmodelle
- Sequenzkennzeichnung & Named Entity Recognition
- Vortrainierte Sprachmodelle
- Kontextuelle Embeddings
- Chatbots und Dialogsysteme
- Datenannotation für qualitative Analyse und maschinelles Lernen
- Evaluierung von Modellen und Werkzeugen
- Industrielle Anwendungen und gesellschaftliche Implikationen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- die typischen Herausforderungen der Verarbeitung natürlicher Sprache (Mehrdeutigkeit, Kontextabhängigkeit) darzulegen
- die aktuellen Methoden der maschinellen Sprachverarbeitung zu beschreiben
- die passende technische Lösung für typische Anwendungsfälle zu identifizieren und exemplarische Verarbeitungsmethoden auf einfache Beispiele anzuwenden

Literaturliste

Altinok, D.: Mastering spaCy: An end-to-end practical guide to implementing NLP applications using the Python ecosystem, 2021.

Carstensen, K.: Computerlinguistik und Sprachtechnologie: Eine Einführung, 2009.

Jurafsky, D.; Martin, J.H.: Speech and Language Processing. Entwurf der 3. Auflage online verfügbar <https://web.stanford.edu/jurafsky/slp3/>, 2021.

Verwendete Software:

- Python:
<https://www.python.org>
- Spacy:
<https://spacy.io/>

2.18 Einführung in die Robotik

Informationen über das Modul

Name / engl.	Einführung in die Robotik / Introduction to Robotics
Kürzel	EROB4.WP
Verantwortlicher	Prof. Dr. Michael Strohmeier
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Einführung in die Robotik (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen zur Einführung in die Robotik
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: nicht-prog. Taschenrechner, 2 DIN-A4-Seiten handgeschriebene Formelsammlung
Prüfungsnummer	IN 3970400, 2970898 TI 3976623, 2976724 WI 3975818 IIS 9775178
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Übersicht über verschiedene Anwendungsfelder der Robotik
- Räumliche Darstellung: Koordinatensysteme und Homogene Transformationen
- Einführung in gängige Regelungsarchitekturen der Robotik
- Direkte und inverse Kinematik für mobile Roboter und Manipulatoren
- Dynamikprinzipien am Beispiel von einfachen Robotern und Multicoptern
- Überblick über Sensoren in der Robotik und deren Messprinzipien
- Sensorfusion: Komplementärfilter und Kalman Filter
- Kartierung und Lokalisierung, z.B. Partikelfilter und SLAM
- Grundlegende Techniken zur Pfadplanung und Hindernisvermeidung
- Maschinelles Lernen: Einführung in Reinforcement Learning

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul verstehen die Studierenden die grundlegenden Prinzipien der Robotik. Sie können einfache Robotersysteme in Bezug auf ihre Kinematik, Dynamik und Regelungsstrukturen analysieren und entwerfen. Die Studierenden kennen verschiedene Sensortechnologien und Messprinzipien. Sie verstehen und können die Grundlagen der Sensorfusion anwenden. Sie verstehen grundlegende Algorithmen zur Kartierung, Navigation und Hindernisvermeidung. Zudem verstehen sie die Grundlagen von Machine Learning-Techniken und kennen deren Anwendung in der Robotik.

Literaturliste

Hertzberg J., Lingemann K., Nüchter A. *Mobile Roboter: Eine Einführung aus Sicht der Informatik*, Springer-Verlag, 1. Ausgabe, 2012

Siciliano B., Sciavicco L., Villani L., Oriolo G. *Robotics: Modelling, Planning and Control*, Springer, 1st Edition, 2008

Siegwart R., Nourbakhsh I.R., Scaramuzza D. *Introduction to Autonomous Mobile Robots*, MIT press, 2nd Edition, 2011

Sola, J. *Quaternion kinematics for the error-state Kalman filter*, arXiv preprint, 2017

Corke P.I., Witold J., Remo P. *Robotics, vision and control*, Springer, 2011.

2.19 Elektronische Handelssysteme

Informationen über das Modul

Name / engl.	Elektronische Handelssysteme / Electronic Trading Systems
Kürzel	ELHS4.WP
Verantwortlicher	Prof. Dr. Arne Mayer
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird unregelmäßig bzw. auf Nachfrage angeboten.
Lehrveranstaltungen	Elektronische Handelssysteme (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht zu Beginn – Unterstützt durch Case Studies, Gruppendiskussionen und Gastvorträge. Im weiteren Verlauf Arbeit in Kleingruppen, in denen sich die Studierenden die praxisrelevanten Inhalte selbst erarbeiten. Anschließend werden diese Ergebnisse im Kreis aller Teilnehmenden vorgestellt und diskutiert.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Präsentation, 15 Minuten, 60%• Studienarbeit, 10-15 Seiten, 40%
Prüfungsnummer	IN 3970376, 2970874 TI 3976565, 2976685 WI 3975794 IIS 9775120
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Elektronischer Handel (e-Commerce) als Teil des E-Business wird immer bedeutender und drängt klassische, direkte Handelsbeziehungen in den Hintergrund. In diesem Modul werden die zugrundeliegenden IT-Systeme – aus fachlicher, geschäftlicher Sicht – beleuchtet:

- Teilgebiete des E-Business
- Technische/Technologische Rahmenbedingen der Internet-Ökonomie als Treiber für E-Business
- Aufbau und Bestandteile von Elektronischen Handelssystemen
- Spezifika des elektronischen Handels (E-Commerce) wie Plattformökonomie, Erlösmodelle
- Technologische Trends
- Analyse in der Praxis existierender elektronischer Handelssysteme: Modellierung/Dokumentation derer Geschäftsprozessen mittels BPML
- Implementierung von elektronischen Handelssystemen: In Kleingruppen designen und implementieren die Studierenden einen e-shop - mit Hilfe bestehender Software oder selbst (bei Wunsch und entsprechenden Vorkenntnissen!)

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Mit einer erfolgreichen Teilnahme am Modul können Studierende:

- die Bedeutung des E-Business und dessen Teilgebiete für die Wirtschaft erkennen und einordnen
- die Eigenschaften und notwendigen Prozesse des e-Commerce und insbesondere elektronischer Handelssysteme analysieren und verstehen
- Umsetzungskompetenz für Beruf oder Gründung erlangen
- die erarbeiteten Ergebnisse zielgruppengerecht präsentieren

Literaturliste

Wird in der ersten Präsenzveranstaltung bekannt gegeben.

2.20 Embedded Linux

Informationen über das Modul

Name / engl.	Embedded Linux / Embedded Linux
Kürzel	EMLI6.WP
Verantwortlicher	Prof. Dr. Hubert Högl
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Embedded Linux (6 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	<ul style="list-style-type: none">• Seminaristischer Unterricht• Praktische Übungen und Projekte
Voraussetzungen	Kenntnisse von Linux auf dem Desktop-Rechner, vor allem das Arbeiten auf der Kommandozeile (z.B. durch Wahlpflichtfach "LPIC") und Mikrocomputertechnik (z.B. Embedded Systems I und II) sind hilfreich, aber nicht zwingend notwendig.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Fernklausur mit Videoaufsicht, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Motive für Linux auf eingebetteten Systemen
- Typische Hardware von Embedded Linux Rechnern
- Installation des Entwicklungsrechners
- Bootloader
- Linux Kernel
- Gerätetreiber
- Schnittstellen (UART, GPIO, SPI, I2C, ADC, PWM) und ihre Programmierung
- Anwendungsprogrammierung
- Filesysteme
- Debugging
- Echtzeit

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden erlangen:

- Kenntnis des GNU/Linux Entwicklungsprozesses
- Verständnis der Funktion eines Gerätes auf der Basis von Embedded Linux
- Fähigkeit, eine eigene Produktidee in der Praxis mit Embedded Linux umzusetzen

Literaturliste

Chris Simmonds, Mastering Embedded Linux Programming, Packt Publishing 2015.

Rodolfo Giometti, GNU/Linux Rapid Embedded Programming, Packt Publishing 2017.

Weitere Informationen auf der Homepage von Prof. Högl
<http://hhoegl.informatik.hs-augsburg.de>

2.21 Flugrobotik

Informationen über das Modul

Name / engl.	Flugrobotik / Flying Robots
Kürzel	FLUGRO4.WP
Verantwortlicher	Prof. Dr. Constantin Wanninger
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Flugrobotik (2 SWS) Praktikum Flugrobotik (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen zur Einführung in die Flugrobotik
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Mündliche Prüfung, 30 Minuten, 50%• Projektarbeit, Vorführung / Kolloquium, 50%
Prüfungsnummer	
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Vorlesung „Flugrobotik“ vermittelt grundlegende Kenntnisse über unbemannte Flugroboter durch eine Kombination aus Theorie und praktischen Übungen. Folgende Inhalte werden unter anderem behandelt:

- Grundlagen der Flugrobotik
- Sensorik und Aktuatorik
- Kartesische Koordinaten und Transformationen
- Das Robot Operating System (ROS)
- Pfadplanung und Kollisionsvermeidung
- Softwaretechnikexkurs

Abschließend wird ein kleines Projekt mit Drohnen realisiert.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Vorlesung soll die Studierenden befähigen, die Grundlagen der Flugrobotik zu verstehen und praktisch anzuwenden.

- Verständnis der Grundlagen von Flugrobotern
- Praktische Erfahrung mit Drohnensteuerung und -programmierung

Literaturliste

Online Dokumentation des Robot Operating System (ROS), <https://www.ros.org/>

Macenski, Steven: Robot operating system 2: Design, architecture, and uses in the wild, Science robotics (2022).

Gugan, Gopi: Path planning for autonomous drones: Challenges and future directions, Drones (MDPI) (2023).

Yang, Hyunsoo: Multi-rotor drone tutorial: systems, mechanics, control and state estimation, Intelligent Service Robotics (Springer) (2016).

2.22 Formula Student Driverless

Informationen über das Modul

Name / engl.	Formula Student Driverless / Formula Student Driverless
Kürzel	FSD4.WP
Verantwortlicher	Prof. Dr. Gundolf Kiefer
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul hat eine Laufzeit von zwei Semestern und wird bei entsprechender Nachfrage im Wintersemester sowie dem darauffolgenden Sommersemester angeboten.
Lehrveranstaltungen	Formula Student Driverless (2 SWS + 2 SWS)
Lehrsprache	Deutsch, in Ausnahmefällen (internationale Studierende) und bei den Wettbewerbs-Events auch Englisch
Lehr-/Lernmethoden	Projektarbeit, Seminar, seminaristischer Unterricht, regelmäßige Statusbesprechungen
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Studienarbeit, 10-15 Seiten, 80%• 6 Kurzvorträge, 10-20 Minuten, 20%
Prüfungsnummer	IN 3970373, 2970871 TI 3976587, 2976682 WI 3975791 IIS 9775123
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Studierenden übernehmen die Verantwortung für einen technischen oder organisatorischen Teilbereich der Entwicklung eines Formula-Student-Driverless-Fahrzeugs und entwickeln die dazu gehörigen Komponenten zusammen mit einem studentischen Team.

Die Entwicklung eines Fahrzeuges erstreckt sich üblicherweise über ein Jahr (Wintersemester mit anschließendem Sommersemester) und gliedert sich in die folgenden Phasen, die jeweils mit einer Kurzpräsentation abgeschlossen werden:

- Erarbeitung der Anforderungen für das Teilsystem und Abstimmung im Team (Anforderungsfreeze: Präsentation 1)
- Erstellung eines Designs und Abstimmung der Schnittstellen mit den angrenzenden Komponenten (Designfreeze: Präsentation 2)
- Implementierung / Produktion des Teilsystems (Vorstellung Prototyp: Präsentation 3)
- Komponenten- / Teilsystemtests (Vorstellung der Testergebnisse gegen die Anforderungen: Präsentation 4)
- Integration der Komponente / des Teilsystems ins Gesamtsystem und Durchführung der Integrationstests (Vorstellung der Integrationstestergebnisse mit Fokus auf die Komponente / das Teilsystem: Präsentation 5)
- Betreuung des Teilsystems beim Rennen im Fahrzeug (Erfolgspräsentation / Ausblick: Präsentation 6)

Neben den eigentlichen Präsentationen finden die regelmäßigen Teamtreffen zur Abstimmung der Vorgehensweise und zur Feststellung des Entwicklungsstatus statt.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Kenntnisse:

- Studierende kennen den Aufbau und die Architektur des Gesamtsystems in einem autonomen Elektrorennfahrzeug.
- Sie kennen den Entwicklungsprozess und wissen diesen termingerecht zu durchlaufen.
- Sie wissen sich in ein interdisziplinäres Team zu integrieren und die technischen und organisatorischen Schnittstellen abzustimmen.
- Sie wissen um die Bedeutung der koordinierten Eskalation von technischen, terminlichen und kommunikativen Problemen im eigenen Entwicklungsbereich, sowie an den Schnittstellen zu Teammitgliedern, Lieferanten und Sponsoren.

Fertigkeiten:

- Die Studierenden können ein Teilsystem durch den kompletten Entwicklungsprozess führen und wissen, wie man es termingerecht zu einem Reifegrad führt, der einen robusten und sicheren Betrieb im Fahrzeug beim Rennen gewährleistet.
- Durch den Kontakt mit Sponsoren und Partnern aus der Industrie und dadurch gewonnene Erfahrung können die Studenten sich selbst und ihre Entwicklungsergebnisse in englischer und deutscher Sprache präsentieren.

Kompetenzen:

- Die Studierenden sind in der Lage Risikobeurteilungen durchzuführen, Rückfalllösungen vorzubereiten und termingerecht zu entscheiden, wann diese zum Einsatz kommen müssen.
- Im Rahmen der Teamführung für ein Teilsystem beurteilen die Studierenden den kontinuierlichen Fortschritt und Reifegrad und können technische Entscheidungen fundiert herbeiführen.

Literaturliste

- Reglement der Formula Student Driverless und Formula Student Electric
- Dokumentation der bereits entwickelten FSD- und FSE-Fahrzeuge der HSA

2.23 Fullstack-Webentwicklung

Informationen über das Modul

Name / engl.	Fullstack-Webentwicklung / Fullstack Web Development
Kürzel	FSWD6.WP
Verantwortlicher	Prof. Dr. Wolfgang Kowarschick
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig im Sommersemester angeboten. Das Modul findet nur statt, wenn sich genügend Teilnehmer anmelden.
Lehrveranstaltungen	Fullstack-Webentwicklung (2 SWS) Fullstack-Webentwicklung Praktikum (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Praktikum
Voraussetzungen	Die Kenntnis der Inhalte des Moduls Datenmanagement ist sehr sinnvoll, aber nicht zwingend notwendig.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Studienarbeit(Dauer 110 - 150 h), 90%• Präsentation, 10%
Prüfungsnummer	IN 3970368, 2970866 TI 3976589, 2976677 WI 3975786 IIS 9775125
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Programmierung
 - Grundlagen der Sprachen ECMAScript (JavaScript) und TypeScript
 - Clientprogrammierung (ECAMScript-/TypeScript-basiert), Serverprogrammierung (ECAMScript-/TypeScript-basiert), Datenspeicherung (JSON-Format, RDBMS)
 - Kommunikation zwischen Client und Server (zum Beispiel REST)
 - Entwicklung von einfachen Web-Systemen mit Hilfe geeigneter Frameworks.
- Programmierprinzipien
 - Modularisierung
 - Asynchronität (ohne Threads)
 - Wiederverwendbarkeit (insb. Don't repeat yourself, DRY)
 - Model-View-Controller-Pattern, Observer-Pattern ...
- kollaboratives Arbeiten mittels Git

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Kenntnisse:

Die Studierenden können

- ein Webprojekt nach gegebenen Voraussetzungen und Anforderungen planen und umsetzen.

Fertigkeiten:

Die Studierenden können

- eine REST-API planen und programmieren,
- eine relationale Datenbank für ein Webprojekt planen und implementieren,
- das Frontend einer Web-Applikation den Anforderungen entsprechend designen und mit einem aktuellen Web-Framework umsetzen,
- die Entwicklung eines Webprojekts mit Hilfe von Verwaltungssoftware strukturieren und versionieren,
- unter Zuhilfenahme von Cloud-Plattformen ihre Applikation online bereitstellen und managen.

Kompetenzen:

Die Studierenden sind in der Lage,

- sich selbstständig in neue Web-Technologien einzuarbeiten, um mit der rasanten Entwicklung in diesem Gebiet Schritt zu halten.

Literaturliste

Vorlesungsskript

Vue.js-Dokumentation:

<https://vuejs.org/v2/guide>

Phoenix-Dokumentation:

<https://hexdocs.pm/phoenix/overview.html>

PostgreSQL-Dokumentation:

<https://www.postgresql.org/docs/online-resources>

Deployment:

<https://devcenter.heroku.com>

Deployment:

<https://docs.netlify.com>

2.24 Grundlagen DevOps

Informationen über das Modul

Name / engl.	Grundlagen DevOps / Fundamentals of DevOps
Kürzel	DEVOPS4.WP
Verantwortlicher	Prof. Matthias Kolonko, Ph.D. (ONPU)
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Grundlagen DevOps (4 SWS)
Modulbereich	Anwendungen
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Praktische Übungen
Voraussetzungen	Programmieren 1+2
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970387, 2970885 TI 3976590, 2976707, WI 3975805, IIS 9775167
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Vorlesung befasst sich mit den wichtigsten Elementen der Infrastruktur für einen strukturierten Softwareentwicklungsprozess. Hierbei werden verschiedene Repräsentanten der unterschiedlichen Kategorien dieser Unterstützungstools besprochen und die Unterschiede herausgestellt. Der richtige Einsatz sowie die richtige Anwendung dieser Tools und deren Zusammenspiel sollen dabei besonders beleuchtet werden.

Im einzelnen werden hierbei folgende Kategorien und Tools mit den entsprechenden Repräsentanten betrachtet:

Versionierung Git, SVN, CVS, ...

Bug Tracker JIRA, Mantis, Redmine, ...

Build Tools Ant, Maven, ...

Continuous Integration Jenkins, ...

Hierbei soll auch kurz auf den Bereich ITIL eingegangen werden, wobei besonders die Abgrenzung zwischen o.g. Bug Trackern und Ticketsystemen herausgestellt wird.

Im Rahmen der Veranstaltung sollen die unterschiedlichen Tools und deren Verzahnung auch praktisch angewendet werden. Die Systeme sollen installiert, konfiguriert und mit einfachen Codebeispielen getestet werden.

