



Modulhandbuch

Wahlpflichtmodule zum Ba. Studiengang Informatik (B.Sc.)

Hochschule Landshut
gültig ab dem Sommersemester 2022

beschlossen am 18. Januar 2022

Hinweis:

Aufgrund der aktuellen Coronasituation wird die konkrete Prüfungsart und -dauer für jedes Modul spätestens 1 Woche vor Beginn des Prüfungszeitraums festgelegt.

Inhaltsverzeichnis

Auflistung aller angebotenen Wahlpflichtmodule	3
IB645 Grundlagen der Betriebswirtschaftslehre	4
IB730 SPS-Programmierung mit CoDeSys (IEC61131-3)	5
IB760 Bildverarbeitung	6
IB761 Big Data Algorithms and Systems	7
IB764 Internet of Things	8
IB765 Innovationslabor	9
IB768 Cloud Computing	10
IB769 Ethik der KI	11
IB770 IT-Sicherheit II	13
IB771 Cloud-ready Java Enterprise	14
IB772 Einführung in die Digitale Forensik	16
AIF312 Modellbasierte Entwicklung I	18
AIF630 Autonome Fahrzeuge	20
WIF360 Geschäftsprozesse und Organisation	22
WIF460 Operations Research	23
WIF620 Software Engineering III	24
WIF725 Text Mining	25

Auflistung aller angebotenen Wahlpflichtmodule

FWP-Modul	SS	WS	Sem.	Ansprechpartner/ Dozent	Nr.	Sprache
Grundlagen der Betriebswirtschaftslehre		✓ ¹	3.	Herr Eder	IB645	Deutsch
SPS-Programmierung mit CoDeSys (IEC61131-3)		✓	7.	Herr Franzke	IB730	Deutsch
Bildverarbeitung		✓	7.	Prof. Siebert PhD	IB760	Deutsch
Big Data Algorithms and Systems	✓		6.	Prof. Siebert PhD	IB761	Englisch
Internet of Things	✓		4.	Prof. Dr. Khelil	IB764	Englisch
Innovationslabor IoT Projekt	✓	✓	ab 3.	Prof. Dr. Khelil	IB765	Deutsch (Englisch) ²
Cloud Computing	✓		6.	Prof. Dr. Mock	IB768	Englisch
Ethik der KI		✓	3.	Prof. Dr. Busse	IB769	Deutsch
IT-Sicherheit II		✓	7.	Prof. Dr. Uhrmann	IB770	Deutsch
Cloud-ready Java Enterprise		✓	7.	Herr Hanel	IB771	Deutsch
Einführung in die digitale Forensik	✓		4.	Prof. Dr. Scholz	IB772	Deutsch
Machine Learning in the Cloud		✓	7.	Prof. Dr. Mock	KI760	Englisch
Geschäftsprozesse und Organisation		✓	3.	Prof. Dr. Dorfner	WIF360	Deutsch
Operations Research	✓		6.	Prof. Dr. Sagraloff	WIF460	Deutsch
Software Engineering III	✓		6.	Prof. Dr. Scholz	WIF620	Deutsch
Text Mining	✓		4.	Prof. Dr. Busse	WIF725	Deutsch
Modellbasierte Entwicklung		✓	3.	Prof. Dr. Pellkofer	AIF312 ³	Deutsch
Autonome Fahrzeuge	✓		6.	Prof. Dr. Pellkofer	AIF630	Deutsch
Module anderer Fakultäten nur nach Genehmigung durch die Prüfungskommission.						
Module der virtuellen Hochschule Bayern nur nach Genehmigung durch die Prüfungskommission ⁴ .						

¹nur für Studierende mit Studienbeginn ab WS19/20

²Wird in Englisch durchgeführt, wenn englischsprachige Studierende die Veranstaltung besuchen.