Dabei sollen die Teilnehmer auch selbständig die Vor- und Nachteile der Anwendung dieser Tools erkennen und gegenüberstellen.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Absolvierung des Moduls sind die Teilnehmer in der Lage

- die aktuellen Tools der genannten Kategorien zu nennen.
- die Vor- und Nachteile der unterschiedlichen Tools zu beschreiben.
- die in der Vorlesung besprochenen Tools richtig anzuwenden.
- einen integrierten Ansatz für die Entwicklung eines Softwareprojektes mit Hilfe der verschiedenen Tools zu entwickeln.

Literaturliste

Literatur wird in der Veranstaltung bekannt gegeben.

2.25 Hard- und Software für das Internet der Dinge

Informationen über das Modul

Name / engl.	Hard- und Software für das Internet der Dinge / Hard- and software for the internet of things
Kürzel	HARSO.WP
Verantwortlicher	Prof. Dr. Volodymyr Brovkov
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Hard- und Software für das Internet der Dinge (2 SWS) zugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: 1 DIN-A4-Seite handgeschrieben
Prüfungsnummer	IN 3970347, 2970842 TI 2976653 WI 3975759 IIS 9775126
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Mikrocontroller: typische Bestandteile/ Einsatz/ Programmierung in C und Python
- Typische Schnittstellen (GPIO, UART, I2C, SPI), Signalpegel, Kompatibilität.
- Typische Sensoren (Temperatur, Feuchtigkeit, Distanz, Beschleunigung, Bewegung, ...)
- Typische Aktoren (Servo, Relais, DC Motor, ...)
- MQTT Protokoll in Internet der Dinge / Raspberry Pi als MQTT Broker / Mikrocontroller WeMos D2 als MQTT Client.
- Stromversorgung in autonomen Systemen
- Beispielimplementierung eines Sensornetzes

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Arbeitsweise des Mikrocontrollers und dessen Schnittstellen zu kennen.
- Mikrocontroller mit Hilfe der Programmiersprache C programmieren zu können.
- Arbeitsweise von typischen Sensoren und Aktoren zu verstehen.
- Kommunikation von mehreren Geräten mit Hilfe von MQTT Protokoll in einem Netz zu erstellen.
- Ein einfaches Datenerfassungssystem mit einigen Sensoren aufgrund eines einfachen Mikrocontrollers implementieren zu können.

Literaturliste

Banzi, Massimo, 2015. Arduino für Einsteiger: 160/ST 170 B219 A6. ISBN: 978-3-95875-055-5,3-95875-055-9

Kofler, Michael, 2016. Raspberry Pi: 160/ST 160 K78(3).

Engelhardt, Erich F., 2016. Sensoren am Raspberry Pi: 160/ST 160 S294. ISBN: 978-3-645-60490-1

Hüning, Felix, 2016. Sensoren und Sensorschnittstellen: 160/ZQ 3120 H887. ISBN: 978-3-11-043854-3,3-11-043854-2,978-3-11-043855-0,978-3-11-042973-2.

Boyd, Bryan, 2014. Building Real-time mobile solutions with MQTT and IBM Message-Sight: ISBN: 978-0-7384-4005-7.

2.26 Hochschul Innovationsprojekt

Informationen über das Modul

Name / engl.	Hochschul Innovationsprojekt / University Innovation Project
Kürzel	HIP.WP
Verantwortlicher	Prof. Dr. Alexander von Bodisco
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Winter- und Sommersemester
Lehrveranstaltungen	Hochschul Innovationsprojekt
Lehrsprache	Das Modul wird in deutscher und englischer Sprache unterrichtet.
Lehr-/Lernmethoden	Studierende erarbeiten in Kleingruppen IT-Lösungen zu einem praxisorientierten Thema für ein IT- bzw. interdisziplinäres Projekt. Ziel ist es einen Projektablauf möglichst realitätsnah mit allen Facetten abzubilden. Die Projektthemen werden von Prüfungsberechtigten der Fakultät für Informatik vergeben und umfassen einen praktischen Teil (Software/Hardware), eine Dokumentation (Studienarbeit) und eine Präsentation. Der praktische Teil (Software und ggf. Hardware) ist im Rahmen der Studienarbeit zu beschreiben. Die Präsentation findet in der Regel im Rahmen eines Projekttag oder eines Seminars statt. Die Abstimmung mit dem Projektsteller erfolgt in regelmäßigen persönlichen Treffen und über elektronische Kanäle. Die Bearbeitung ist nicht notwendigerweise an die Vorlesungszeit gebunden.
Voraussetzungen	Solide Kenntnisse aus den wichtigsten Themenbereichen der Informatik, wie z.B. Algorithmen und Datenstrukturen, Programmieren, Datenbanken, Datenkommunikation, Software Engineering und Betriebssysteme. Die erworbenen Kenntnisse sollten bereits in einem Teamprojekt praktisch angewendet worden sein.
Verwendbarkeit	Wahlpflichtfach für die Bachelorstudiengänge Informatik, International Information Systems, Technische Informatik, Wirtschaftsinformatik

Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h
-------------------------------------	---

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none"> • Projektarbeit, 10-20 Seiten, 80% • Präsentation, 10-20 Minuten, 20%
Prüfungsnummer	IN 3970401, 2970899 TI 3976624, 2976725 WI 3975819 IIS 9775179
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Studierenden führen in Gruppen eigenständige IT-Kleinprojekte durch oder erweitern/unterstützen laufende IT- bzw. interdisziplinäre Projekte aus informatiknahen Studiengängen. Zu den Aufgaben der Studierenden zählen Projektmanagement, Softwareentwicklung, sowie je nach Projekt die selbstständige Einarbeitung in interdisziplinäre Themen.

In moodle finden Sie die aktuellen Themen, die gerade angeboten werden:

<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Softwareaufgaben im Team hinsichtlich Zeit, Aufwände und Ressourcen zu planen und durchzuführen.
- Softwareentwicklungsmethoden praktisch anzuwenden.
- Neue Softwaretechniken selbstständig zu erlernen und geeignete Methoden auszuwählen.
- Interdisziplinäre Themen im Selbststudium aufzubereiten und Fragestellungen zu erarbeiten.
- Projektergebnisse verständlich zu dokumentieren und ansprechend zu präsentieren.

Literaturliste

Projektspezifische Literatur wird vom Betreuer vor Beginn des Projektes bekanntgegeben.

2.27 Industrielle Bildverarbeitung

Informationen über das Modul

Name / engl.	Industrielle Bildverarbeitung / Industrial Image Processing
Kürzel	INDBV4.WP
Verantwortlicher	Prof. Dr. Peter Rösch
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Industrielle Bildverarbeitung (4 SWS)
Verantwortliche Hochschule	TH Augsburg
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Teilnehmerinnen und Teilnehmer erarbeiten Inhalte im Selbststudium anhand von Lehrbüchern und Veröffentlichungen, unterstützt durch vom Dozenten erstellte Lehrvideos und Anleitungen. Im Präsenzteil implementieren Studierende ausgewählte Verfahren und wenden diese auf Bilder aus der Praxis an.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 90 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970389, 2970887 TI 3976593, 2976709 WI 3975807 IIS 9775169
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

In der automatisierten industriellen Produktion ist Bildverarbeitung unverzichtbar, insbesondere für die Qualitätssicherung. Im Verlauf der Lehrveranstaltung lernen Studierende die Methoden der industriellen Bildverarbeitung kennen und erstellen eigene Anwendungen unter Verwendung frei verfügbarer Werkzeuge und Bibliotheken.

- Grundlagen der Bildverarbeitung
- Bildaufnahme
- Bildvorverarbeitung
- Lageerkennung
- Kennzeichnungsidentifikation
- Anwesenheitskontrolle
- Vermessung
- Oberflächenprüfungen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Gängige Methoden der industriellen Bildverarbeitung verbal zu beschreiben.
- Für die Lösung einer Bildverarbeitungsaufgabe geeignete Werkzeuge aus einer Programm-bibliothek auszuwählen und anzuwenden.
- Verschiedene vorgegebene Komponenten zur industriellen Bildverarbeitung systematisch bezüglich Effektivität und Effizienz zu bewerten.
- Lösungen für Bildverarbeitungsaufgaben mittlerer Komplexität selbständig zu entwickeln.

Literaturliste

C. Demant, B. Streicher-Abel, A. Springhoff: Industrielle Bildverarbeitung, 3. Auflage, Springer (2011)

W. Burger, M.J. Burge: Digitale Bildverarbeitung, 3. Auflage, Springer (2015)

R. C. Gonzalez, R. E. Woods: Digital Image Processing, 4th Ed., Pearson (2018)

J. Howse, J. Minichino: Learning OpenCV 4 Computer Vision with Python 3, 3rd Ed., Packt Publishing (2020)

scikit-image, Online-Dokumentation,
<http://scikit-image.org/docs/stable>

2.28 Industrielle Informationsverarbeitung

Informationen über das Modul

Name / engl.	Industrielle Informationsverarbeitung / Industrial Data Processing
Kürzel	INDIV4.WP
Verantwortlicher	Prof. Dr.-Ing. Thomas Kirchmeier
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Industrielle Informationsverarbeitung (4 SWS)
Verantwortliche Hochschule	TH Augsburg
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	<p>Das Modul wird hybrid (Präsenz und Online) unterrichtet und mit Flipped Classroom als Lehrkonzept; bedeutet:</p> <ul style="list-style-type: none">• 2 SWS: Der Vorlesungsinhalt (Lehrvideos, Präsentationen, Aufgaben etc.) ist selbstständig bis zur nächsten Veranstaltung zu bearbeiten (Bearbeitungszeit: 1 Woche).• 2 SWS: Am Vorlesungstag werden Fragen zum Vorlesungsinhalt besprochen und ggf. werden weiterführende Zusammenhänge erläutert.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 90 Minuten, Hilfsmittel: 1 selbstgeschriebene DIN A4 Seite (keine Kopie)
--------------	---

Prüfungsnummer	IN 3970402, 2970900 TI 3976625, 2976726 WI 3975820 IIS 9775180
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Im Rahmen der Vorlesung industrielle Informationsverarbeitung wird die Ableitung von Kenntnissen und Wissen aus verschiedenen Daten in Bezug auf industrielle Anwendungen betrachtet, denn Daten alleine führen per se nicht zu Handlungsempfehlungen. Erst eine Modellbildung aus Daten mit entsprechendem Kontextwissen ermöglicht eine Prädiktion und somit eine Steuerung von Systemen und zukünftigen Entscheidungen. Genauso wichtig wie die Modellbildung selbst ist auch die Ergebnisbewertung mittels Stochastik zur Erhöhung der Aussagekraft sowie zum Ausschluss zufälliger Einflussfaktoren. Auf bestehende Software-Tools wird dabei vorerst verzichtet, da für deren Anwendung ein entsprechendes Grundlagenwissen vorausgesetzt ist. Die für die Vorlesung erforderlichen Teilaspekte der folgenden Themengebiete werden sukzessive anhand von einfachen Beispielen vermittelt:

- Stochastik (beschreibende und bewertende Statistik sowie Wahrscheinlichkeitsrechnung)
- Numerische Algorithmen zur rechnergestützten Lösung gewöhnlicher Differenzialgleichungen
- Gewöhnliche Differentialgleichungen mit Anfangs- und Randwertprobleme
- Mathematische Optimierungsalgorithmen (Least Square, Recursive Least Square, Gradientenabstiegsverfahren)
- Stabilitätsbetrachtungen

Die Umsetzung der Beispielm Modelle und Aufgaben erfolgt in Python. Bezogen auf die Modelle werden nur die grundlegenden Ansätze von Regressionsgleichungen und Neuronale Netze betrachtet. Über die statische Datenverarbeitung hinaus werden auch Ansätze der dynamischen und adaptiven Regelung unbekannter Systeme diskutiert (MRAC - model reference adaptive control).

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Am Ende der Veranstaltung sollen die Teilnehmenden,

- unabhängige von abhängigen Daten und Labels bestimmen können,
- ein Grundverständnis zur Erstellung von Modellen aus Daten besitzen,
- Kostenfunktionen und Lernalgorithmen zur Adaption von Modellparametern anwenden können,
- Modell- und Adaptionsstabilität von Lernalgorithmen analysieren können,
- Modellergebnisse und -aussagen mit Hilfe der Stochastik beurteilen können,
- einfache dynamische und nichtlineare Prozesse adaptiv regeln können.

Literaturliste

J. D. Kelleher, B. Tierney: Data science, MIT Press, Cambridge, 2018.

J. Cleve, U. Lämmel: Data Mining. De Gruyter: Berlin, 2020.

J. Starmer: The StatQuest Illustrated Guide To Machine Learning. StatQuest Publications, 2022.

G. C. Goodwin, K. S. Sin: Adaptive Filtering Prediction and Control. Dover Books on Electrical Engineering, Dover. 1984

G. Schulz: Regelungstechnik, Mehrgrößenregelung - Digitale Regelungstechnik - Fuzzy-Regelung. Oldenbourg, 2002.

H.-J. Reinhardt: Numerik gewöhnlicher Differentialgleichungen, Anfangs- und Randwertprobleme, De Gruyter, 2012.

2.29 Informatik und Umwelt

Informationen über das Modul

Name / engl.	Informatik und Umwelt / Information technology and the environment
Kürzel	INUM4.WP
Verantwortlicher	Prof. Dr. Jürgen Scholz
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Informatik und Umwelt (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	<p>In Gruppenarbeit werden gewonnene Erkenntnisse anschließend präzisiert und für einen INFO-Shop aufbereitet. Aufbauend auf diesen Ergebnissen werden kleine Aufgabenstellungen für Teams von 2-4 Bearbeitern erarbeitet und im Rahmen von Projekten bearbeitet.</p> <p>Am Semesterende ist eine Informatik & Umweltmesse vorgesehen, in der jede Projektgruppe ihren "Messestand" aufbaut und Interessenten die Ergebnisse präsentiert.</p>
Voraussetzungen	Informatik Grundkenntnisse (Programmieren, Grundlagen der Informatik)
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	<p>Portfolioprüfung:</p> <ul style="list-style-type: none">• Ausarbeitung, 35%• Referat, 15%• Mitwirkung am Gesamtprojekt, 50%
--------------	--

Prüfungsnummer	IN 3970393, 2970891 TI 2976713 WI 3975811 IIS 9775170
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Herausforderung unsere Umwelt zu schützen (Luftverschmutzung, Erderwärmung, ...) betrifft jeden. Ein weiterer Aspekt ist es, die Abhängigkeit von Importen - ganz besonders von fossilen - Energieträgern zu reduzieren. Jeder hat seine Verantwortung zu tragen, auch wir als Technische Informatiker, Informatiker und Wirtschaftsinformatiker. Welche Möglichkeiten bieten sich hier für uns Informatiker unseren Beitrag zu leisten? Was können wir bewirken? Das ist das Leitmotto der Veranstaltung "Informatik und Umwelt".

Zunächst erfolgt ein Überblick über den Themenbereich der Informatik und Umwelt. Hierzu wird in die physikalischen/elektrotechnischen Zusammenhänge unter praktischem Aspekt eingeführt. Das ist keine Physik-Vorlesung, sondern das, was man wissen muss um hier als Informatiker arbeiten zu können, zusammengefasst. Also keine Relativitätstheorie nach Einstein, sondern pragmatisch zusammengefasst nach Jürgen Scholz.

Nach dieser Einführung geht es recht schnell in praktische Themen, wo die Studierenden in kleineren Teams selbst kleinere Themenstellungen erarbeiten. Zu gestelltem Material recherchieren die Studierenden das genannte Thema. Sie bearbeiten das Themengebiet und erstellen zu ihren Ergebnissen ein Poster für einen Info-Shop.

Im "Info-Shop" zeigen die Studierenden anhand des Posters das Ergebnis ihres Teams den anderen Teams. Nach Möglichkeit soll die Ausarbeitung zu den Info-Shops und die Durchführung des Info-Shops am selben Tag stattfinden.

Aus den Info-Shop Arbeiten und Themen leiten sich konkrete, semesterübergreifende Projekt-Themenstellungen ab, die ebenso in Teams erarbeitet werden. Das Semesterprojekt kann von praktischen anfassbaren Themen (Bauen einer kleinen Schaltung, die Energie spart, Programme, Apps) bis hin zu theoretischen Auswertungen sein. Eine Liste von Anregungen zu Themenstellungen wird gegeben. Einzige Voraussetzung: der Themenkontext der Vorlesung muss im Thema und der Bearbeitung erkennbar sein.

Nach Möglichkeit werden die Ergebnisse in größerem Rahmen (ggf am Projekttag) vorgestellt.

Besonderheit:

Begleitend zur - und im Rahmen der Veranstaltung sind Vorträge von Referenten aus der Industrie und Behörden geplant, die einige der heute bereits seitens der Industrie betriebenen Ansätze in den verschiedenen Bereichen zeigt.

Am Ende des Semesters ist eine Informatik & Umwelt – Messe geplant, in der die Studierenden ihre Projekte weiteren Interessierten vorstellen.

Die Dokumentation der Ergebnisse der Teams werden am Semesterende zu einem Dokument zusammengebunden.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Der Studierende lernt die Bereiche, in denen die Informatik Einfluss auf die Umwelt hat, kennen.

Der Studierende hat die Fähigkeit, theoretische oder praktische Projekte durchzuführen, also von der Konzeption bis zur Konstruktion kleiner Geräte, einer Software oder wirtschaftliche Abschätzungen oder Systeme zur Abschätzung von Umwelteinfüssen, usw.

Er ist in seinem Informatiker-Leben bei seinen Arbeiten für den Umweltaspekt sensibilisiert.

Literaturliste

wird in der Veranstaltung bekannt gegeben.

2.30 Integrierte Geschäftsprozesse mit SAP ERP

Informationen über das Modul

Name / engl.	Integrierte Geschäftsprozesse mit SAP ERP / Integrated Business Processes with SAP ERP
Kürzel	SAPERP4.WP
Verantwortlicher	Dipl.-Ing. Harald Röser
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Integrierte Geschäftsprozesse mit SAP ERP (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen, Praktika
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 90 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970321, 2970782 TI 3976543, 2976555 WI 3975702 IIS 9775129
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Überblick zu den Komponenten eines ERP-Systems sowie Grundlagen zu wesentlichen Prozessen der Logistik und deren Integration.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden sollten folgendes können:

- die Kernfunktionen von SAP ERP beschreiben
- die Komponenten eines Geschäftsprozesses benennen
- die einzelnen Prozessschritte erläutern
- die im Geschäftsprozess verwendeten Organisationsebenen beschreiben
- die Stammdaten auflisten
- die Integrationsstellen eines Prozesses erkennen

Literaturliste

Wird zu Beginn der ersten Veranstaltung des Moduls bekannt gegeben.

2.31 Interaction Engineering

Information about the module

engl. Name	Interaction Engineering
Code	INTENG4.WP
Coordinator	Prof. Dr. Michael Kipp
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Interaction Engineering (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	The course includes a series of lectures by the lecturer. Students will give oral presentations and work on assignments at home, both individually and in teams. Students will also work on a final team project which engages them in scientific thinking, practical implementation and critical reflection.
Prerequisites	The requirements for this course are solid programming skills, prior experience with working scientifically, a good command of the English language (reading, writing and speaking) and an interest in working both analytically and creatively to develop novel interaction methods.
Usage possibilities	Required elective for bachelor's degree programs: Computer Science and Computer Engineering
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 15 minutes, 25%• Project work, 50%• Written assignment, 15-20 pages, 25%
---	--

Examination number IN 3970326, 2970796
TI 3976567, 2976571

Grading According to § 20 of the APO in the currently valid version.

Content of the module

In the course students will learn about fundamental concepts of human-computer interaction and various research areas that try to improve traditional ways of human-computer interaction by including touch, gesture, facial and bodily actions to make the interaction more intuitive, natural and efficient.

Students will also get to know and apply methods to evaluate interactive systems objectively (measurable aspects) and subjectively (user feedback).

Qualification aims for the module learning objectives/skills

Knowledge

- Fundamentals of human-computer interaction
- Touch interaction
- Gestural interaction
- Tangible interaction
- Proxemic, spatial, full-body interaction
- Cross-device interaction

Skills

- Understanding and presenting a research publication
- Implementing a running prototype of an interactive system
- Applying evaluation methods for an interactive system
- Critically discussing research publications
- Working in a team

Competencies

- Understanding and further developing a research topic
- Informally evaluating a prototype

Reading list

- B. Buxton, S. Greenberg, S. Carpendale, N. Marquardt (2012)** Sketching User Experiences: The Workbook, Morgan Kaufmann, 262 pages.
- B. Albert, T. Tullis (2013)** Measuring the User Experience, 2. Edition, Morgan Kaufmann, 301 pages.
- J. Butler, K. Holden, W. Lidwell (2010)** Universal Principles of Design, Rockport Publishers, 272 pages.

2.32 Interaktive Computergrafik

Informationen über das Modul

Name / engl.	Interaktive Computergrafik / Interactive Computer Graphics
Kürzel	IACOG6.WP
Verantwortlicher	Prof. Dr. Peter Rösch
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt ein Semester. Das Modul wird im Sommersemester angeboten, falls genügend Anmeldungen vorliegen.
Lehrveranstaltungen	Interaktive Computergrafik (6 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen Verwendete Programmiersprachen und Schnittstellen: Python (panda3d und WorldViz Vizard) OpenGL Shading Language (GLSL) JavaScript (babylon.js)
Voraussetzungen	Lineare Algebra (Matrizen, Vektoren, Transformationen)
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h Eine Online-Teilnahme am Präsenzteil ist möglich.

Prüfung

Prüfungsform	Klausur, 90 Minuten, keine Hilfsmittel
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Zusammenfassung

Die Leistung aktueller Hardware ermöglicht die Ausführung anspruchsvoller interaktiver Grafik-Anwendungen nicht nur auf speziell ausgestatteten Rechnern, sondern in zunehmendem Maße auch auf mobilen Geräten. Gleichzeitig können 3D-Inhalte ohne Installation spezifischer Software direkt im Web-Browser präsentiert werden, so dass die Bedeutung der Computergrafik z.B. für die Visualisierung komplexer Inhalte oder für die Präsentation von Produkten weiter steigen wird.

Die Veranstaltung gliedert sich in drei Teile. Zunächst werden grundlegende Methoden und Algorithmen der Computergrafik eingeführt und anhand der plattformunabhängigen Schnittstelle OpenGL praktisch angewendet, wobei die Grafik-Hardware auch direkt mit eigenen Shader-Programmen angesteuert wird.

Ausgestattet mit diesen Grundlagen begeben wir uns in die "Virtuelle Realität" und verwenden die 3x2m große Projektionsfläche im Labor für 3D-Visualisierung in Kombination mit einem optischen Tracking-System, um mit stereoskopisch dargestellten 3D-Modellen zu interagieren. Die verwendete Software "WorldViz Vizard" reduziert dabei den Programmieraufwand erheblich und erlaubt eine Konzentration auf den Aufbau der Szene, die Physik-Simulation und die Interaktion.

Abschließend wird die WebGL-Schnittstelle eingeführt und dazu verwendet, 3D-Inhalte plattformunabhängig im Web-Browser darzustellen.

- Geometrie - Objekte und Transformationen
- Virtuelle Kamera, Projektionen
- Beleuchtung und Schatten
- Texturen und fortgeschrittene Oberflächen-Effekte
- Interaktion mit dem Benutzer
- Shader-Programmierung
- Stereoskopische Ausgabe
- 3D-Tracking
- Physik-Simulation
- Interaktive 3D-Grafik im Web-Browser

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Grundlegende Begriffe der Computergrafik zu definieren.
- Algorithmen zur Darstellung von Szenen zu erklären.
- Komponenten aus Bibliotheken zu Computergrafik-Anwendungen mittlerer Komplexität zu kombinieren.
- Quellcode insbesondere bezüglich der Effizienz zu beurteilen.
- Interaktive Computergrafik-Anwendungen selbständig zu implementieren.

Literaturliste

T. Akenine-Möller et al.: Real-Time Rendering, 4th Ed., CRC Press (2018)

D. Wolff: OpenGL 4 Shading Language Cookbook, 3rd Ed., Packt Publishing (2018)

J.D. Foley, A. van Dam, S.K. Feiner: Computer Graphics – Principles and Practice, Addison Wesley, 3rd Ed., Pearson (2014)

R. J. Rost, J. M. Kessenich, B. Lichtenbelt: OpenGL Shading Language, 3rd Ed., Addison Wesley (2009)

2.33 IT-Consulting

Informationen über das Modul

Name / engl.	IT-Consulting / IT-Consulting
Kürzel	ITC4.WP
Verantwortlicher	Prof. Dr. Stephan Zimmermann
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	IT-Consulting (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht mit begleitenden Übungen und Fallstudien zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützen die Übungen das Selbststudium.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Projektarbeit, 10-25 Seiten, 60%• Präsentation, 10-30 Minuten, 40%
Prüfungsnummer	IN 3970379, 2970877 TI 3976595, 2976688 WI 3975797 IIS 9775131
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Beratungs-Skills sind zentrale Anforderungen an alle, die Informationssysteme und digitale Technologien einführen und weiterentwickeln. Die Beratungsbranche selbst ist ein milliardenschweres Geschäft und zieht viele Hochschulabsolventen an. Aber auch in-house Consultants, die Beratung im eigenen Unternehmen erbringen, sind gefragt. Im Kontext der digitalen Transformation stellt das IT-Consulting daher ein großes Zukunftsthema dar:

- Bei der Analyse und Einführung innovativer Informationstechnologien,
- bei der Verzahnung von Informationssystemen und Geschäftsprozessen und
- beim Management der IT im Unternehmen.

In diesem Modul werden die Techniken, persönlichen Skills und Herausforderungen von IT-Consultants beleuchtet und angewendet:

- Grundlagen, Strukturen und Ziele der Unternehmens- und IT-Beratung
- Leistungsangebote im Bereich IT-Consulting
- Phasen im IT-Beratungsprozess: Projektakquise, Marktrecherche, Projektmanagement, Business Analyse, Ergebnispräsentation
- Analytische Methoden und Techniken in IT-Beratungsprojekten (u.a. Hypothesis-based Problem-solving, Ideation & Design Thinking, Geschäftsmodellanalyse, Reengineering von Geschäftsprozessen & Prozessmodellierung, Analyse von Informationssystemen, Requirements Engineering, Solution Design, ...)
- Methoden des IT-Consultings: Management-Skills, Recherche- und Analysetechniken, Workshop-, Tagungs- und Meeting-Gestaltung, Moderationstechniken, Präsentation, Slide-Deck-Visualisierung
- Profil des IT-Beraters: Know-how, Social & Team Skills

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul können Studierende:

- Zielsetzungen, Abläufe und Herausforderungen von IT-Beratungsunternehmen einordnen.
- Die Aufgaben und Methoden im IT-Consulting diskutieren.
- Projektmanagement-, Business Analyse und Consultingmethoden im Hinblick auf IT-Beratungsprojekte durchführen und anpassen.
- Unternehmensfragstellungen beim Einsatz von Informationssystemen und -technologien analysieren und modellieren.
- Workshops, Tagungen und Meetings in Beratungsprojekten durchführen.
- Beratungsaufträge anhand von Fallstudien planen und organisieren.

Literaturliste

Cadle, James; Paul Debra; Turner Paul (2014): Business Analysis Techniques – 99 Essential Tools for Success (2. Auflage). BCS, The Chartered Institute for IT

Conn, Charles; McLean Robert (2018): Bulletproof Problem Solving. Hoboken, New Jersey: John Wiley & Sons, Inc.

Hamilton, Pamela (2016): The Workshop Book – How to design and lead successful workshops. Pearson

Lippold, Dirk (2020): Grundlagen der Unternehmensberatung (2. Auflage). Berlin/Boston: De Gruyter

Weiss, Alan (2021): The Consulting Bible (2. Auflage), Wiles

Williams, Robin (2017): Non-Designer's Presentation Book, The: Principles for effective presentation design, 2nd Edition, Peachpit Press

2.34 IT-Forensik

Informationen über das Modul

Name / engl.	IT-Forensik / IT Forensics
Kürzel	ITFORE4.WP
Verantwortlicher	Prof. Dr. Kay Werthschulte
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	IT Forensik (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum das Selbststudium.
Voraussetzungen	Vorlesung IT Sicherheit wünschenswert aber nicht Ausschlusskriterium
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 90 Minuten, keine Hilfsmittel
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Einführung in die Digitale Forensik
- Vorgehensmodelle
- Sicherstellung digitaler Spuren
- Analyse digitaler Spuren
- Festplattenforensik
- Windows Forensik
- Arbeitsspeicherforensik
- Netzwerkforensik
- Mobile Forensik
- Malware Analyse
- Präsentation der Beweise vor Gericht
- Rechtliche Aspekte

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Vorlesung Digitale Forensik befasst sich mit der Sicherstellung, Analyse und Präsentation digitaler Spuren nach einem Vorfall. Die Studierenden bekommen dabei einen Überblick über forensische Vorgehensweisen, über IT Angriffe sowie über die zugrundeliegenden Technologien.

Da es sich um eine integrierte Vorlesung handelt, wird das Gehörte direkt in der Vorlesung umgesetzt, wodurch eine enge Kopplung zwischen Theorie und Praxis erreicht wird.

Die Teilnehmer sollten nach der Vorlesung in der Lage sein, festzustellen ob ein Angriff stattgefunden hat und wissen wie man digitale Beweise sicherstellt, analysiert und vor Gericht richtig präsentiert.

Literaturliste

Dan Farmer, Wietse Venema: Forensic Discovery, Addison-Wesley Longman, Amsterdam; Auflage: illustrated edition (13. Januar 2005)

Brian Carrier: File System Forensic Analysis, Addison-Wesley Longman, Amsterdam (7. April 2005)

Harlan Carvey: Windows Forensic Analysis DVD Toolkit, Second Edition, Syngress; 2 edition (June 11, 2009)

Lee Reiber: Mobile Forensic Investigations, McGraw-Hill Education, Auflage: 2., 2019

2.35 IT-Sicherheit

Informationen über das Modul

Name / engl.	IT-Sicherheit / IT Security
Kürzel	ITSICH4.WP
Verantwortlicher	Prof. Lothar Braun Prof. Dr.-Ing. Dominik Merli
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt ein Semester. Das Modul wird regelmäßig sowohl im Wintersemester als auch im Sommersemester angeboten.
Lehrveranstaltungen	IT-Sicherheit (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen und Präsentationen zur Anwendung und Vertiefung der erworbenen Kenntnisse.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Grundlagen der IT-Sicherheit
 - Grundbegriffe
 - Relevante Standards
 - Typische Angriffe
 - Sicherheitsprozesse
 - Analyse von Bedrohungen und Risiken
- Kryptographische Grundlagen
 - Symmetrische Verschlüsselung
 - Hashfunktionen
 - Asymmetrische Kryptographie
 - Schlüsselverwaltung
 - Sicherheitsprotokolle
- Anwendungsbezogene Sicherheit
 - Eingebettete Systeme
 - Netzwerke
 - Web-Anwendungen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Grundbegriffe der IT-Sicherheit zu erklären.
- typische Angriffe zu beschreiben.
- die Methodik der Bedrohungs- und Risikoanalyse auf ein Szenario anzuwenden.
- die Grundlagen kryptographischer Algorithmen darzustellen.
- einfache kryptographische Anwendungen zu implementieren.
- einfache Sicherheitseigenschaften von Netzwerken, Geräten und Web-Anwendungen zu analysieren.
- einfache Sicherheitsmaßnahmen für Netzwerke, Geräte und Web-Anwendungen zu planen.

Literaturliste

- A. Shostack:** "Threat Modeling: Designing for Security", Wiley, 2014
- M. Howard, S. Lipner:** "The Security Development Lifecycle", Microsoft Press, 2006
- C. Paar, J. Pelzl:** "Understanding Cryptography: A Textbook for Students and Practitioners", Springer, 2010
- C. Eckert:** "IT-Sicherheit: Konzepte - Verfahren - Protokolle", Oldenbourg, 2012
- M. Ruef:** "Die Kunst des Penetration Testing", C & L, 2007

2.36 IT Sourcing and Cloud Transformation

Information about the module

engl. Name	IT Sourcing and Cloud Transformation
Code	ITSCT4.WP
Coordinator	Prof. Dr. Arne Mayer
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	IT Sourcing and Cloud Transformation (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar-based instruction at the beginning - Supported by case studies, group discussions and guest lectures. In the further course, work in small groups, in which the students work out the practice-relevant content themselves.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: non-programmable calculator
Examination number	IN 3970380, 2970878 TI 3976596, 2976689 WI 3975798 IIS 9775133
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Offshoring and outsourcing as well as the change from classic IT models to the cloud are a 'must have' for organizations in high-wage countries like Germany. This stems not only from an economic point of view, but also against the background of the permanent shortage of IT specialists. As a result, complexity and demands on the IT of organizations increase significantly. In this module - with a strong focus on relevant, current problems - students are prepared for opportunities and challenges in their future professional life.

The following blocks are covered:

- Off- and nearshoring (regional IT sourcing)
- Outsourcing (external IT sourcing)
- Transformation to the Cloud / Everything as a Service
- Low code platforms as game changers in software development

Qualification aims for the module learning objectives/skills

With successful participation in the module, students can:

- Understand the challenges in today's information management
- Be familiar with and discuss the IT measures and technologies mentioned
- Generate solution proposals for current problems and create implementation approaches

Reading list

Will be announced in the first lecture.

2.37 Klassische Projekttechniken modernisiert

Informationen über das Modul

Name / engl.	Klassische Projekttechniken modernisiert / Classic Project Management Modernized
Kürzel	KLPRO.WP
Verantwortlicher	Prof. Dr. Wolfgang Kowarschick
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Klassische Projekttechniken modernisiert (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht unter Einsatz von Arbeitsblättern zur Vertiefung der erworbenen Kenntnisse.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für alle Bachelorstudiengänge, außer IA
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Studienarbeit, 10 Seiten
Prüfungsnummer	IN 3970371, 2970869 TI 3976598, 2976680 WI 3975789 IIS 9775135
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Zu Beginn der Veranstaltung werden die wesentlichen Begriffe des Critical-Chain-Projektmanagements definiert: Projektziele, Projektbeteiligte, Aufgaben des Managements (Menschenführung, Risikomanagement, Planung, Kontrolle) und Projekterfolg. Nach einer Einführung in das Risikomanagement wird der Projektverlauf näher untersucht: Phasen und Vorgänge, Wasserfall- und Spiralmodell, V-Modell XT. Darauf aufbauend werden verschiedene Schätzmethoden sowie deren Vor- und Nachteile vorgestellt. Anschließend werden gängige Planungstechniken diskutiert: Work Breakdown Structures, Netzpläne, Balkendiagramme, Kostenplanung. Ein Schwerpunktthema ist dabei die Methode der kritischen Kette (an Stelle des kritischen Pfades) und das damit verbundene Puffermanagement (als sehr wichtiger Bestandteil des Risikomanagements). Abschließend werden die Themengebiete „Projektkontrolle anhand des Puffermanagements“ und „Earned-Value-Analyse“ diskutiert.

Parallel zu den klassischen Planungs- und Kontrollthemen wird während des gesamten Semesters immer wieder die Wichtigkeit der Menschenführung betont. Wichtige Aspekte sind hierbei: Führungsstile, Teamarbeit, Motivation und Vermeidung von Druck.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Kenntnisse:

- Die Studierenden kennen die wesentlichen Begriffe und Ziele des Critical-Chain-Projektmanagement.
- Die Studierenden kennen die Unterschiede zwischen klassischen und agilem Projektmanagement.
- Es ist Ihnen bewusst, dass explizites Puffermanagement in beiden Bereichen gewinnbringend eingesetzt werden kann.
- Es ist ihnen bewusst, dass agiles Projektmanagement nur in gewissen Teilbereichen eines Projektes eingesetzt werden kann, das nicht ausschließlich auf Softwareentwicklung basiert.
- Es ist ihnen überdies bekannt, welche typischen Managementfehler häufig für das Scheitern eines Projektes verantwortlich sind.
- Die Dokumentationsarchitektur des V-Modell XT ist den Studierenden bekannt.

Fertigkeiten:

- Die Studierenden können Medienprojekte als Projektmitarbeiter erfolgreich durchführen.
- Die Studierenden können an der Planung eines Projektes mitarbeiten, so dass mit großer Wahrscheinlichkeit alle Projektziele (Dauer, Kosten, Funktionalität, Qualität) erfüllt werden. Insbesondere können sie die Prinzipien des expliziten Puffermanagements gewinnbringend einsetzen.
- Studierende können Projektrisiken abschätzen, geeignete Vorsorgemaßnahmen und, falls nötig, geeignete Gegenmaßnahmen ergreifen.
- Sie können Projektdokumentation gemäß den Vorgaben des V-Modell XT erstellen.
- Sie können Vorgaben des V-Modell XT an konkrete Projekte anpassen (Tailoring).

Kompetenzen:

- Die Studierenden können ihre Entscheidungen, die sie als Projektmitarbeiter treffen, begründen.
- Sie können eine Vielzahl von Projekttechniken kategorisieren und bewerten.