³6 SWS / 7 ECTS

⁴Siehe: <https://kurse.vhb.org/VHBPORTAL/kursprogramm/kursprogramm.jsp>

Grundlagen der Betriebswirtschaftslehre

gültig für Studierende mit Studienbeginn ab WS2019/20

IB645

Modulverantwortlicher:	M.Sc. Thomas Franzke
Dozent:	Klaus Eder, Diplom-Betriebswirt (FH)
Studiengang:	Bachelor
Modultyp:	Pflichtfach
Sprache:	Deutsch
Angebot:	im siebten Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	-
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit im Unterricht 90 Stunden Selbststudium
Lehrformen:	4 SWS seminaristischer Unterricht und Übungen
Leistungsnachweise und Prüfung:	schriftl. Prüfung 90 Minuten am Semesterende

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden haben Kenntnisse über die Gründung, den organisatorischen Aufbau, die Funktionen und die Leitung von Unternehmen.

Lehrinhalte:

- Betriebswirtschaftliche Grundbegriffe
- Bedürfnisse, Bedarf, Bedarfsdeckung durch Betriebe
- Motivation (Intrinsische/ extrinsischer Motivation , Maslow, Herzberg usw.)
- Managementmethoden (Lean Six Sigma, Projektmanagement, Balanced Scorecard, Big Data, Überwachungskapitalismus usw.)
- Unternehmensgründung (Rechtsformen von Unternehmen, Gründungsformalitäten, Standortwahl)
- Unternehmensführung (Führungsstile, Planung, SWOT-Analysen usw.)
- Entscheidungsfelder im Lebenszyklus eines Unternehmens (Mission, Vision, Werte, Produktlebenszyklus)
- Materialwirtschaft/ Produktionswirtschaft (Fertigungstypen, PPS)
- Marketing (Marketing Mix)
- Personalwirtschaft (Personalbedarf, -einstellung, Förderung, Motivation und Freistellung, Arbeitsleistung)
- Finanzierung von Unternehmen (Außen- und Innenfinanzierung, Liquiditätsplanung)
- Betriebliches Rechnungswesen, Kosten- und Leistungsrechnung, Betriebliche Kennzahlen, Bilanzen
- Aufbau und Ablauforganisation von Unternehmen

Literatur:

Bestmann, Uwe: Kompendium der Betriebswirtschaftslehre, Oldenbourg.
 Wöhe Günter, Döring Ulrich: Einführung in die Allgemeine Betriebswirtschaftslehre, Vahlen.
 Jürgen Härdler, Torsten Gonschorek (Hrsg.) Betriebswirtschaftslehre für Ingenieure

SPS-Programmierung mit CoDeSys (IEC61131-3) IB730

Modulverantwortlicher:	Thomas Franzke M.Sc.
Dozent:	Thomas Franzke M.Sc.
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Deutsch
Angebot:	im Wintersemester.
Dauer:	Ein Semester
Vorkenntnisse:	Erster Studienabschnitt oder vergleichbare Kenntnisse
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im seminaristischen Unterricht, 30 Stunden Präsenzzeit im Praktikum, 90 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS begleitendes Praktikum
Leistungsnachweise und Prüfung:	schriftl. Prüfung 90 Minuten am Semesterende

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Kenntnis über das selbständige Erstellen von Steuerungssoftware, hauptsächlich auf Basis von Ablaufsprache (AS) und strukturiertem Text (ST), definiert in IEC61131-3.

Lehrinhalte:

Die Studierenden erhalten einen Überblick über die Funktionsweise von Speicherprogrammierbaren Steuerungen und deren Möglichkeiten der Programmierung. Die Norm IEC61131-3 definiert diverse grafische und textuelle Programmiermodelle, unter anderem die Sprachen ST und AS, auf die besonders vertieft eingegangen werden soll. Die Programmier- und Laufzeitumgebung CoDeSys realisiert diese Modelle und wird daher im Unterricht und im Praktikum verwendet.

Weitere Themen sind die Anbindung an externe Komponenten, Prozessvisualisierung und aktuelle Bus-Systeme.

Literatur:

John Karl-Heinz, Tiegelkamp Michael, SPS-Programmierung mit IEC 61131-3, Springer

Wellenreuther Günter, Zastrow Dieter, Automatisieren mit SPS - Theorie und Praxis, Vieweg + Teubner
DIN EN 61131-3:2003-12, Beuth

Bildverarbeitung

IB760

Modulverantwortlicher:	Prof. Andreas Siebert, Ph.D
Dozent:	Prof. Andreas Siebert, Ph.D
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Deutsch
Angebot:	im Wintersemester
Dauer:	ein Semester
Vorkenntnisse:	Kenntnisse in der Java-Programmierung
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im seminaristischen Unterricht, 30 Stunden Präsenzzeit im Praktikum, 90 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS begleitendes Praktikum
Leistungsnachweise und Prüfung:	Leistungsnachweise im Praktikum, schriftliche Prüfung 90min. am Ende des Semesters.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden kennen Aufbau und Arbeitsweise von Bildverarbeitungssystemen und den zugrunde liegenden typischen Bildverarbeitungsoperatoren. Sie wissen, welche Art von Problemen mit maschinellem Sehen gelöst werden können und kennen Beispiele dazu. Sie können aus den Bildverarbeitungsoperatoren Anwendungen zusammensetzen und deren Grenzen abschätzen.

Lehrinhalte:

ImageJ
 Histogramme, Kanten, Filter
 Hough-Transformation, RANSAC
 Biometrie (Iriserkennung)
 Abtasttheorem, Fourier Transformation
 Anwendungen

Literatur:

W. Burger, M. Burge: Digitale Bildverarbeitung: Eine algorithmische Einführung mit Java, Springer, 2015.
 Weitere Literatur in der Veranstaltung

Big Data Algorithms and Systems

IB761

Modulverantwortlicher:	Prof. Andreas Siebert, Ph.D
Dozent:	Prof. Andreas Siebert, Ph.D
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Englisch
Angebot:	im Sommersemester
Dauer:	ein Semester
Vorkenntnisse:	Algorithmen und Datenstrukturen; Programmierkenntnisse
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im seminaristischen Unterricht, 30 Stunden Präsenzzeit im Praktikum, 90 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS begleitendes Praktikum
Leistungsnachweise und Prüfung:	Leistungsnachweis im Praktikum, schriftl. Prüfung 90 Minuten am Semesterende.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden sind mit grundlegenden Algorithmen und deren Analyse im Big Data Bereich vertraut und können diese anwenden. Sie kennen die Problematiken von unsicheren und unstrukturierten Datenmengen.

Lehrinhalte:

- Online-Algorithmen
- Competitive Analysis
- High-Frequency Trading
- Punkt-Suche, Konvexe Hülle, Voronoi-Diagramme
- Textanalyse-Algorithmen
- Recommender Systeme
- Data Privacy, Datenbankrekonstruktionen
- Aktuelle Themen

Literatur:

Verschiedene Artikel

Internet of Things (IoT)

IB764

Modulverantwortlicher:	Prof. Dr. Abdelmajid Khelil
Dozent:	Prof. Dr. Abdelmajid Khelil
Studiengang:	Bachelor
Modultyp:	FWPF aus dem Bereich IF
Sprache:	Englisch
Angebot:	im Sommersemester
Dauer:	ein Semester
Vorkenntnisse:	Erster Studienabschnitt oder vergleichbare Kenntnisse
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 90 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	schriftliche Prüfung 90 min.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Lernziel ist die Vermittlung von Kompetenzen im Bereich der vernetzten intelligenten Objekte. Die Studierenden lernen die technologischen Grundlagen des Internet der Dinge (Internet of Things, IoT), z.B. intelligente Objekte, Protokolle, Architekturen, Energieeffiziente SW-Entwicklung, etc.

Lehrinhalte:

Eingebettete Systeme sind heute allgegenwärtig und werden zunehmend mit dem, bzw. über das Internet vernetzt. Der Begriff IoT drückt dabei den Trend der intelligente Vernetzung aller Dinge aus, um den Menschen in seinen Tätigkeiten unmerklich zu unterstützen. In diesem Modul soll den Studierenden die Konzepte und Werkzeuge von IoT vermittelt werden: Die wichtigsten aktuellen Anwendungsgebiete; Elemente der Vernetzung; typische Aktoren und Sensoren; Protokolle (insb. MQTT, CoAP); SW-Plattformen und Interoperabilität. Das Praktikum vertieft das in der Vorlesung erworbene Wissen in ausgewählten Praxisprojekten. Dabei werden verschiedenen IoT Plattformen (z.B. Arduino, Raspberry Pi und Libelium) verwendet um unterschiedliche IoT-Anwendungen (Smart City, Smart Building, eHealth, Smart Agriculture, Industrie 4.0, etc) zu implementieren.