Literaturliste

Für die Vorlesung werden ein sehr umfangreiches Skript sowie digitale Unterlagen zur Verfügung gestellt.

2.38 Konzepte der Datenbanktechnologie

Informationen über das Modul

Name / engl.	Konzepte der Datenbanktechnologie / Concepts of Database Technology
Kürzel	KDBT4.WP
Verantwortlicher	Prof. Dr. Michael Predeschly
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Konzepte der Datenbanktechnologie (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördern das Praktikum das Selbststudium.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970397, 2970895 TI 3976545, 2976717 WI 3975815 IIS 9775175
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Vorlesung stellt verschiedene Konzepte vor, die in unterschiedlichen Arten von Datenbanken Verwendung finden. Hierbei werden unterschiedliche Architekturen vorgestellt.

Ein Fokus der Veranstaltung liegt in der Speicherung von Daten. Hierbei werden folgende Themen behandelt:

- Speicherstrukturen und Zugriffspfade
- Pufferverwaltung
- Einbringungsstrategien
- Indexe

Ein zweiter zentraler Aspekt widmet sich der Konsistenz von Datenbanken mittels:

- Transaktionen
- Concurrency Control
- Serialisierbarkeit
- Recovery
- Schema Migration

Darüber hinaus wird das Themengebiet der Anfragenoptimierung sowohl algebraisch als auch algorithmisch betrachtet.

Abschließend werden Konzepte des Datenschutzes und der Datensicherheit in Datenbanken beleuchtet.

Es werden dabei sowohl theoretische Grundlagen vermittelt als auch deren Anwendung in der Praxis aufgezeigt und umgesetzt.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Studierende erhalten einen Überblick über das Themengebiet verschiedener Datenbanktechnologien. Nach erfolgreicher Teilnahme am Modul erlangen die Studierenden folgende Fähigkeiten:

- Kenntnis, der für die Implementierung von Datenbanksystemen wichtigen Architekturprinzipien
- Verständnis von Datenstrukturen und Algorithmen sowie die Fähigkeit diese vergleichen, analysieren, bewerten und implementieren zu können
- Vertieftes Verständnis des Aufbaus und der internen Strukturen eines komplexen Softwaresystems.
- Optimierung der Arbeitsweise von Datenbanksystemen
- Planung eines Datenbanksystems und dessen sicherer Betrieb
- Konzepte und Techniken des Datenschutzes, als auch der Datensicherheit

Literaturliste

Eine Literaturliste wird in der Veranstaltung bereitgestellt.

2.39 Künstliche Intelligenz in sicherheitskritischen Anwendungen

Informationen über das Modul

Name / engl.	Künstliche Intelligenz in sicherheitskritischen Anwendungen / Artificial intelligence in safety-critical applications
Kürzel	KISICH4.WP
Verantwortlicher	Dr. Marc Zeller (Siemens AG, München)
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Künstliche Intelligenz in sicherheitskritischen Anwendungen (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitende Übungen
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: Taschenrechner
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Grundlagen der funktionalen Sicherheit
 - Grundbegriffe
 - Safety Engineering Life Cycle
 - Risikoanalyse und Sicherheitseinstufung
 - Sicherheitsnachweisführung und Zertifizierung
- Sichere Softwareentwicklung in unterschiedlichen Industriedomänen
 - Sicherheitskonzepte und Fehleranalysemethoden
 - Test- und Verifikationsmethoden für sichere Software
 - Relevante Normen und deren praktische Anwendung
- Sichere und robuste Artificial Intelligence (AI)
 - AI und ML = Software 2.0
 - Relevante Normen
 - Safety Of The Intended Functionality (SOTIF)
 - Analysemethoden von AI/ML-Modellen bzgl. Robustheit, Unsicherheit und Transparenz
 - Out-of-Distribution Erkennung und Laufzeitüberwachung
 - Iterative und agile Entwicklung (MLOps) und Sicherheit

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Grundbegriffe der funktionalen Sicherheit für Software-, Künstliche Intelligenz (KI) und Machine Learning (ML) basierte System zu erklären
- Aspekte der funktionalen Sicherheit sowie SOTIF-Aspekte autonomer Systeme in unterschiedlichen Industriedomänen zu beschreiben
- Methoden der Risiko- und Fehleranalyse anzuwenden und Anforderungen an die Sicherheit des Systems abzuleiten
- Test- und Verifikationsmethoden für sichere Software anzuwenden
- Grundlagen von Robustheits-, Unsicherheits- und Transparenzanalysen an KI-/ML-Modellen darzustellen
- Safety-Konzepte für die Entwicklung und den Betrieb von sicheren und robusten autonomen Systemen zu erstellen

Literaturliste

Bücher:

Laprie, Jean-Claude: Dependability: Basic concepts and terminology. Springer Vienna, 1992.

Koopman, Phil: How Safe is Safe Enough?: Measuring and Predicting Autonomous Vehicle Safety. Carnegie Mellon University, 2022.

Wolfgang Ertel: Grundkurs Künstliche Intelligenz – Eine praxisorientierte Einführung, Springer Verlag, Wiesbaden.

Normen:

- Automotive (ISO 26262-6)
- Railway (EN 50128, EN 5065, SIRF)
- Avionics (DO-178C)
- Medical Devices (IEC 62304)
- Industry Automation (ISO 13849)
- Artificial Intelligence (EU AI Act, UL4600, VDE-AR-E_2842-61-5)

2.40 Lean IT & Enterprise Architecture

Information about the module

engl. Name	Lean IT & Enterprise Architecture
Code	LEANIT4.WP
Coordinator	Prof. Dr. Stephan Zimmermann
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Lean IT & Enterprise Architecture (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Lecture and seminar lessons with laboratory exercises and case studies to apply the knowledge acquired. In addition, the exercises support self-study.
Prerequisites	The requirements for this course are a basic command of the English language, and an interest in better managing IT organizations and enterprise architectures.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Project work, 15-30 pages
Examination number	IN 3970394, 2970892 TI 3976600, 2976714, WI 3975812, IIS 9775171
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

IT in companies is becoming more and more important and complex. A numerous and increasing number of applications, systems and IT services used in business processes and delivered by IT organizations substantiates this development.

Lean IT and Enterprise Architecture Management (EAM) help companies to address related challenges. While Lean IT uses lean principles to develop and manage IT products and services with the central concern to eliminate waste in the context of IT that adds no value for the customer or user, EAM describes the management practice to transform the IT landscape by defining, communicating, and using a coherent set of strategies and guidelines.

In this course students will learn about the fundamental concepts of lean IT and enterprise architectures, and how these two topics connect. They also get to know techniques to develop strategies, analyze waste and work in value streams, and build business, information system and technology architectures.

Students will play several lean games to increase their lean mindset and solve several case studies regarding enterprise architecture challenges in practice. Supported by the novel “The Phoenix Project” they will have an additional touchpoint to practical challenges.

Knowledge focus:

- Lean IT concepts (value, waste, value streams, pull, flow)
 - Value stream mapping
 - The Four Types of Work
 - Kanban-Boards
- Enterprise Architecture concepts: Business, Information System and Technology Architecture
 - Business Capability Management
 - IT Portfolio Management
 - The Open Group Architecture Framework (TOGAF)
 - Visualization of IT landscapes

Qualification aims for the module learning objectives/skills

After successful participation in the module, the students can:

- illustrate waste, work, and Kanban in a lean IT context
- apply value stream mapping for IT services & products
- demonstrate competencies with the application of EA methods and IT landscape modelling
- apply business capability management and IT portfolio techniques
- illustrate enterprise architecture frameworks
- solving practical case studies and scenarios
- articulate course related ideas and concepts in English.

Reading list

Ahlemann, F., Stettiner, E., Messerschmidt, M., Legner, C. (2012): Strategic Enterprise Architecture Management Challenges, Best Practices, and Future Developments, Springer-Verlag Berlin Heidelberg.

Kim, Gene; Behr, Kevin; Spafford, George (2013) : The Phoenix Project – A novel about IT, DevOps and helping your business win, IT Revolution Press.

Lankhorst M. (2013) : Enterprise architecture at work: Modelling, communication, and analysis. Springer, Berlin.

Peppard J., Ward J. (2016) : The strategic management of information systems: Building a digital strategy. Wiley, Chichester, West Sussex.

The Open Group (2018), The Open Group Architectural Framework (TOGAF) Version 9.2. The Open Group, Reading, UK.

2.41 Linux LPIC

Informationen über das Modul

Name / engl.	Linux LPIC / Linux LPIC
Kürzel	LINLP14.WP
Verantwortlicher	Dieter Thalmayr
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird als Blockveranstaltung regelmäßig sowohl im Wintersemester als auch im Sommersemester angeboten. Hinweis: Die Veranstaltung WPF Linux LPIC wird als Blockveranstaltung an 6 Tagen angeboten. Die Prüfung findet außerhalb des üblichen Prüfungszeitraums in angemessenem Abstand zu der Blockveranstaltung statt. Als Prüfungstag ist ein Samstag vorgesehen.
Lehrveranstaltungen	Linux LPIC (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN -, - TI -, - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Inhalte von Linux LPIC sind an den Themen angelehnt, die das Linux Professional Institute für die Prüfungen LPI 101 und 102 Lehrplan angelehnt:

- Systemarchitektur
- Installation und Paketverwaltung
- GNU- und UNIX-Kommandos
- Geräte, Linux-Dateisystem, Filesystem, Hierarchy Standard
- Datenverwaltung und Rechtekonzept
- Einfache Administrative Aufgaben
- Erlernen eines Linux-Editors
- Paketverwaltung
- GNU- und UNIX-Kommandos
- Shells und Grundlagen der Skriptprogrammierung
- Administrative Aufgaben
- Netzwerk-Grundlagen
- Einrichten eines Netzwerkdienstes
- Sicherheit

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Teilnehmer sollen sowohl Einblicke in die Funktionsweise von GNU/Linux bekommen, als auch die fortgeschrittene Bedienung und die grundlegende Administration von Linux-Rechnern lernen. Am Ende des Blocks können die Teilnehmer optional eine "Linux Professional Institute"(LPI) Prüfung ablegen, um sich ihr Wissen mit einem in der Wirtschaft angesehenen „LPIC“ Zertifikat bestätigen zu lassen.

Literaturliste

Schulungsmaterial der Firma tuxcademy:

www.tuxcademy.org (kostenlos)

Harald Maassen, LPIC-1. Sicher zur erfolgreichen Linux Zertifizierung, Galileo Computing, jeweils neuste Auflage.

(wird in der Vorlesung bekannt gegeben)

Weitere Informationen:

<http://hhoegl.informatik.hs-augsburg.de/hhweb/lpic>

<http://www.lpice.eu/de>

2.42 Methoden der KI

Informationen über das Modul

Name / engl.	Methoden der KI / Artificial Intelligence
Kürzel	METHKI6.WP
Verantwortlicher	Prof. Dr. Thomas Rist
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Methoden der KI (4 SWS) zugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	<p>Seminaristischer Unterricht zur Vermittlung theoretischer Grundlagen konzeptioneller Lösungsansätze, die anhand ausgewählter Problemstellungen mit den Studierenden gemeinsam bearbeitet werden.</p> <p>In einem begleitenden Praktikum bearbeiten die Teilnehmer selbstständig eine konkrete Anwendungsaufgabe (z.B. aus dem Bereich Maschinelles Lernen und Data Mining, Robotik, Expertensystem, Spiele-KI).</p> <p>Die Unterrichtsart ist Vorlesung und Praktikum.</p>
Voraussetzungen	Grundlagen der Informatik des Grundstudiums und Vertrautheit mit einer Programmiersprache (z.B. Java, Python, C++)
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	ePortfolio
Prüfungsnummer	IN -, - TI -, - WI - IIS -

Inhalte des Moduls

Grundlagen

- KI- Begriff aus wissenschaftlicher, technischer und gesellschaftlicher Perspektive
- starke versus schwache KI
- Modellierung intelligenter Fähigkeiten in technischen Systemen

Problemlösung als Suchaufgabe

- Modellierung von Anwendungsproblemen
- Suchverfahren (u.a. A*, MinMax, Strategiespiele)
- Modellierung mit Constraints und Constraint Solver

Wissensbasierte Systeme

- Regelsysteme, Expertensysteme
- Logisches Schlussfolgern, Logik-Kalküle, SAT-Solver
- Wissensrepräsentation, ontologisches Modellieren
- Verfahren zur Handlungsplanung
- Probabilistisches Schließen, Bayes-Netzze, Fuzzy-Inferenz

Lernfähige Systeme, Maschinelles Lernen und Data Mining

- Grundbegriffe: überwachtes / unüberwachtes Lernen, symbolische / nicht-symbolische Ansätze
- Clustering, Klassifikation, Knowledge-Discovery
- Reinforcement Learning
- Neurocomputing und künstliche Neuronale Netze
- Ansätze zum Deep Learning

Ausblick auf aktuelle und sich abzeichnende Forschungsfelder

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Zu den behandelten Themenstellungen werden Software-Werkzeuge bzw. Bibliotheken vorgestellt, mit denen praktische Problemstellungen bearbeitet werden können.

- verfügen über einen fundierten Überblick zu gängigen KI Methoden und KI-Techniken,
- kennen typische Anwendungsfelder, in denen KI-Techniken zum Einsatz kommen,
- sind in die Lage, ausgewählte Problemstellungen mit dafür passenden KI-Methoden zu bearbeiten.

Literaturliste

Stuart Russel, Peter Norvig: Künstliche Intelligenz Pearson Studium – IT, Gebundene Ausgabe, 2012.

Wolfgang Ertel: Grundkurs Künstliche Intelligenz: Eine praxisorientierte Einführung. 4. Auflage, Springer Verlag 2016.

Jürgen Cleve, Uwe Lämmel: Data Mining. De Gruyter Studium, Taschenbuch 2014.

Peter Buxmann, Holger Schmidt (Hrsg.): Künstliche Intelligenz: Mit Algorithmen zum wirtschaftlichen Erfolg, Springer Gabler, 2018.

Weitere Literatur (darunter aktuelle Fachpublikationen) wird jeweils passend zu den besprochenen Themen während der Vorlesung empfohlen.

2.43 Mobile Robotik

Informationen über das Modul

Name / engl.	Mobile Robotik / Mobile Robots
Kürzel	MOBRO2.WP
Verantwortlicher	Prof. Dr. Constantin Wanninger
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Mobile Robotik (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und abschließende Blockveranstaltung zur Einführung in die mobile Robotik
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 2, CPs: 5, Präsenzzeit: 30 h, Selbststudium: 120 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Präsentation, 5 Minuten, 3 Folien, 25%• Klausur, 45 Minuten, (closed book), 50 %• Projektarbeit, Vorführung / Dokumentation / Modelle, 25%
Prüfungsnummer	IN -, - TI - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Wahlpflichtmodul kombiniert theoretische Vorlesungen mit einer schriftlichen Prüfung und praxisorientierte Projekte im Blockformat, um ein umfassendes Verständnis der behandelten Themen zu gewährleisten. Die Teilnehmer erwerben sowohl tiefgehendes Fachwissen als auch praktische Erfahrung, die sie optimal auf zukünftige Herausforderungen vorbereitet. Folgende Inhalte werden unter anderem behandelt:

- Grundlagen der mobilen Robotik
- Elektronikgrundlagen für mobile Roboter
- Roboernavigation
- Steuerungsalgorithmen
- Autonomie und Entscheidungsfindung

Abschließend wird ein kleines Projekt mit mobilen Robotern realisiert.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Das Wahlpflichtmodul vermittelt den Studierenden fundiertes Fachwissen im Bereich der mobilen Robotik und die Fähigkeit, dieses eigenständig und kreativ auf ein vorgegebenes Projekt anzuwenden. Sie lernen, komplexe Fragestellungen zu analysieren und innovative Lösungen zu entwickeln, die sie im Rahmen des Projekts umsetzen. Dabei stärken sie ihre Selbstorganisation, Teamarbeit sowie Kommunikationsfähigkeiten.

Literaturliste

Online Dokumentation der Arduino Plattform, <https://www.arduino.cc/>

Banzi, Massimo: Getting Started with Arduino. Maker Media, Inc. (2022).

Purdum, Jack: Arduino C, Springer (2012).

Maier, Helmut: Grundlagen der Robotik, VDE Verlag GmbH (2016).

2.44 Mustererkennung und maschinelles Lernen

Informationen über das Modul

Name / engl.	Mustererkennung und maschinelles Lernen / Pattern recognition and machine learning
Kürzel	MKML4.WP
Verantwortlicher	Prof. Dr.-Ing. Alexandra Teynor
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Mustererkennung und maschinelles Lernen (2 SWS) zugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse.
Voraussetzungen	Ausreichende Mathematikgrundlagen (lineare Algebra, Statistik)
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970344, 2970837 TI 3976548, 2976602 WI 3975752 IIS 9775140
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Grundlagen der Mustererkennung
- Vorverarbeitung und Merkmalsextraktion
- Performanzmaße
- Einfache Klassifikatoren (z.B. Minimum-Distanz Klassifikatoren)
- Probabilistische Klassifikatoren
- Unüberwachtes Lernen / Clustering
- Neuronale Netze
- Deep Learning Ansätze

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage

- relevante Basistechniken der Mustererkennung zu verstehen
- geeignete Merkmale für die Weiterverarbeitung auszuwählen, zu extrahieren und/oder zu kombinieren
- für gegebene Klassifikationsprobleme geeignete Klassifikatoren auszuwählen und anzuwenden
- Clustering-Algorithmen zur sinnvollen Gruppierung von Daten anzuwenden
- die Leistungsfähigkeit von Mustererkennungssystemen auf Grund von anerkannten Leistungsmerkmalen zu vergleichen

Literaturliste

R. Duda et al., „Pattern classification“, Wiley, 2000

C. M. Bishop, „Pattern recognition and Machine learning“, Springer, 2006

T. Hastie et al.: „The Elements of Statistical Learning“, Springer 2011

Aurelien Geron: Hands-On Machine Learning with Scikit-Learn & TensorFlow, O'Reilly, 2017

2.45 Network Penetration Testing

Informationen über das Modul

Name / engl.	Network Penetration Testing / Network Penetration Testing
Kürzel	NETP.WP
Verantwortlicher	Dr. Lothar Braun
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Network Penetration Testing (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Vorlesung, Übung, Studienarbeit
Voraussetzungen	Kenntnisse über <ul style="list-style-type: none">• IT-Sicherheit• Netzwerke• Linux von Vorteil (aber nicht notwendig)
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: 1 DIN-A4-Seite handgeschrieben
Prüfungsnummer	IN 3970358, 2970855 TI 3976602, 2976666 WI 3975773 IIS 9775141
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Planung von Penetration Tests für Netzwerke
- Erstellung von Berichten
- Informationsgewinnung im Netzwerk
 - Techniken zur Erkennung von Maschinen und Diensten in Netzwerken mit gängigen Werkzeugen
 - Untersuchung von Angriffsoberflächen von Netzwerkdiensten
 - Identifikation von potentiellen Schwachstellen in Netzwerkdiensten
- Angriffe auf Netzwerkdienste
 - Passwortangriffe
 - Angriffe auf Web-Anwendungen
 - Analyse, Anpassung und Verwendung von Exploits
 - Buffer-Overflow Exploits
 - Entwicklung von Scripten zur Durchführung von Angriffen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden erwerben Kenntnisse über die Durchführung von Penetration Tests in Computernetzwerken.