Literatur:

- [1] Jean-Philippe Vasseur, Adam Dunkels, Interconnecting Smart Objects with IP: The next Internet, Morgan Kaufmann, 2010
- [2] Adrian McEwen, Hakim Cassimally, Designing the Internet of Things, John Wiley & Sons; November 2013
- [3] Fleisch, E.: Das Internet der Dinge, Springer 2005
- [4] Charles Bell, Beginning Sensor Networks with Arduino and Raspberry Pi, Apress; Auflage: 2013

Innovationslabor

(IoT-Projekt)

IB765

Modulverantwortlicher:	Prof. Dr. Abdelmajid Khelil
Dozent:	Prof. Dr. A. Khelil, Prof. Dr. E. Kromer, Prof. Dr. M. Mock, Prof. Dr. J. Uhrmann
Studiengang:	Bachelor
Modultyp:	FWPF aus dem Bereich IF
Sprache:	Deutsch / Englisch
Angebot:	jedes Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	Programmieren I, Software Engineering I
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	150 Stunden nicht ständig betreute Projektarbeit im Labor
Lehrformen:	4 SWS nicht ständig betreute Projektarbeit. Eigenverantwortliches Arbeiten der Studierenden in Teams von einer kritischen Größe, so dass das Auftreten typischer Schnittstellenprobleme gewährleistet ist, regelmäßige Projekttreffen mit dem Betreuer. Präsentation des Projektergebnisses zum Semesterende in einem Seminar.
Leistungsnachweise und Prüfung:	Benotete individuelle schriftliche Ausarbeitung jedes Teammitglieds zum eigenen Beitrag im Projekt, im Team erstellte Gesamtdokumentation, im Team durchgeführte Präsentation des Projekts. Das Gesamtprojekt wird benotet. Die Note der Teammitglieder wird als Mittelwert aus der individuellen Note und der Projektnote gebildet.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden identifizieren reale Problemstellungen und erkennen die Problematik der Erstellung komplexer Lösungen mit Hilfe unterschiedlichster IoT-Plattformen. Sie sind in der Lage die Umgebung der Problemstellung zu analysieren und können diese in Zusammenarbeit mit Unternehmen im Vorfeld diskutieren. Kenntnisse über Design Thinking, agiles Projektmanagement und eigenverantwortlicher Durchführung von Projekten erwerben Studierende in der Teamarbeit. Sie sind in der Lage, fachübergreifende Kenntnisse anzuwenden, den Problemsteller in das Projekt agil einzubinden und Arbeitsergebnisse zu präsentieren.

Lehrinhalte:

Die kooperierenden Unternehmen bieten den Studierenden reale Problemstellungen aus den wichtigsten IoT-Domänen, wie etwa Smart Agriculture, Smart Building, Smart Energy, Smart Production, eHealth etc. Die Problemstellung wird anhand definierter Anwendungsfälle detailliert beschrieben. Zusätzlich werden zur Problemstellung die Aspekte IoT Cloud und IoT Security untersucht. Die Studierenden werden vom Dozenten und dem Coach des Innovationslabors fachlich betreut.

Literatur:

Siehe Projektbeschreibung. Weitere Anregungen:

- [1] Jean-Philippe Vasseur, Adam Dunkels, Interconnecting Smart Objects with IP: The next Internet, Morgan Kaufmann, 2010.
- [2] Charalampos Doukas, Building Internet of Things with the Arduino, CreateSpace Independent Publishing Platform, 2012.
- [3] Charles Bell, Beginning Sensor Networks with Arduino and Raspberry Pi, Apress; Auflage: 2013.
- [4] E.F. Engelhardt, Sensoren am Raspberry Pi, Franzis Verlag GmbH, 2014.
- [5] Vic (J.R.) Winkler, Securing the Cloud, Syngress, 2011.