Studierende lernen die Anwendung von Techniken zur Informationsgewinnung im Netzwerk. Sie kennen die relevanten Techniken zur Identifikation von Schwachstellen.

Die Studierenden lernen die Techniken zur Durchführung von Angriffen zur Demonstration gefundener Schwachstellen kennen, und sind in der Lage diese mittels bekannter Tools anzuwenden. Sie sind in der Lage Handlungsempfehlungen zur Beseitigung der Schwachstellen zu geben.

Literaturliste

Georgia Weidman: Penetration Testing: A Hands-On Introduction to Hacking, No Starch Press, 2014

Google Hacking for Penetration Testers, Third Edition, Syngress, Dezember 2015

Script

2.46 Neuronale Netze und Deep Learning

Informationen über das Modul

Name / engl.	Neuronale Netze und Deep Learning / Neural Networks and Deep Learning
Kürzel	NNDL4.WP
Verantwortlicher	Prof. Dr. Michael Kipp
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Neuronale Netze und Deep Learning (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Vorlesung mit Praxisanteilen und wöchentlichen Aufgaben zur Anwendung und Vertiefung der erworbenen Kenntnisse.
Voraussetzungen	Grundlagen der Programmierung und der Mathematik wie sie in den ersten zwei Semestern der Informatik-Studiengänge vermittelt werden.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	Wirtschaftsinformatik, Informatik, Technische Informatik: SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h Interaktive Medien: SWS: 4, CPs: 8, Präsenzzeit: 60 h, Selbststudium: 180 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: Taschenrechner
Ergänzende Hinweise zur Prüfungsform	für Interaktive Medien Portfolioprüfung: <ul style="list-style-type: none">• Klausur, 60 Minuten, Hilfsmittel: Taschenrechner, 70%• Präsentation, 10 Minuten, 30%

Prüfungsnummer	IN 3970367, 2970865 TI 3976603, 2976676 WI 3975785 IIS 9775142
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Einführung in das maschinelle Lernen
- Grundlagen Neuroner Netze (Feedforward-Netze)
- Training und Evaluation (Backpropagation, Hyperparameter, Optimierung)
- Erstellung, Training und Evaluation Neuroner Netze in Python (Tensorflow/Keras)
- Konvolutionsnetze am Beispiel der Bilderkennung
- Netzwerkarchitekturen
- Rekurrente Neuronale Netze (GRU und LSTM) am Beispiel der Sprachverarbeitung
- Transformer-Netze

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Aufbau und Funktionsweise Neuroner Netze mathematisch zu beschreiben
- Verschiedene Typen und Architekturen Neuroner Netze und ihre Einsatzgebiete zu unterscheiden
- Für vorgegebene Datensätze in einer Umgebung wie Jupyter Notebook die Daten vorzuverarbeiten, geeignete Netze zu wählen, zu erzeugen, zu trainieren und zu bewerten
- Mit Standardbibliotheken wie TensorFlow, Keras oder PyTorch datenbasiert Probleme zu lösen mit Hilfe von Hyperparameter-Tuning, Visualisierung und systematischer Evaluation

Literaturliste

- M. Kipp (2023):** Neuronale Netze und Deep Learning, Onlineskript unter <https://michaelkipp.de/deeplearning>
- F. Chollet (2021):** Deep Learning With Python, 2nd Edition. Manning Publications.
- R. Schwaiger, J. Steinwendner (2019):** Neuronale Netze programmieren mit Python. Rheinwerk Computing.
- M. Ekman (2021):** Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow. Addison-Wesley.

2.47 NoSQL

Informationen über das Modul

Name / engl.	NoSQL / NoSQL
Kürzel	NoSQL4.WP
Verantwortlicher	Prof. Dr. Michael Predeschly
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	NoSQL (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum das Selbststudium.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Projektarbeit, 75%• Präsentation, 15 Minuten, 25%
Prüfungsnummer	IN 3970383, 2970881 TI 3976549, 2976697 WI 3975801 IIS 9775143
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Vorlesung stellt die Entwicklungen im Bereich der NoSQL-Datenbanken in den Mittelpunkt. Es werden verschiedene Arten von NoSQL-Datenbanken und deren jeweilige Besonderheiten besprochen.

Neben der praktischen Beschäftigung mit unterschiedlichen NoSQL-Systemen stehen die zugrunde liegenden theoretischen Konzepte im Vordergrund.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Studierende erhalten einen Überblick über das Themengebiet der NoSQL-Datenbanken. Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- die Notwendigkeit von NoSQL-Datenbanken zu erkennen und die Sinnhaftigkeit des Einsatzes zu beurteilen.
- verschiedenste NoSQL-Datenbanken zu unterscheiden und sie nach dem jeweiligen Einsatzzweck zu klassifizieren
- eine NoSQL-Datenbank zu entwerfen und zu installieren
- Anfragen, in allen Stufen eines CRUD-Zyklus, an eine ausgewählte NoSQL-Datenbank zu stellen

Literaturliste

Eine Literaturliste wird in der Veranstaltung bereitgestellt.

2.48 Objektorientierte Programmierung mit Python

Informationen über das Modul

Name / engl.	Objektorientierte Programmierung mit Python / Object-oriented programming with Python
Kürzel	OPPYTH4.WP
Verantwortlicher	Prof. Dr.-Ing. Thomas Kirchmeier
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Objektorientierte Programmierung mit Python (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übung, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Elektronische Prüfung, 120 Minuten, alle Hilfsmittel, eigener Laptop, inkl. Chat-GPT
Prüfungsnummer	IN 3970403, 2970901 TI 3976626, 2976727 WI 3975821 IIS 9775181
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Generell, um eine Programmiersprache zu verstehen und eigenständig anwenden zu können ist Erfahrung notwendig. Dies soll durch die zahlreichen Programmieraufgaben und Kontrollfragen(Moodle-Quiz) erreicht werden. Für jeden Vorlesungsabschnitt werden ein oder mehrere Lehrvideos zur Verfügung gestellt (durchschnittlich 45 Minuten). Diese gilt es eigenständig zu bearbeiten (nicht einfach ansehen, sondern das Wichtigste beispielsweise aufzuschreiben) und anschließend sind die Kontrollfragen (Moodle-Quiz) zu beantworten. Zuletzt wird die Transferaufgabe bearbeitet. In den Lehrvideos wird ein Adressmanager sukzessive aufgebaut. Die dabei gelernten Programmieraspekte gilt es auf einen Börsenmanager zu transferieren. Dies schult das Lesen und Verstehen fremden Codes, den Transfer auf die eigene Aufgabenstellung und die Interpretationen der Fehlermeldungen des Python-Interpreters bei der Programmierung. Diese drei Kompetenzen sind auch bei einer Problemsuche im Internet und deren Transfer erforderlich. Inhaltlich werden die folgenden Aspekte adressiert:

- Datenstrukturen und geschachtelte Strukturen
- Objektorientierte Programmierung
- Vererbung und Komposition
- Anwendung von Python-Bibliotheken
 - os (Operating System)
 - sys (System)
 - datetime (Datum und Zeit)
 - urllib (Datenakquise aus dem Internet)
 - argparse (Programmargumente)
 - logging (Datenlogger zum Debuggen)
 - time, threading (Effizientere Programmgestaltung)
- Anwendung von Drittanbietermodulen mittels virtueller Umgebung (venv)
 - numpy (Mathematikbibliothek)
 - matplotlib (Plots und Visualisierung)
 - pandas (Datenverwaltung)
 - read, write von Textdateien

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Kenntnisse:

- Programmiersprache Python
- Objektorientierte Programmierung
- Datenstrukturen und Strukturierungskonzepte
- Organisation und Dateienstruktur kleiner Softwareprojekte
- Anwendung verschiedener Python Bibliotheken

Fertigkeiten:

- Einschätzung des Strukturierungsansatzes für Daten
- Lesen und Interpretation von Code
- Erweiterung des eigenen und fremden Codes
- Interpretation von Programmfehlern und Analyse der Fehlerursache

Kompetenzen:

- Einlesen von Rohdaten und deren Vorverarbeitung
- Strukturierung von Daten im Sinne der Aufgabe für eine zielgerichtete Weiterverarbeitung
- Verarbeitung und Auswertung der Daten durch die Erstellung von Funktionen
- Weiterführende Strukturierung von Funktionen und Daten in Objekten

Literaturliste

Online documentation of the Python programming language. URL: <https://docs.python.org>

Eric Matthes: Python Crash Course. 3. Aufl. No Starch Press, Incorporated, 2019. ISBN: 9781718502703

Johannes Ernesti und Peter Kaiser: Python 3. Rheinwerk Verlag GmbH, 2015. 1126 S. ISBN: 3836291290.

URL: https://www.ebook.de/de/product/44876051/johannes_ernesti_peter_kaiser_python_3.html

Thomas Theis: Einstieg in Python. Ideal für Programmieranfänger geeignet. Galileo Press GmbH, 2014. ISBN: 9783836228619.

Michael Weigend: Python 3 - Lernen und professionell anwenden. mitpVerlag, 2016. ISBN: 9783826694561.

2.49 Open-Source Softwareentwicklung

Informationen über das Modul

Name / engl.	Open-Source Softwareentwicklung / Open Source Software Development
Kürzel	OSSWE4.WP
Verantwortlicher	Prof. Dr. Hubert Högl
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Open-Source Softwareentwicklung (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Fernklausur mit Videoaufsicht, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970317, 2970742 TI 3976604, 2976526 WI 3975662 IIS 9775144
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Geschichtliche Entwicklung der freien Software
- Das GNU Projekt
- Open-Source Software
- Produktion von freier Software
- Rechtliche Aspekte von freier Software
- Wichtige freie Projekte
- Das Open-Source Prinzip in anderen Bereichen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

- Verständnis der historischen Entwicklung von freier Software zu „Open-Source“ Software.
- Einblicke in die typischen Werkzeuge zur Entwicklung von freier Software.
- Kenntnis der kollaborativen Techniken, um bei einem freien Projekt mitzumachen.
- Fähigkeit, ein eigenes freies Projekt zu beginnen.
- Überblick über freie Programme aus den wichtigsten Gebieten.
- Kenntnisse im Bereich der Open-Source Lizenzen.

Literaturliste

Volker Grassmuck, Freie Software zwischen Privat- und Gemeineigentum, Bundeszentrale für politische Bildung.

<http://freie-software.bpb.de>

Karl Fogel, Producing Open Source Software. How to Run a Successful Free Software Project, O'Reilly 2005, 302 pages. Das Buch erschien unter der Creative Commons Attribution-ShareAlike Lizenz und ist somit auch frei erhältlich.

<http://producingoss.com>

Open-Sources, Voices from the Open-Source Revolution, O'Reilly 1999.

<http://oreilly.com/openbook/opensources/book/>

Joseph Feller, Perspectives on Free and Open Source Software, MIT Press, 2005.

<https://mitpress.mit.edu/books/perspectives-free-and-open-source-software>

Material von der Website "Teaching Open-Source"

<http://teachingopensource.org>

Jono Bacon, The Art of Community, O'Reilly, 2nd edition, 2012

<http://www.artofcommunityonline.org>

Greg Wilson, The Architecture of Open-Source Applications

<http://aosabook.org/en/index.html>

Hinweise: [Homepage der Veranstaltung](#):

<http://elk.informatik.hs-augsburg.de/hhweb/oss/index.html>

2.50 Praktische Robotik mit Matlab

Informationen über das Modul

Name / engl.	Praktische Robotik mit Matlab / Practical Robotics with Matlab
Kürzel	PRRO.WP
Verantwortlicher	Prof. Dipl.-Ing. Georg Stark
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird regelmäßig im Wintersemester und im Sommersemester als Blockveranstaltung angeboten.
Lehrveranstaltungen	Praktische Robotik mit Matlab (4 SWS) dazugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher und englischer Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum mit gruppenbezogenen Programmierübungen. Durch deren enge Verzahnung wird ein vertieftes Lernen der erworbenen Kenntnisse erreicht.
Voraussetzungen	Grundlagen der Mathematik und Programmierung
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 7.5, Präsenzzeit: 90 h, Selbststudium: 135 h, Gesamtaufwand: 225 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970354, 2970850 TI 3976550, 2976660 WI 3975768 IIS 9775146
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Einführung in die Robotik

- Definitionen, Praktische Robotik
- Roboterklassen und ihre Einsatzgebiete

Robotermathematik I

- Einfache geometrische Elemente
- Lineare Abbildungen

MATLAB-Programmiertechniken I

- Einfache Verfahren der Robotermathematik
- Einführung in die Funktionsbibliothek ROBOMATS

Modellierung und Implementierung von einfachen kinematischen Modellen

- Vorwärtstransformation
- Rücktransformation

Einführung in die Anwendungsprogrammierung einer modernen Roboter-Industriesteuerung Zukünftige Entwicklung

Praktikum

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Die einzelnen Roboterarten und ihre Einsatzgebiete zu beschreiben,
- Einfache MATLAB-Programme zu entwickeln,
- Mit Hilfe von MATLAB kinematische Robotermodelle zu implementieren,
- Einfache Roboter-Anwendungsprogramme zu entwickeln,
- Die Anforderungen der Industrie an moderne Robotersteuerungen zu erklären,
- Die Methoden der Praktischen Robotik auf allgemeine mechatronische Systeme zu übertragen und anzuwenden.

Literaturliste

Verwendete Literatur

Stark G.: Robotik mit Matlab. Hanser, 2009.

http://www.hs-augsburg.de/stark/robotik_mit_matlab/

Dieses Buch sollte beschafft werden, da die Vorlesung größtenteils darauf basiert.

Weiterführende Literatur

Einführung in die Robotik, Anwendungen

Craig, J. J.: Introduction to Robotics. Pearson Education, 2005.

Haun, M.: Handbuch Robotik. Programmieren und Einsatz intelligenter Roboter. Springer, 2007.

Hesse, S.: Grundlagen der Handhabungstechnik. Hanser, 2006.

Grundlagen der Robotermathematik

Hoffmann, A.; Marx, B.; Vogt, W.: Mathematik für Ingenieure. Pearson Education, 2005.

Papula, L.: Mathematik für Ingenieure und Naturwissenschaftler Bd. 1/2. Vieweg, 2001

Papula, L.: Mathematische Formelsammlung für Ingenieure und Naturwissenschaftler. Vieweg, 2006.

Programmieren mit MATLAB, Fehlerbehandlung und Optimierung

Beucher, O.: Matlab und Simulink. Grundlegende Einführung für Studenten und Ingenieure in der Praxis. Pearson Education, 2006.

Schweizer, W.: MATLAB kompakt. Oldenbourg, 2006.

Stein, U.: Einstieg in das Programmieren mit Matlab. Hanser, 2007.

Kinematische Struktur, Bahnsteuerung

Corke, P.: Robotics, Vision and Control. Springer, 2017.

Siegert, H.-J.; Boncione, S.: Programmierung intelligenter Roboter. Springer 1996.

Vidyasagar, M.; Spong, M.W.; Hutchinson, S.: Robot Modeling and Control. John Wiley & Sons, 2006.

Weber, W.: Industrieroboter. Methoden der Steuerung und Regelung. Hanser, 2002.

2.51 Process Intelligence

Informationen über das Modul

Name / engl.	Process Intelligence / Process Intelligence
Kürzel	PRCINT4.WP
Verantwortlicher	Prof. Dr. Wolfgang Kratsch
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Process Intelligence(4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum, in welchem Case Studies über das Semester in Kleingruppen bearbeitet werden.
Voraussetzungen	Programmier-Grundkenntnisse von Vorteil
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Projektarbeit, 10-25 Seiten, 60%• Präsentation, 20-30 Minuten, 40%
Ergänzende Hinweise zur Prüfungsform	
Prüfungsnummer	IN 3970398, 2970896 TI 3976551, 2976718 WI 3975816 IIS 9775176
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Modul „Process Intelligence“ vermittelt den Studierenden grundlegende Konzepte und fortgeschrittene Techniken im Bereich des datengetriebenen Prozessmanagements. Die Studierenden lernen, wie sie Geschäftsprozesse mithilfe von Technologien wie Process Mining, Predictive Process Monitoring, Context-Aware Process Mining und Robotic Process Automation analysieren, optimieren und automatisieren können.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme an diesem Modul werden die Studierenden in der Lage sein,

- Technologien aus dem Bereich Process Intelligence zur Optimierung der Prozesse einzuordnen und in einem begrenzten Rahmen anzuwenden
- Mittels Process Mining Prozessschwachstellen zu identifizieren und Verbesserungspotenziale evidenzbasiert aufzuzeigen
- Vorhersagemodelle für Prozessverläufe mittels Machine Learning zu entwickeln
- Einfache Prozesse mittels RPA-Software zu automatisieren
- Mit Standardbibliotheken wie PM4Py, SKlearn oder Keras selbst Python-basierte Lösungen im Bereich Process Intelligence zu entwickeln

Literaturliste

Van Der Aalst, W. (2016): Process Mining. *Data science in action*. Springer Berlin Heidelberg.

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018): *Fundamentals of business process management* (Vol. 2). Heidelberg: Springer.

2.52 Programmieren mit Datenbanken

Informationen über das Modul

Name / engl.	Programmieren mit Datenbanken / Programming using Databases
Kürzel	DBP4.WP
Verantwortlicher	Prof. Matthias Kolonko, Ph.D. (ONPU)
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Programmieren mit Datenbanken (4 SWS)
Modulbereich	Anwendungen
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Praktische Übungen
Voraussetzungen	Datenbanken, Programmieren 1+2 Die Vorlesung Datenbankanwendungen wird empfohlen.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970384, 2970882 TI 3976552, 2976698 WI 3975802 IIS 9775147
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Vorlesung befasst sich mit den Möglichkeiten der Anbindung relationaler Datenbanken an die Geschäftslogik, welche in unterschiedlichen Programmiersprachen erstellt sein kann.

Dabei werden folgende grundsätzliche Zugriffsmöglichkeiten näher beleuchtet:

- Direkter Zugriff via eingebettetem SQL
- Zugriff über ein individuelles API
- Zugriff über bestehende Frameworks wie Objekt-relationalem Mapping (ORM) oder Data Transfer Objects (DTO)

Die grundsätzlichen Möglichkeiten und Konzepte werden schwerpunktmäßig anhand der Programmiersprache Java beleuchtet. Ergänzend werden auch weitere aktuelle Programmiersprachen beleuchtet, demonstriert und verglichen. (PHP, Python, C/C++, ...)

Den Teilnehmern soll dabei auch der richtige Aufbau innerhalb der Softwarearchitektur aufgezeigt werden, indem Vor- und Nachteile diskutiert werden. Hierbei werden auch Sicherheitsaspekte berücksichtigt.

Im Rahmen der Vorlesung werden auch Konzepte zum Einsatz von „Polyglot Persistence“ vorgestellt, um Möglichkeiten der Diversifikation der Datenspeicherung aufzuzeigen.

Die besprochenen Inhalte werden durch die Studenten im Rahmen eines begleitenden Praktikums selbst nachvollzogen und geübt.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach Absolvierung des Moduls sind die Teilnehmer in der Lage

- die Möglichkeiten zur Anbindung von Datenbanken zu unterscheiden und zu beschreiben.
- die verschiedenen Möglichkeiten der Datenbankanbindung einzusetzen.
- die Anforderungen an die Datenbankanbindung zu analysieren.
- eine passende Datenbankanbindung zu implementieren.
- Möglichkeiten der Polyglot Persistence zu erkennen.

Literaturliste

2.53 Programmieren mit Python

Informationen über das Modul

Name / engl.	Programmieren mit Python / Programming with Python
Kürzel	PROGPY6.WP
Verantwortlicher	Prof. Dr. Peter Rösch
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Programmieren mit Python (4 SWS) zugehöriges Praktikum (2 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Praktikum, teilweise mit Teamarbeit.
Voraussetzungen	<ul style="list-style-type: none">• Objektorientierte Softwareentwicklung.• Grundlagen der Vektorrechnung und Analysis.
Verwendbarkeit	WPF nur für Bachelorstudiengänge: Wirtschaftsinformatik, International Information Systems, Technische Informatik und Interaktive Medien. Für Informatik (Bachelor) handelt es sich um ein Pflichtfach (Programmieren 3).
Arbeitsaufwand / Zusammensetzung	SWS: 6, CPs: 8, Präsenzzeit: 90 h, Selbststudium: 150 h, Gesamtaufwand: 240 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	TI 3976553, 2976599 WI 3975746 IIS 9775148
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Zusammenfassung

Kunden erwarten performante, leicht zu bedienende Lösungen für immer komplexer werdende Aufgaben, wobei die Beschleunigung der Marktzyklen zu einem enormen Zeit- und Erfolgsdruck für viele Software-Projekte führt.

Moderne Konzepte der Softwaretechnik versprechen Abhilfe, führen aber nur dann zum Erfolg, wenn grundlegende, seit Jahrzehnten bekannte Methoden des Software-Baus zum Einsatz kommen. Das zu lösende Problem muss grundsätzlich zuerst verstanden und systematisch analysiert werden, bevor alternative Lösungsansätze erarbeitet und deren Machbarkeit ggf. durch Prototypen gezeigt werden kann. In der Regel wird ein Ansatz weiter verfolgt, der schließlich in das endgültige Produkt mündet.

Eine wichtige Aufgabe in diesem Prozess ist die Wahl geeigneter Programmiersprachen, wobei in den einzelnen Phasen des Projekts verschiedene Sprachen zum Einsatz kommen können. Damit ein Entwickler oder Projektleiter die „richtige“ Sprache für ein Teilproblem wählen kann, sollte er Erfahrungen mit mehreren Sprachen gesammelt haben und einen Überblick über deren Vor- und Nachteile besitzen.

Die Lehrveranstaltung führt Python als Vertreter der objektorientierten Skriptsprachen ein. Die Syntax dieser Sprache ist so einfach und die Erweiterungs-Bibliotheken sind so mächtig, dass sich Entwickler bei der Umsetzung fortgeschrittener Konzepte auf die Aufgabenstellung konzentrieren können, ohne von Inkonsistenzen oder verzwickten Sprach-Konstrukten abgelenkt zu werden.

Im Rahmen der Übungen werden Aufgabenstellungen aus den Bereichen Mathematik und Simulation behandelt, um zentrale Techniken des Software-Baus zu erarbeiten und praktisch anzuwenden.

Effiziente Software-Entwicklung mit Python

- Python – Einführung
- Interaktive Software-Entwicklung mit Jupyter Notebooks
- Automatisierung von Tests
- Systematische Optimierung
- Grafische Benutzerschnittstellen

Fortgeschrittene Programmier Techniken mit Python

- Nebenläufigkeit
- Entwurfsmuster
- Integration heterogener Komponenten
- Wissenschaftliche Anwendungen
- Verteilte Anwendungen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Die syntaktischen Konstrukte der Programmiersprache Python zu beschreiben.
- Vorgegebenen Quellcode bezüglich Effizienz und Qualität zu klassifizieren.
- Anwendungen bezüglich des Verbrauchs von Rechenzeit und Speicher zu optimieren.
- Die Implementierungen von Algorithmen mittlerer Komplexität in verschiedenen Programmiersprachen zu vergleichen.
- Aufgabenstellungen durch die geschickte Kombination existierender Komponenten zu lösen.
- Probleme mittlerer Komplexität in Teilprobleme zu zerlegen.
- Software-Komponenten für die Lösung von Problemen mittlerer Komplexität selbst zu entwickeln, zu testen und zu dokumentieren.

Literaturliste

Johannes Ernesti, Peter Kaiser: Python3 – Das umfassende Handbuch, 5. Auflage, Rheinwerk Computing (2017)

Bernd Klein: Einführung in Python 3, Hanser (2014)

Mark Pilgrim: Python 3 – Intensivkurs, Springer (2010)

Dusty Phillips: Python 3 Object-Oriented Programming, 3. Auflage, Packt Publishing (2018)

Eric Freeman, Elisabeth Freeman: Entwurfsmuster von Kopf bis Fuß, O'Reilly (2015)

Mark Summerfield: Rapid GUI Programming with Python and Qt - The definitive Guide to PyQt Programming, Prentice Hall (2015)

Python-Homepage: <https://www.python.org/>

2.54 Programmierung von Web-Anwendungen

Informationen über das Modul

Name / engl.	Programmierung von Web-Anwendungen / Programming of Web Applications
Kürzel	PWA4.WP
Verantwortlicher	Prof. Dr. Anja Metzner
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Programmierung von Web Anwendungen (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Projektarbeit, Präsentationen (Hinweis: Bei geringer Teilnehmerzahl kann die Vorlesung auch im Format Directed-Reading mit verringerter Präsenzzeit durchgeführt werden.)
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Projektarbeit, 75%• Studienarbeit, 5-15 Seiten, 25%
Prüfungsnummer	IN 3970381, 2970879 TI 3976605, 2976690 WI 3975799 IIS 9775149
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Webbasierte Systeme entwickelten sich zu einem wichtigen Wirtschaftsfaktor. Diese Vorlesung gibt Übersicht in die Programmierung von Webanwendungen mit aktuellen Frameworks.

Grundlegende Themen:

- Architekturen webbasierter Systeme (u.a. MPA, SPA, progressive Web Apps)
- Auszeichnungs- und Skriptsprachen des Web (z.B. HTML, CSS; JavaScript)
- Übersicht aktueller Frameworks und Bibliotheken (Frontend, Backend)
- Programmierung von Webanwendungen (mit einschlägigen Entwicklungstools, IDEs, Build-Tools, Validierung, Debugging und Test)

Exemplarische Auswahl an Themen für eine Spezialisierung der Studierenden:

- Datenbankverbindungsmöglichkeiten
- Fortgeschrittene Webtechnologien (z.B. AJAX: asynchrone Datenübertragung, REST, Websockets, GraphQL, Cross-Plattform-Entwicklung mobiler Apps)
- Webdesign und UX/UI (u.a. Responsive Design, CSS-Frameworks, Prinzipien, Barrierefreiheit)
- Sicherheitsthemen (z.B. HTTPS, Authentifizierung, Sicherheitslücken)

Einführung:

Mithilfe von professoralen Kurzvorträgen über Web-Architektur, Skriptsprachen, aktuellen Frameworks, einer einschlägigen Materialsammlung und Projektbegleitung werden Studierende zur Webprogrammierung und zum weiteren Selbststudium befähigt.

Wahl der Skriptsprachen und Frameworks:

Die Studierenden werden in der Lage sein, die Skriptsprachen des Web, insbesondere HTML, JavaScript, CSS **und mindestens ein Framework** ihrer Wahl, zu verstehen und zu programmieren. **Dabei hängt die Wahl der möglichen Skriptsprachen und Frameworks vom Studiengang der Studierenden ab. Ausgeschlossen sind solche, die bereits im Curriculum oder in anderen zugelassenen Wahlpflichtfächern des jeweiligen Studienganges studierbar sind.**

Projektarbeit und Präsentation:

Mithilfe von Projekt-Experimenten erlernen Studierende die Programmierung eigener Webprojekte. Die dabei verwendeten Techniken werden in studentischen Projektvorträgen allen Teilnehmern vorgestellt, so dass ein breitgefächelter Einblick über viele aktuelle Frameworks und Bibliotheken entsteht. Jeder Vortrag wird schließlich als Studienarbeit dokumentiert und (freiwillig) allen Teilnehmern zur Verfügung gestellt.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Studierende sind anschließend in der Lage die Programmierung von Web-Anwendungen zu lesen, zu verstehen und selbst Webprojekte zu erstellen. Sie erhalten somit die Grundlagen als Fullstack-Programmierer tätig werden zu können.

1. Grundlagen der Webprogrammierung verstehen

- Die Studierenden können die grundlegenden Konzepte und Technologien des Webs erklären (HTML, CSS, JavaScript, HTTP/HTTPS).
- Die Studierenden kennen die Architektur des Internets und können die Funktionsweise von Webservern und -clients beschreiben.
- Die Studierenden können die Bedeutung von Webstandards (W3C) erläutern und wissen, warum diese wichtig sind.
- Die Studierenden sind mit aktuellen Best Practices für die Webentwicklung vertraut, beispielsweise Responsivität und Barrierefreiheit.

2. Webprogrammierung anwenden

- Die Studierenden können strukturierte und semantisch korrekte HTML-Dokumente erstellen.
- Die Studierenden können CSS verwenden, um das Layout und das Design von Webseiten zu gestalten und anzupassen.
- Die Studierenden können grundlegende JavaScript-Programme schreiben, die Interaktivität auf Webseiten ermöglichen.
- Die Studierenden verstehen das DOM (Abk. Document Object Model) und können JavaScript verwenden, um es zu manipulieren.
- Die Studierenden sind in der Lage einen selbst gewählten Technologie-Stack für die Programmierung zu installieren und einzusetzen.
- Die Studierenden sind in der Lage, mindestens ein aktuelles Framework anzuwenden und bei Bedarf JavaScript-Bibliotheken einzubinden.
- Die Studierenden lernen bei der Programmierung mit unterschiedlichen Browser umzugehen.
- Die Studierenden können Fehler in Webanwendungen systematisch identifizieren und beheben.
- Die Studierenden lernen Themen der Spezialisierung, wie beispielsweise Datenbankbindung, JSON oder AJAX kennen.

3. Teamarbeit und Projektmanagement

- Die Studierenden können in Teams zusammenarbeiten, um komplexe Webprojekte zu planen und umzusetzen.
- Die Studierenden kennen Methoden des agilen Projektmanagements und können diese in der Webentwicklung anwenden (z.B. Scrum, Kanban).

4. Dokumentation und Präsentation

- Die Studierenden können ihre Projekte und den Entwicklungsprozess schriftlich dokumentieren und in Präsentationen vorstellen.
- Die Studierenden sind in der Lage, technische Probleme und Lösungen klar und verständlich zu kommunizieren.

5. Technologietransfer

- Die Studierenden sind in der Lage, ihre Kenntnisse auf neue Technologien und Frameworks zu übertragen, indem sie grundlegende Prinzipien der Webentwicklung anwenden.

Literaturliste

Philip Ackermann: Webentwicklung: Das Handbuch für Fullstack-Entwickler in neuer Auflage, Rheinwerk Computing, 2023.

David Flanagan, Jens Olaf Koch , et al.: JavaScript - Das Handbuch für die Praxis, O'Reilly, 2021.

Cybellium Ltd, Kris Hermans: Mastering Back-End Development, Independently published, 2023.

Kiet Huynh: Front-End Web Development: Techniques and Trends, Independently published, 2023.

Sebastian Springer: Node.js: The Comprehensive Guide, Rheinwerk Computing, 2022.

2.55 Project Jupyter

Informationen über das Modul

Name / engl.	Project Jupyter / Project Jupyter
Kürzel	PRJU4.WP
Verantwortlicher	Prof. Dr. Nik Klever
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Lehrveranstaltungen in den Modulen (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Die Veranstaltung ist in vier Teile gegliedert: <ul style="list-style-type: none">• Teil 1 - Einführung in die Anwendungen von Project Jupyter und Übungen hierzu (1. Block 2 Tage)• Teil 2 - Brainstorming und Ideenfindung von Studienarbeitsthemen aus z.B. folgenden Bereichen (2. Block 2 Tage)• Teil 3 - Umsetzung der Studienarbeitsthemen (Online ca. 11 Wochen)• Teil 4 - Präsentation der Studienarbeiten (3. Block 1 Tag)
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• Übungen, 20%• Beschreibung Brainstorming und Ideenfindung, 10%• Studienarbeit, 10-40 Seiten, 70%
--------------	--

Prüfungsnummer	IN 3970374, 2970872 TI 2976683 WI 3975792 IIS 9775150
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Project Jupyter umfasst inzwischen einige Anwendungen, hervorgegangen ist es aus dem IPython Notebook, welches anschliessend in das inzwischen, insbesondere für Data Science und AI/KI Anwendungen weit verbreitete Jupyter Notebook überging.

Jupyter Notebook

Jupyter Notebook wird seit einigen Jahren nicht nur für Informatiker sondern auch für Naturwissenschaftler, Wirtschaftswissenschaftler und auch Ingenieure immer beliebter. Warum ist das so ? Dies liegt u.a. daran, dass Jupyter Notebook eine einfache Kombination unterschiedlichster Materialien wie normalen Text, Bilder, Grafiken mit HTML, LaTeX, SVG-Grafiken und insbesondere dies mit Programmiercode unterschiedlichster Programmiersprachen wie Python, Java, JavaScript, C++, R, Scala, u.a. vermischen kann. Dabei liegt insbesondere der Vorteil auch darin, dass die Benutzerschnittstelle eines Jupyter Notebook Servers zur Erstellung eines Jupyter Notebooks einzig und allein ein Browser ist.

JupyterLab

Die Weiterentwicklung des Jupyter Notebook ist das JupyterLab, welches eine erweiterte webbasierte interaktive Entwicklungsumgebung für Jupyter Notebooks, Programmcode oder Daten ist. JupyterLab ist flexibler als Jupyter Notebook, da die Benutzeroberfläche konfigurierbar und selbst angeordnet werden kann. Damit kann eine Vielzahl von Abläufen in den Bereichen Data Science, Scientific Computing und maschinelles Lernen unterstützt werden. JupyterLab ist zudem über Plugins und Komponenten erweiterbar und modular.

JupyterHub

Jupyter Notebook und JupyterLab sind Single-User Webserver, die auf jedem Rechner einfach zu installieren und lauffähig sind. Die Erweiterung dieser Single-User Webserver für Firmen, Organisationen, Hochschulen, Arbeitsteams, etc. zu einem Multi-User Webserver ist durch den JupyterHub Server erfolgt. Auch für den JupyterHub Server gibt es entsprechende Erweiterungen, wie z.B. nbgrader, ein auf Jupyter Notebook und JupyterHub basierendes automatisiertes Verteilungs- und Codeprüfungs Framework.

Voilà

Als jüngstes Mitglied von Project Jupyter ist Voilà hinzugekommen, eine Anwendung, die Jupyter Notebooks in eine eigenständige Webanwendung in der Art umwandelt, dass nur der Programmcode aus dem Jupyter Notebook für die Benutzer sichtbar und anwendbar ist, der vom Jupyter Notebook Besitzer dafür freigegeben worden ist. Diese Freigabe wird über ein sicheres und anpassbares interaktives Dashboard gesteuert werden.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden können die einzelnen Anwendungen aus dem Project Jupyter einordnen, verstehen und konfigurieren sowie anwenden. Desweiteren sollten sie einzelne Anwendungen in Form von Plugins oder Patches verbessern oder sogar weiterentwickeln können.

Literaturliste

Weitere Informationen unter

<https://klever.hs-augsburg.de/nb/OWL/>

2.56 Projekt - Forschung und Transfer

Informationen über das Modul

Name / engl.	Projekt - Forschung und Transfer / Project - Research and Transfer
Kürzel	FUT.WP
Verantwortlicher	Prof. Dr. Alexander von Bodisco
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Winter- und Sommersemester
Lehrveranstaltungen	Forschungs- und Transferprojekt
Lehrsprache	Das Modul wird in deutscher und englischer Sprache unterrichtet.
Lehr-/Lernmethoden	<p>Studierende erarbeiten in Kleingruppen IT-Lösungen zu einem aktuellen Forschungsthema. Ziel ist es anwendungsorientierte Forschung, sowie den Transfer und die damit verbundenen Problemstellungen realitätsnah kennenzulernen. Die durchgeführten Projekte haben einen klaren Praxisbezug und finden typischerweise im Rahmen von Förder-/Drittmittelprojekten oder in Kooperation mit Unternehmen statt.</p> <p>Die Projektthemen werden von Prüfungsberechtigten der Fakultät für Informatik vergeben und umfassen einen praktischen Teil (Software/Hardware), eine Dokumentation (Studienarbeit) und eine Präsentation. Der praktische Teil (Software und ggf. Hardware) ist im Rahmen der Studienarbeit zu beschreiben. Die Präsentation findet in der Regel im Rahmen eines Projekttag oder eines Seminars statt. Die Abstimmung mit dem Projektsteller erfolgt in regelmäßigen persönlichen Treffen und über elektronische Kanäle. Die Bearbeitung ist nicht notwendigerweise an die Vorlesungszeit gebunden.</p>
Voraussetzungen	Solide Kenntnisse aus den wichtigsten Themenbereichen der Informatik, wie z.B. Algorithmen und Datenstrukturen, Programmieren, Datenbanken, Datenkommunikation, Software Engineering und ggf. Betriebssysteme. Die erworbenen Kenntnisse sollten bereits in einem Teamprojekt praktisch angewendet worden sein.