Cloud Computing

IB768

Modulverantwortlicher:	Prof. Dr. Markus Mock
Dozent:	Prof. Dr. Markus Mock
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Englisch
Angebot:	im Sommersemester
Dauer:	ein Semester
Vorkenntnisse:	Algorithmen und Datenstrukturen; Programmierkenntnisse
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im seminaristischen Unterricht, 30 Stunden Präsenzzeit im Praktikum, 90 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS begleitendes Praktikum
Leistungsnachweise und Prüfung:	Schriftl. Prüfung 90 Minuten am Semesterende.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden sind mit der Bedeutung von Ressourcenmanagement und dem Konzept der Elastizität vertraut. Sie können Strategien zum Synchronisieren von verteilten Datenquellen erklären. Sie sind in der Lage, die Vor- und Nachteile von virtualisierten Infrastrukturen zu erklären. Sie sind fähig, eine Anwendung, die Cloud Infrastrukture zur Verarbeitung oder Datenspeicherung verwendet, in der Cloud zu starten. Außerdem können sie eine Anwendung angemessen zwischen Client und Cloud Ressourcen strukturieren.

Lehrinhalte:

- Computing and Internet Scale – Cluster, Grids, und Netze
- Cloud Dienste (z.B. AWS, Azure, oder Google Cloud)
- IaaS, SaaS, PaaS und Ressourcenelastizität
- Virtualisierung, Replikation und Prozessmigration
- Sicherheit in der Cloud, Virtual Private Networks
- Weakly consistent data stores, CAP Theorem
- Verteilte File Systems, z.B. HDFS
- Mapreduce und Hadoop: Paradigma zur verteilten Berechnung

Literatur:

Verschiedene Artikel

Ethik der KI

IB769

Modulverantwortlicher:	Prof. Dr. Johannes Busse
Dozent:	Prof. Dr. Johannes Busse
Studiengang:	Bachelor
Modultyp:	FWP
Sprache:	Deutsch
Angebot:	im Wintersemester
Dauer:	ein Semester
Vorkenntnisse:	-
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 90 Stunden Selbststudium
Lehrformen:	4 SWS seminaristischer Unterricht
Leistungsnachweise und Prüfung:	Studienarbeit

Qualifikationsziele und Inhalte:**Qualifikationsziele:**

Die Literatur zur "Ethik der KI" wächst seit einigen Jahren stark an. Einzelne Argumentationen klingen zunächst plausibel, halten einer genaueren Nachfrage jedoch nicht immer stand und kollidieren teilweise sogar mit unserer europäischen Rechtsordnung. Fast immer besteht in den Grundannahmen ein verborgener normativer Bias, der nur selten so ausreichend explizit gemacht wird, dass die jeweilige Position transparent einer Kritik unterzogen werden kann.

Wir untersuchen unseren Gegenstand nicht als materiale Wertethik, sondern vorwiegend aus Perspektive der deskriptiven Ethik und der Meta-Ethik. Eine moralische Belehrung findet nicht statt. Inhaltliches Lernziel ist die Kenntnis, Reflektion und Verortung einschlägiger Grundannahmen und Argumentationsmuster, die wir aus den derzeit zahlreichen Neuerscheinungen im Feld "Ethik der KI" herausarbeiten.

Das Modul verschafft eine Grundorientierung im Bereich der Ethik / Moralphilosophie, um Texte wie die unten exemplarisch genannten (a) zunächst inhaltlich genau zu verstehen (Lernzielkategorie "Wissen") und dann (b) auch fundierter bewerten zu können ("Kompetenzen").

Lehrinhalte:

Ein wesentlicher Inhalt der Veranstaltung besteht aus einem reflektierten Verständnis unserer Fokus-Lektüre (s.u.), die nach Bedarf durch ausgewählte Theoriebestandteile aus der systematischen Literatur der philosophischen Ethik unterfüttert wird.

Anwendung findet unser Verständnis in der aktuellen Literatur, insbes. aktuelle Studien zur Ethik der KI, automatisierten Entscheidungen, algorithmengestütztem Handeln etc.

Die Argumentationen unserer Fokus-Lektüre werden nach Bedarf unterfüttert durch ausgewählte Theoriebestandteile aus der systematischen Literatur der philosophischen Ethik. Ihre Anwendung findet unser Verständnis in der aktuellen Literatur, insbes. aktuelle Studien zur Ethik der KI, automatisierten Entscheidungen, algorithmengestütztem Handeln etc.

Medien und Methoden: Die Veranstaltung ist als Lese-, Schreib- und Diskurs-Seminar angelegt.

- Alle Teilnehmer bereiten defaultmäßig alle Texte aller Sitzungen vor. Neben der reinen Lektüre gehören hier auch eine knappe Zusammenfassung sowie eine eigene diskursive Stellungnahme dazu.
- Ergänzend bereitet jeder Teilnehmer je eine Sitzung vertieft vor und gibt eine kurze Einführung in das Thema, das dann auf Grundlage der gemeinsam diskutiert wird (Anwesenheit zwingend erforderlich).

Die Gesamtheit aller so erstellten Schriftstücke ergibt quasi von selbst die Studienarbeit, mit der das Modul dann formal abgeschlossen wird.

Weltanschauungs-Neutralität: Wir behandeln unsere Themen säkular aus einer den Idealen der Aufklärung verpflichteten sog. Philosophischen Ethik". Wir verzichten auf Argumentationen, die sich an zentraler Stelle auf religiöse Glaubensinhalte berufen, schaffen bei entsprechender Nachfrage der Teilnehmer jedoch Raum, auch solche Perspektiven auszutauschen. Die Veranstaltung ist damit für Angehörige aller Glaubensrichtungen und insbesondere auch für nicht-religiöse Menschen geeignet.

Literatur:

- Julian Nida-Rümelin und Nathalie Weidenfeld: Digitaler Humanismus. Eine Ethik für das Zeitalter der Künstlichen Intelligenz. Piper 2018

Eine wesentliche Rolle spielen aktuelle Studien zur Ethik der KI, automatisierten Entscheidungen, algorithmengestütztem Handeln etc. (Auswahl): semesterweise aktualisierte Literatur siehe die erweiterte Homepage zur Veranstaltung: http://jbusse.de/public/Modul_etki.html.

IT-Sicherheit II

IB770

Modulverantwortlicher:	Prof. Dr. Johann Uhrmann
Dozent:	Prof. Dr. Johann Uhrmann
Studiengang:	Bachelor
Modultyp:	Wahlpflichtmodul
Sprache:	Deutsch
Angebot:	im Wintersemester
Dauer:	ein Semester
Vorkenntnisse:	
Voraussetzungen:	IT-Sicherheit, Programmieren I oder Programmieren II
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 90 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	schriftliche Prüfung 90 Minuten am Semesterende.

Qualifikationsziele und Inhalte:**Qualifikationsziele:**

Absicherung von Netzwerken gegen Angriffe, Cloud Security, Behandeln von Sicherheitsvorfällen

Lehrinhalte:

- Angriffe auf Netzwerke erkennen
- Abwehrmechanismen
- Vorfallsbehandlung
- relevante IT-Sicherheitsstandards
- Analyse von Schadsoftware
- aktuelle Entwicklungen in der IT-Sicherheit

Literatur:

Michael Messner, Hacking mit Metasploit, dpunkt Verlag, 2015.

Chris Eagle, The IDA Pro Book, no starch press, 2011.

Weitere Literatur wird in der Vorlesung bekanntgegeben.

Cloud-ready Java Enterprise

IB771

Modulverantwortlicher:	Thomas Franzke M.Sc.
Dozent:	Dipl. Inf. (FH) Thomas Hanel
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Deutsch
Angebot:	im Wintersemester (nur im 7. Semester!)
Dauer:	ein Semester
Vorkenntnisse:	Gute Java-Kenntnisse und Linux/UNIX-Kenntnisse zwingend erforderlich, sowie erster Studienabschnitt oder vergleichbare Kenntnisse
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im seminaristischen Unterricht, 30 Stunden Präsenzzeit im Praktikum, 90 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS begleitendes Praktikum
Leistungsnachweise und Prüfung:	Mündliche Prüfung am Ende des Semesters.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Dieses Modul bietet eine Einführung in die grundlegenden Eigenschaften und Bestandteile der Jakarta EE Plattform, Eclipse Microprofile und Quarkus sowie deren Einsatzmöglichkeiten und Verwendungen. Im Zuge der Implementierung und Anpassung einer mehrschichtigen "Enterprise Application" werden Methoden, Techniken und Werkzeuge für die Entwicklung von "Cloud-ready" Applikationen vorgestellt sowie dedizierte Aspekte erläutert. Nach erfolgreichem Abschluß dieses Moduls