Verwendbarkeit	Wahlpflichtfach für die Bachelorstudiengänge Informatik, International Information Systems, Technische Informatik, Wirtschaftsinformatik
Arbeitsaufwand / Zusammensetzung	SWS: 8, CPs: 10, Präsenzzeit: 120 h, Selbststudium: 180 h, Gesamtaufwand: 300 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none"> • Projektarbeit, 10-20 Seiten, 80% • Präsentation, 10-20 Minuten, 20%
Prüfungsnummer	IN 3970404, 2970902 TI 3976627, 2976728 WI 3975822 IIS 9775182
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Das Forschungs- und Transferprojekt bietet den Studierenden die Möglichkeit, theoretisches Wissen in der Praxis anzuwenden und gleichzeitig innovative Lösungen für reale Herausforderungen zu entwickeln. Das Projekt legt einen Schwerpunkt auf Forschung, Teamarbeit und den Transfer von Ergebnissen in die Praxis. Zu den Aufgaben der Studierenden zählen das Projektmanagement, die Softwareentwicklung, die selbstständige Einarbeitung in Forschungsthemen, das Aufbereiten von Forschungsergebnissen und deren Präsentation im Hinblick auf die praktische Anwendung.

In moodle finden Sie die aktuellen Themen, die gerade angeboten werden:
<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Softwareaufgaben im Team hinsichtlich Zeit, Aufwände und Ressourcen zu planen und durchzuführen.
- Softwareentwicklungsmethoden praktisch anzuwenden.
- Neue Softwaretechniken zu erlernen und geeignete Methoden anzuwenden.
- Forschungsthemen selbstständig zu erarbeiten.
- Projektergebnisse verständlich zu dokumentieren und ansprechend zu präsentieren.

Literaturliste

Wird individuell für jedes Projekt festgelegt und orientiert sich an der aktuellen wissenschaftlichen Forschung im gewählten Bereich.

2.57 RFID und NFC Technik

Informationen über das Modul

Name / engl.	RFID und NFC Technik / RFID and NFC technology
Kürzel	RFID3.WP
Verantwortlicher	Prof. Dr. Volodymyr Brovko
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Das Modul wird regelmäßig sowohl im Wintersemester als auch im Sommersemester angeboten.
Lehrveranstaltungen	RFID und NFC Technik (3 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungen, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 3, CPs: 5, Präsenzzeit: 45 h, Selbststudium: 105 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, Hilfsmittel: 1 DIN-A4-Seite handgeschrieben
Prüfungsnummer	IN 2970806 TI 2976589 WI ...
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

- Grundlagen der RFID Technik. Codierung und Modulation in RFID. Antikollision in RFID.
- Speicherkarten Architektur. Smardcards Architektur. Java Cards.
- Autonome RFID und NFC Systeme: technische Grundlagen. Softwareentwurf.
- NDEF on Speicherkarte und MIFARE.
- Architektur mobiler NFC Geräte.
- NFC on ANDROID System

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Technische Grundlagen der RFID-Technik zu wissen.
- Speicher- und Smartkartenarchitektur zu verstehen.
- Architektur eines autonomen RFID und NFC-Systems zu kennen.
- Ein einfaches NFC Lese-/Schreibegerät auf der Basis eines Mikrocontrollers programmieren zu können.

Literaturliste

Josef Langer, Michael Roland: Anwendungen und Technik Von Near Field Communication (NFC), Springer-Verlag, 2010 - 265 Seiten

Klaus Finkenzeller: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC, Carl Hansen Verlag München, 2012

Wolfgang Rankl, Wolfgang Effing: Handbuch der Chipkarten, Carl Hansen Verlag München, 2008

2.58 Schwaben Innovation Masterclass

Informationen über das Modul

Name / engl.	Schwaben Innovation Masterclass / Swabia Innovation Masterclass
Kürzel	SIM8.WP
Verantwortlicher	Prof. Dr. Christoph Buck (THA) Prof. Dr. Bayer/ Prof. Dr. Daniel Schallmo (HNU) Prof Dr. Erik Lehmann (UniA) Prof. Dr. Bernd Lüdemann-Ravit (HKE)
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	Die Dauer des Moduls beträgt zwei Semester. Die Schwaben Innovation Masterclass wird regelmäßig mit Start im Wintersemester angeboten.
Lehrveranstaltungen	Schwaben Innovation Masterclass (4 SWS pro Semester, gesamt 8 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Workshop-Einheiten, Best-Practices, Team-/Gruppenarbeit, Präsentationen, maximale Teilnehmerzahl 6
Voraussetzungen	Beide Module müssen belegt werden, es ist nicht möglich nur eine Masterclass zu besuchen. Insgesamt können 10 ECTS erworben werden. Reise- und Übernachtungskosten für die Anreise nach Neu-Ulm, Kempten sowie Bergamo werden über das Verbundprojekt übernommen.
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	Pro Semester: SWS: 4, CPs: 5, insgesamt 8 SWS/ 10 CPs Präsenzzeit: 30 h, Selbststudium (Vor-/ Nachbereitung): 120 h, Gesamtaufwand: 150h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none"> • Teilprüfung 1: Präsentation, 5-10 Minuten (70%) und Schriftliche Ausarbeitung, 3-8 Seiten (30%), 25% • Teilprüfung 2: Präsentation, 5-10 Minuten (70%) und Schriftliche Ausarbeitung, 3-8 Seiten (30%), 25% • Teilprüfung 3: Präsentation, 5-25 Minuten (70%) und Abschlussbericht, 10-14 Seiten (30%) , 25% • Teilprüfung 4: Präsentation, 15-25 Minuten (70%) und Pitchfolien, 7-10 Folien (30%) , 25%
Prüfungsnummer	
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Dieser Kurs besteht aus vier Kapiteln: future (&) trends, social impact, business ideation und international entrepreneurship. Kapitel 1 wird an der Technischen Hochschule Augsburg angeboten, Kapitel 2, 3 und 4 als externe Lehrveranstaltungen an den Hochschulen Kempten und Neu-Ulm und der Universität Augsburg inkl. Auslandsaufenthalt in Bergamo (Italien).

Kapitel 1: Future (&) Trends (THA)

Kapitel 1 ist in zwei Phasen unterteilt: die Trend- und die Szenarioanalyse. Im Rahmen der Trendanalyse setzen sich die Studierenden mit aktuellen Entwicklungen im Bereich digitaler Technologien auseinander. Sie erfassen den Status Quo und identifizieren aufkommende Trends. Ein interdisziplinärer Ansatz ermöglicht es den Studierenden, die Themen aus verschiedenen Blickwinkeln zu betrachten und gesellschaftliche, wirtschaftliche, soziale, politische, rechtliche und ökologische Rahmenbedingungen zu berücksichtigen. In der zweiten Phase, der Szenarioanalyse, bauen die Studierenden auf den Ergebnissen der Trendanalyse auf und entwickeln eigene Ideen für innovative Produkte oder Dienstleistungen. Sie setzen sich mit den möglichen Auswirkungen der identifizierten Trends auseinander und erarbeiten Szenarien für zukünftige Entwicklungen.

Die Lehrmethoden umfassen Vorlesungen und Workshops, die von externen und internen Dozenten aus Wissenschaft und Industrie geleitet werden. Zusätzlich erhalten die Studierenden regelmäßiges Coaching und Feedback zu ihrer Arbeit. Ergänzend haben die Studierenden im Rahmen einer gemeinsamen Zwischen- und Abschlusspräsentation die Möglichkeit, ihre Ergebnisse und Ideen im Plenum zu diskutieren und weiterzuentwickeln.

Kapitel 2: Social Impact (HKE)

In Kapitel 2 werden die Studierenden in die Vielfalt sozialer Herausforderungen auf globaler, regionaler und lokaler Ebene eingeführt. Sie erlernen die Konzepte des Impact Thinkings, einschließlich Skalierbarkeit, Nachhaltigkeit und Auswirkungsmessung in sozialen Innovationen. Durch die Analyse von Fallstudien erfolgreicher Projekte gewinnen sie praxisnahe Einblicke. Im Bereich "Technologie für sozialen Impact" werden digitale Technologien wie Blockchain, Künstliche Intelligenz und das Internet der Dinge diskutiert, sowie ihre Anwendungsmöglichkeiten für die Bewältigung sozialer Herausforderungen erörtert. Die Projektentwicklung und -Managementphase ermöglicht den Studierenden die praktische Umsetzung ihres Wissens durch die Entwicklung technologiebasierter Lösungen in interdisziplinären Teams, begleitet von Mentor:innen und Expert:innen. Die abschließende Challenge beinhaltet die Präsentation der Ergebnisse vor einer interdisziplinären Jury sowie die Reflexion über den Prozess.

Kapitel 3: Business Ideation (HNU)

Die Masterclass bietet den Studierenden eine praxisnahe Auseinandersetzung mit unternehmerischen Herausforderungen, insbesondere im Kontext von Innovation und Startup-Initiativen. Die Studierenden entwickeln ein fundiertes Verständnis für die Bedeutung von Innovationen. Sie erwerben etablierte Methoden und praktische Werkzeuge aus

verschiedenen Bereichen, darunter Kreativitätstechniken, Design Thinking sowie Lean Startup, um kunden- und bedarfsorientierte Innovationen zu generieren. Gleichzeitig erhalten sie die Möglichkeit, eigene Innovationsprojekte in Zusammenarbeit mit regionalen Unternehmen zu entwickeln, um einen realitätsnahen Einblick in unternehmerische Prozesse zu erhalten.

Kapitel 4: International Entrepreneurship (UA)

Hauptziel des Moduls ist es, dass die Studierenden anknüpfend an Kapitel 3 in einem mehrtägigen Kurs die internationalen Aspekte von Entrepreneurship und Innovation vertiefen. Sie erlernen dabei internationale Kompetenzen im Rahmen einer Summer School. Diese startet an der Universität Augsburg und endet mit einer Exkursion an der Universität Bergamo in Italien. Im Vordergrund stehen dabei Lernerfahrungen mit anderen Kulturkreisen. Ein besonderer Schwerpunkt des Kapitels stellen die Gastvorträge internationaler renommierter Gastdozenten von der Indiana University Bloomington, USA dar. Diese sollen den Studierenden in englischer Sprache Wissen, Erfahrungen und Ansichten mit wissenschaftlichem Blickwinkel vermitteln. Ergänzend werden praktische Elemente vermittelt, indem Führungskräfte von Unternehmen der Region Augsburg Einblicke geben und ihr Wissen teilen.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme an den **Kapiteln 1 und 2:**

- sind die Studierenden in der Lage die Herausforderungen der Zusammenarbeit in interdisziplinären Projektteams zu verstehen.
- Sie sind außerdem befähigt, Trend- und Zukunftsforschungsmethoden in einem Projektteam anzuwenden und gemeinsam einen Trendbericht zu aktuellen Entwicklungen im Bereich digitaler Technologien zu verfassen.
- Die Studierenden können eine Status-quo-Analyse durchführen, Trends und zukünftige Entwicklungen identifizieren sowie Ideen für zukünftige Produkte oder Dienstleistungen bilden.
- Das Modul zielt darauf ab, den Studierenden praktische Fähigkeiten in der Anwendung von Forschungsmethoden, im wissenschaftlichen Schreiben, in der Diskussion und Präsentation von Themen sowie in der interdisziplinären Zusammenarbeit zu vermitteln.
- Die Studierenden werden dazu befähigt, soziale Herausforderungen auf verschiedenen Ebenen zu verstehen und Impact Thinking zu entwickeln.
- Sie lernen komplexe Probleme aus verschiedenen Blickwinkeln zu betrachten und sich mit der Skalierbarkeit, Nachhaltigkeit und Messung von Auswirkungen in sozialen Innovationen auseinanderzusetzen.
- Durch die Präsentation und Diskussion von Fallstudien erfolgreicher Projekte erhalten sie Einblicke in bewährte Praktiken. Des Weiteren werden sie mit verschiedenen digitalen Technologien vertraut gemacht, um soziale Probleme anzugehen, und haben die Möglichkeit, diese in Workshops und Gastvorträgen zu vertiefen.
- Durch Projektentwicklung und -management in interdisziplinären Teams entwickeln sie technologiebasierte Lösungen für spezifische soziale Probleme und präsentieren ihre Ergebnisse vor einer Jury.
- Durch Reflexion und Dokumentation wird ihr Lernprozess unterstützt und vertieft.

Kapitel 3: Business Ideation

Fachbezogene Kompetenzen:

- Die Studierenden werden befähigt, Konzepte wie Lean Startup und Design Thinking anzuwenden, um Innovationsprozesse zu steuern und erfolgreich umzusetzen.
- Sie erlangen ein tiefgreifendes Verständnis für die Businessplanung und lernen, Tools effektiv einzusetzen, um agile und erfolgreiche Gründungsprozesse zu gestalten.

- Die Identifikation von Nachfragepotentialen, Zielgruppen und Wettbewerbsvorteilen stehen dabei ebenfalls im Fokus.

Methodische Kompetenzen:

- Die Studierenden erlernen Methoden zur Ideenfindung und Entwicklung von Geschäftsmodellen sowie zur Marktanalyse, um fundierte Entscheidungen im Unternehmenskontext treffen zu können.
- Ein besonderer Schwerpunkt liegt auf der Anwendung von Design Thinking, einer kreativen Problemlösungsmethode, die darauf abzielt, nutzerzentrierte Lösungen zu entwickeln.
- Zusätzlich werden sie in die Methodik des Design Sprints Lean Startups eingeführt, um schnell und effizient Lösungen zu konzipieren, zu prototypisieren und zu testen.

Soziale Kompetenzen

- Die Studierenden entwickeln Fähigkeiten zum effektiven Treffen von Entscheidungen im Team und lernen, innovative Lösungen zu verstehen sowie vor Unternehmensvertretern überzeugend zu präsentieren.
- Der Austausch mit Kommilitonen und die Zusammenarbeit in interdisziplinären Teams fördern das Verständnis für verschiedene Perspektiven und die Entwicklung von Lösungsansätzen.

Persönliche Kompetenzen

- Die Studierenden werden ermutigt, die Konsequenzen ihrer Entscheidungen zu reflektieren und ihre persönlichen Fähigkeiten im Bereich der unternehmerischen Initiative und Risikobewertung weiterzuentwickeln. Durch die praktische Arbeit an Innovationsprojekten stärken sie ihre Problemlösungsfähigkeiten und gewinnen Selbstvertrauen in ihre Fähigkeiten.

Kapitel 4: International Entrepreneurship

Durch die Teilnahme an dieser Lehrveranstaltung sollen die Studierenden ein vertieftes Verständnis von internationalem Entrepreneurship zur Entwicklung von Innovationen erlangen. Nach erfolgreicher Teilnahme an diesem Modul kennen die Studierenden die Grundlagen des Unternehmertums und sind in der Lage:

Fachbezogene Kompetenzen

- internationales Unternehmertum zu verstehen,
- ihre eigenen Gründungsideen und Geschäftsmodelle zu internationalisieren,
- die Perspektive zentraler sozialer, ökologischer und ökonomischer Veränderungen in Gesellschaft, Umwelt und Wirtschaft international und interkulturell einzunehmen

- innovative Ideen auf den internationalen Markt zu projizieren

Methodische Kompetenzen

- bearbeiten komplexe Gründungsvorhaben zielorientiert,
- führen systematische Bedarfs- und Handlungsanalysen im gesellschaftlichen Kontext aus unterschiedlichen Perspektiven durch.

Interdisziplinäre Kompetenzen

- wenden multiperspektivisches Denken an,
- erkennen und fördern Chancen für soziale, ökonomische und ökologische Verbesserungen aus unterschiedlichen Perspektiven,
- setzen innovative Lösungen im Kontext des internationalen Unternehmertums um.

Schlüsselkompetenzen

- reflektieren Strategien zur Unternehmensgründung,
- entwickeln und begründen selbstständig strategische Überlegungen,
- denken und arbeiten interdisziplinär,
- lösungsorientierte und interkulturelle Kommunikation.

Literaturliste

Kapitel 1: Wird zu Beginn des Semesters bekannt gegeben

Kapitel 2:

- Chang, Ann Mei: Lean Impact, How to Innovate for Radically Greater Social Good, San Francisco, 2018.
- Impact Measurement – Wirkung- und Wirkungsmessung Sozialer Innovationen.
- Kursbuch Wirkung, Praxishandbuch für alle, die Gutes noch besser tun wollen. Social Reporting Standard, Leitfaden zur wirkungsorientierten Berichterstattung
- u.v.m

Kapitel 3 und 4:

Audretsch, David: Everything in Its Place: Entrepreneurship and the Strategic Management of Cities, Regions, and States. New York: Oxford University Press (2015).

Audretsch, David; Lehmann, Erik: The seven secrets of Germany. Economic Resilience in an Era of Global Turbulence. New York: Oxford University Press (2016).

Schallmo, D.: Design Thinking erfolgreich anwenden, Springer Verlag, Wiesbaden (2017).

Pijl, P. v. d., Lokitz, J., Solomon, L., Pluijm, E. v. d., Lieshout, M. v., Schallmo, D.: Design a better business: Neue Werkzeuge, Fähigkeiten und Mindsets für Strategie und Innovation, Vahlen Verlag, München (2018).

Schallmo, D.: Jetzt Design Thinking anwenden, Springer Verlag, Wiesbaden (2018).

Schallmo, D.: Jetzt digital transformieren. So gelingt die erfolgreiche Digitale Transformation Ihres Geschäftsmodells, Springer Verlag, Wiesbaden (2016)

Brown, T.: Change by Design. Harper Business (2009).

Curedale, R.: Design Thinking. Design Community College (2013).

d.school: Bootcamp Bootleg. Hasso Plattner Institute, Stanford (2010).

Liedtka, J. & Ogilvie, T.: Designing for Growth. Columbia Business School (2011).

Plattner, H.; Meinel, Ch. & Weinberg, U.: Design Thinking. Innovation lernen, Ideenwelten öffnen. München (2009).

Stickdorn, M. & Schneider, J.: This is service design thinking. BIS publishers (2014).

2.59 Service Learning Projekt

Informationen über das Modul

Name / engl.	Service Learning Projekt / Service Learning Project
Kürzel	SLP.WP
Verantwortlicher	Prof. Dr. Alexander von Bodisco
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Winter- und Sommersemester
Lehrveranstaltungen	Hochschul Innovationsprojekt
Lehrsprache	Das Modul wird in deutscher und englischer Sprache unterrichtet.
Lehr-/Lernmethoden	Studierende erarbeiten in Kleingruppen individuelle IT-Lösungen aus dem Bereich Service Learning für einen realen zivilgesellschaftlichen Partner. Ziel ist neben den klassischen Projektkompetenzen die Kommunikation mit dem Projektpartner zu schulen und ein Projekt auf eine konkrete Dienst- bzw. Serviceleistung auszurichten. Die Projektthemen werden von Prüfungsberechtigten der Fakultät für Informatik vergeben und umfassen einen praktischen Teil (Software/Hardware), eine Dokumentation(Studienarbeit) und eine Präsentation. Der praktische Teil (Software und ggf. Hardware) ist im Rahmen der Studienarbeit zu beschreiben. Die Präsentation findet in der Regel im Rahmen eines Projekttag, eines Seminars oder einer Demonstration beim Projektpartner statt. Die Abstimmung mit dem Themensteller/Projektpartner erfolgt in regelmäßigen persönlichen Treffen und über elektronische Kanäle. Die Bearbeitung ist nicht notwendigerweise an die Vorlesungszeit gebunden.
Voraussetzungen	Solide Kenntnisse aus den wichtigsten Themenbereichen der Informatik, wie z.B. Algorithmen und Datenstrukturen, Programmieren, Datenbanken, Datenkommunikation, Software Engineering und Betriebssysteme. Die erworbenen Kenntnisse sollten bereits in einem Teamprojekt praktisch angewendet worden sein.
Verwendbarkeit	Wahlpflichtfach für die Bachelorstudiengänge Informatik, International Information Systems, Technische Informatik, Wirtschaftsinformatik

Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h
-------------------------------------	---

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none"> • Projektarbeit, 10-20 Seiten, 80% • Präsentation, 10-20 Minuten, 20%
Prüfungsnummer	IN 3970405, 2970903 TI 3976628, 2976729 WI 3975823 IIS 9775183
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Studierenden führen in Gruppen weitestgehend eigenständig IT-Kleinprojekte aus dem Bereich Service Learning durch. Zu den Aufgaben der Studierenden zählen Projektmanagement, Softwareentwicklung, die selbstständige Einarbeitung in interdisziplinäre Themen und die Projektausrichtung im Hinblick auf die individuellen Anforderungen der jeweiligen Zielgruppe.

In moodle finden Sie die aktuellen Themen, die gerade angeboten werden:

<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Projekte auf die Anforderungen von realen zivilgesellschaftlichen Partnern auszurichten.
- Projektaufgaben im Team hinsichtlich Zeit, Aufwände und Ressourcen zu planen und durchzuführen.
- Softwareentwicklungsmethoden praktisch anzuwenden.
- Interdisziplinäre Themen im Selbststudium aufzubereiten.
- Fragestellungen und Lösungen im Dialog mit Projektpartnern zu erarbeiten.

Literaturliste

Projektspezifische Literatur wird vom Betreuer vor Beginn des Projektes bekanntgegeben.

2.60 Softwaretechnik für datenintensive Systeme

Informationen über das Modul

Name / engl.	Softwaretechnik für datenintensive Systeme / Software engineering for data-intensive systems
Kürzel	SDS4.WP
Verantwortlicher	Dr. Jan-Philipp Steghöfer
Fakultät	Fakultät für Geistes- und Naturwissenschaften
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester Die Anmeldung erfolgt per E-Mail direkt bei Herrn Steghöfer: jan-philipp.steghoefer@tha.de
Lehrveranstaltungen	Softwaretechnik für datenintensive Systeme (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	
Voraussetzungen	
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	<ul style="list-style-type: none">• Erfahrungsbericht (abzugeben in der Gruppe)• Einzelgespräche• Peer Assessment• Quellcode-Analyse
Prüfungsnummer	IN -, - TI - WI - IIS -
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Daten und Modelle, mit denen Data Scientists arbeiten, werden immer im Kontext einer Anwendung eingesetzt. Insbesondere ist es notwendig, Daten und auf ihnen basierende Machine Learning Modelle in eine Applikation einzubetten. Diese Applikation muss den Anforderungen der Nutzer genügen, eine ansprechende und bedienbare Schnittstelle aufweisen. Außerdem ist es notwendig, diese auf einfache Weise auszurollen, zu aktualisieren und zu betreiben.

Anhand einer konkreten Applikation, die Studierende in Zusammenarbeit mit externen Akteuren entwickeln, werden in diesem Kurs die Grundlagen der Applikationsentwicklung, insbesondere Grundlagen der Software-Architektur sowie Scrum als agiler Softwareentwicklungsprozess, geübt. Außerdem liegt ein Fokus auf dem Betrieb von Software in Containern mit Hilfe von Docker und Kubernetes. Dabei arbeiten Studierende der Informatik und aus der Data Science zusammen in einem Team, unterstützen sich gegenseitig mit ihren jeweiligen Fachkenntnissen und interagieren gemeinsam mit den externen Partnern. Die Zusammenarbeit mit externen Partnern erfordert außerdem das Erheben von Anforderungen und regelmäßige Feedbackschleifen.

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreichem Abschluss des Kurses werden die Studierenden in der Lage sein:

- den Kontext der Arbeit mit Daten und Modellen und die Bedeutung ihrer Einbettung in eine Anwendung zu beschreiben,
- agile Softwareentwicklungsprozesse, insbesondere Scrum zu verstehen, und wie diese bei der Entwicklung von Data Science-Anwendungen eingesetzt werden können, zu demonstrieren,
- ihre Programmierkenntnisse zur Entwicklung einer Data-Science-Anwendung, die Daten und maschinelle Lernmodelle einbezieht, anzuwenden,
- effektiv mit externen Partnern zusammenzuarbeiten, um Anforderungen zu erfassen, regelmäßiges Feedback einzuholen und die erfolgreiche Bereitstellung der Anwendung sicherzustellen,
- Verständnis der Prinzipien der Softwarearchitektur und ihrer Rolle bei der Entwicklung skalierbarer und wartbarer Data-Science-Anwendungen zu demonstrieren, und
- ihre Fähigkeit, Anwendungen in Docker-Containern zu verpacken und mit Hilfe von Kubernetes zu deployen, warten und zu betreiben, demonstrieren.

Literaturliste

Literatur wird in der Veranstaltung bekannt gegeben.

2.61 Software-Projektmanagement

Informationen über das Modul

Name / engl.	Software-Projektmanagement / Software Project Management
Kürzel	SWPJMG.WP, SWPMG4.WP
Verantwortlicher	Dipl.-Wirt.-Inf. (FH) Andrea Obermeyer, MBA
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Software-Projektmanagement (4 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht, Übungsgruppen, Präsentation von Spezialinhalten durch Masterstudierende
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	IN 3970330, 2970802 TI 3976555, 2976576 WI 3975722 IIS 9775154
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Die Lehrveranstaltung verbindet theoretische Inhalte mit praktischen Übungskomponenten, studentischen Lehrelementen und ausführlichen Fallstudien aus dem Software-Projektmanagement. Folgende Schwerpunktbereiche werden behandelt:

- Einführung in Projektmanagement: Aufgaben, Schnittstellen, Projektphasen und Projektorganisation
- Vorgehensmodelle, Software-Lebenszyklen und Entwicklungsmethoden (agile vs. konventionell)
- Projekttypen
- Projektplanung: Machbarkeitsstudien, Requirements Engineering
- Aufwandsabschätzung
- Projektüberwachung/-controlling
- Führung: Unternehmenskultur, Leadership, Teambuilding
- Soft- und Social-Skills für Projektteams und Mitarbeiter
- Risikomanagement
- Fallstudien zu ausgewählten Beispielprojekten

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul Software-Projektmanagement können die Studierenden:

- Verstehen, was modernes Software-Projektmanagement ist
- Verstehen, wie kleine und große, technische und wirtschaftsorientierte Softwareprojekte organisiert und zum Erfolg geführt werden oder scheitern
- Methoden, Techniken und Hilfsmittel für das Projektmanagement auswählen und anwenden
- Team-Dynamik begreifen
- Verstehen, welche Soft- und Social-Skills dazu entwickelt werden sollten

Literaturliste

Literatur wird in der Veranstaltung bekannt gegeben.

2.62 Startitup - Entrepreneurial Thinking and Business Design

Informationen über das Modul

Name / engl.	Startitup - Entrepreneurial Thinking and Business Design / Startitup - Entrepreneurial Thinking and Business Design
Kürzel	START4.WP
Verantwortlicher	Prof. Dr. Christoph Buck
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Sommersemester
Lehrveranstaltungen	Startitup - Entrepreneurial Thinking and Business Design(2 SWS) Praktikum Startitup (2 SWS)
Lehrsprache	Das Modul wird in deutscher und englischer Sprache unterrichtet.
Lehr-/Lernmethoden	Seminaristischer Unterricht und begleitendes Praktikum zur Anwendung und Vertiefung der erworbenen Kenntnisse. Zusätzlich unterstützt und fördert das Praktikum das Selbststudium.
Voraussetzungen	Keine
Verwendbarkeit	Wahlpflichtfach für Bachelorstudiengänge
Arbeitsaufwand / Zusammensetzung	SWS: 4, CPs: 5, Präsenzzeit: 60 h, Selbststudium: 90 h, Gesamtaufwand: 150 h

Prüfung

Prüfungsform	Portfolioprüfung: <ul style="list-style-type: none">• 3 Präsentationen, je 20-30 Minuten, 75%• Studienarbeit, 8-12 Seiten, 25%
Ergänzende Hinweise zur Prüfungsform	Die Präsentationen sind Gruppen-Präsentationen. Studienarbeit: in Gruppenarbeit soll ein während der Veranstaltung erarbeitetes Geschäftsvorhaben als Business Plan angefertigt werden.

Prüfungsnummer	IN 3970395, 2970893 TI 3976611, 2976715 WI 3975813 IIS 9775172
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Unternehmertum ist heute eine der lebendigsten Disziplinen und kann erlernt werden. In diesem Kurs:

- entwickeln Studierende Ihre eigene Geschäftsidee und durchdenken sie von A-Z.
- Studierende lernen eine systematische und strukturierte Herangehensweise an Innovation und Unternehmertum kennen (Strukturierung von Wertschöpfung, Potenzialanalysen, Rapid Prototyping, etc.)
- Studierende wenden zahlreiche Innovationsmethoden und Innovationswerkzeuge an (Value Proposition Canvas, Business Model Canvas, UX-Design, etc.)
- Wird die Kompetenz der Präsentation aktiv gefördert, indem Fortschritte regelmäßig vorgestellt werden müssen

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage:

- Eine innovative Geschäftsidee (gewinnorientiert oder nicht gewinnorientiert) von A bis Z (BYO - bring your own, DYOW - develop your own) zu durchdenken
- Innovationsmethoden eigenständig anzuwenden
- Geschäftspotenziale zu erkennen und zu bewerten
- Innovationsansätze strukturiert auszuarbeiten

Literaturliste

Aulet, Bill (2013): Disciplined entrepreneurship: 24 steps to a successful startup. John Wiley & Sons.

Nambisan, Satish, et al. (2017): "Digital innovation management." MIS quarterly 41.1. 223-238.

Osterwalder, Alexander; and Pigneur, Yves (2010): Business model generation: a handbook for visionaries, game changers, and challengers. Vol. 1. John Wiley & Sons.

Osterwalder, Alexander (2015): Value proposition design: How to create products and services customers want. John Wiley & Sons.

2.63 Systemnahe Programmierung

Informationen über das Modul

Name / engl.	Systemnahe Programmierung / Systems programming
Kürzel	SNP5.WP
Verantwortlicher	Prof. Dr. Hubert Högl
Fakultät	Fakultät für Informatik
Art	Wahlpflichtmodul
Dauer / Angebot	ein Semester, jeweils im Wintersemester
Lehrveranstaltungen	Systemnahe Programmierung (5 SWS)
Lehrsprache	Das Modul wird in deutscher Sprache unterrichtet.
Lehr-/Lernmethoden	Vorlesung, Seminaristischer Unterricht, Praktikum
Voraussetzungen	Keine
Verwendbarkeit	WPF nur für den Bachelorstudiengang Technische Informatik. Für Informatik (Bachelor) handelt es sich um ein Pflichtfach
Arbeitsaufwand / Zusammensetzung	SWS: 5, CPs: 6, Präsenzzeit: 75 h, Selbststudium: 105 h, Gesamtaufwand: 180 h

Prüfung

Prüfungsform	Klausur, 60 Minuten, keine Hilfsmittel
Prüfungsnummer	TI 3976556, 2976720
Benotung	Gemäß § 20 der APO in der jeweils gültigen Fassung.

Inhalte des Moduls

Einführung in die systemnahe Programmierung am Beispiel einer zeitgemässen Programmiersprache zur sicheren Programmierung (Rust).

- Klärung des Begriffs Systemprogrammierung
- Einsatzbereiche für Systemprogrammierung
- Erstellen von ausführbaren Programmen und Bibliotheken
- Grundlegende Sprachelemente (Variablen, Datentypen, Strings, Kontrollstrukturen)
- Eigentümerschaft (ownership), Referenzen und Borgen (borrowing)
- Kollektionen
- Fehlerbehandlung
- Generische Programmierung
- Tests
- Zeiger
- Nebenläufigkeit

Qualifikationsziele des Moduls, Lernziele und Kompetenzen

Die Studierenden...

- wissen, welche Bereiche die Systemprogrammierung umfasst.
- kennen die Grundlagen der Systemprogrammierung in der Programmiersprache Rust und können diese anwenden um eigene Programme zu schreiben.
- verstehen den Ablauf von Programmen auf der Maschinenebene und können dadurch die Sprachelemente von Systemprogrammiersprachen optimal anwenden.
- können die Verwendung der verschiedenen Speicherbereiche in einem Programm erklären (u.a. Stack und Heap) und verstehen dadurch die Techniken der Speicherverwaltung in Systemprogrammiersprachen.
- schreiben Testfälle zur Entwicklung von sicherer Software.
- benutzen sichere parallele Sprachkonstrukte zur Beschleunigung der Programmausführung auf Mehrkernprozessoren.

Die Inhalte der Vorlesung werden in einem begleitenden Praktikum an Hand von Übungsaufgaben vertieft.

Literaturliste

Steve Klabnick, Carol Nichols: The Rust Book, 2018 (freier Inhalt)

<https://doc.rust-lang.org/book>

Carlo Milanesi: Beginning Rust, Apress 2018.

Jim Blandy, Jason Orendorff, Leonora F.S. Tindall: Programming Rust, O'Reilly 2021.

2.64 Visual Thinking for Business

Information about the module

engl. Name	Visual Thinking for Business
Code	VISTH.WP
Coordinator	Philip McClenaghan
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course during the semester break. (February/March) and (August/September)
Courses	Visual Thinking for Business (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical classes and workshops
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 10-25 minutes, 40%• Written assignment, 10-15 pages, 60%
Examination number	IN 3970353, 2970849 TI 3976558, 2976659 WI 3975767 IIS 9775160
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Companies in the modern business world are turning to new ways of working such as Design Thinking and Lean Start-Up to keep pace with constantly evolving marketplaces and technological advancements. The visual tools and methods of Visual Thinking support these new working practices by making information, ideas, concepts and processes visible and thus accessible to all.

Visual Thinking extends the verbal and written language using visualization methods that enable the graphic representation of ideas and complex content. In the new world of design thinking, agile innovation, lean start-up, etc., this is essential.

This course is suitable for all students who want to think through new ideas, complex content and procedures in a structured manner and communicate effectively in their professional life.

Qualification aims for the module learning objectives/skills

The students should develop the following skills during the course:

- Basic understanding of the theoretical aspects of visual thinking and visual communication.
- Application and further development of visual storytelling methods.
- The ability to communicate ideas and complex content visually.
- The ability to independently use visual thinking in a business environment.

Reading list

Will be announced in the first lecture.

Index

- 3D-Druck , 8
- ABAP-Grundlagen , 12
- Advanced Security Testing , 16
- Agile Entwicklung eines
Klick-Dummy-Spiels , 20
- Agile Softwareentwicklung (Scrum) ,
24
- Agile Webanwendungen mit Python , 28
- Betriebliche Informationssysteme , 30
- Chancen- & Risikomanagement in
Digitalisierten
Wertschöpfungsnetzen , 32
- Compiler , 36
- Computer Games Development , 40
- Corporate Entrepreneurship , 42
- Datenkommunikation , 46
- Datenkommunikation im Fahrzeug , 48
- Digital Business Leadership Skills , 56
- Digital Transformation in Organizations
, 60
- Digitale Innovationen , 52
- Einführung in die maschinelle
Sprachverarbeitung , 64
- Einführung in die Robotik , 68
- Elektronische Handelssysteme , 70
- Embedded Linux , 72
- Flugrobotik , 74
- Formula Student Driverless , 76
- Fullstack-Webentwicklung , 80
- Grundlagen DevOps , 84
- Hard- und Software für das Internet der
Dinge , 86
- Hochschul Innovationsprojekt , 88
- Industrielle Bildverarbeitung , 92
- Industrielle Informationsverarbeitung ,
96
- Informatik und Umwelt , 100
- Integrierte Geschäftsprozesse mit SAP
ERP , 104
- Interaction Engineering , 106
- Interaktive Computergrafik , 110
- IT Sourcing and Cloud Transformation ,
126
- IT-Consulting , 114
- IT-Forensik , 118
- IT-Sicherheit , 122
- Klassische Projekttechniken
modernisiert , 128
- Konzepte der Datenbanktechnologie ,
132
- Künstliche Intelligenz in
sicherheitskritischen
Anwendungen , 136
- Lean IT & Enterprise Architecture , 140
- Linux LPIC , 144
- Methoden der KI , 148
- Mobile Robotik , 152
- Mustererkennung und maschinelles
Lernen , 154
- Network Penetration Testing , 156
- Neuronale Netze und Deep Learning ,
158
- NoSQL , 162
- Objektorientierte Programmierung mit
Python , 164
- Open-Source Softwareentwicklung ,
168
- Praktische Robotik mit Matlab , 172
- Process Intelligence , 176
- Programmieren mit Datenbanken , 178
- Programmieren mit Python , 180
- Programmierung von
Web-Anwendungen , 184
- Project Jupyter , 188
- Projekt - Forschung und Transfer , 192

RFID und NFC Technik , 196

Schwaben Innovation Masterclass , 198

Service Learning Projekt , 206

Software-Projektmanagement , 212

Softwaretechnik für datenintensive

Systeme , 210

Startitup - Entrepreneurial Thinking
and Business Design , 214

Systemnahe Programmierung , 218

Visual Thinking for Business , 222