- können Studierende die unterschiedlichen Aspekte der verwendeten Frameworks und Standards beurteilen
- können Studierende mehrschichtige Applikationen auf Basis Jakarta EE (Java EE) (bzw. Eclipse Microprofile, Quarkus) entwerfen und implementieren
- haben Studierende die Erstellung und Verwendung von Container-basierten Applikationen für den Einsatz in Cloud-Plattformen kennengelernt
- wurden git, Maven, Docker, etc. in realistischen Einsatzszenarien verwendet
- haben Studierende Einblicke in die Nutzung von Infrastructure-as-Code Prinzipien und etablierte Software-Entwicklungsprozesse bekommen

Lehrinhalte:

- Einführung: Überblick Jakarta EE (JEE) Plattform, Komponenten und Zuständigkeiten, Architektur
- Bestandteile von JEE-Applikationen: JSF, EJB, CDI, JAX-RS, Security
- Persistenz: JPA in JEE und Java SE
- Eclipse Microprofile: Config, Health, Metrics
- Cloud-ready: JEE, Eclipse Microprofile und Quarkus
- Einblicke in die SW-Entwicklung im Geschäftsumfeld: Build-Automatisierung, Continuous Integration, Testing, Qualitätssicherung, Container-Lösungen, IaC

Die Themen werden zum größten Teil direkt am Rechner behandelt. In einer Art Workshop wird gemeinsam eine Applikation schrittweise entwickelt und dabei anschaulich Lerninhalte und Techniken vorgestellt. Ein eigener, mit mindestens 8GB RAM ausgestatteter, leistungsfähiger Laptop wird (u.a. für die Ausführung virtueller Maschinen) dringend empfohlen.

Literatur:

Aktuelle Empfehlungen werden in der Veranstaltung bekannt gegeben

Daschner, Sebastian: "Architecting Modern Java EE Applications", Packt Publishing, 2017

Späth, Peter: "Beginning Jakarta EE: Enterprise Edition for Java: From Novice to Professional", Apress, 2019

Einführung in die Digitale Forensik

IB772

Modulverantwortlicher:	Prof. Dr. Peter Scholz
Dozent:	Prof. Dr. Peter Scholz
Studiengang:	Bachelor
Modultyp:	Wahlpflichtfach
Sprache:	Folien/Unterlagen in Englisch, Vorlesung in Deutsch oder Englisch
Angebot:	im Sommersemester
Dauer:	ein Semester
Vorkenntnisse:	Kenntnisse in Grundlagen der Informationssicherheit und Kryptografie
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit im seminaristischen Unterricht, 105 Stunden Selbststudium.
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS Übung, aufgeteilt in Gruppen- Einzel- und Projektarbeit
Leistungsnachweise und Prüfung:	Leistungsnachweis der Übung/Projektarbeit, schriftliche Prüfung 90 Minuten

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Dieses Modul gibt eine Einführung in die digitale Forensik. Es deckt dabei theoretische, praktische und rechtliche Aspekte ab. Der erste Teil konzentriert sich auf die Anforderungen, die nötig sind um ein guter Computerforensiker zu werden. Dieser Personenkreis ist bei Ermittlungsbehörden und Industrieunternehmen gleichermaßen gesucht. Weiterhin werden Methoden, Techniken und Werkzeuge zur forensischen Auswertung von Computern und Smartphones vorgestellt. Dabei werden begleitend stets rechtliche Aspekte, wie beispielsweise der Datenschutz, beleuchtet. Nach erfolgreichem Abschluss dieses Moduls:

- können sie erste forensische Untersuchungen bzw. Auswertungen bei Computern, Smartphones und Tablets auf Basis von Richtlinien durchführen,
- wissen Studierende, wie sie forensische Berichte und Gutachten schreiben,
- können sie die wesentlichen rechtlichen Aspekte und Zusammenhänge verstehen und hieraus ihre Pflichten als Computerforensiker ableiten,
- können sie einschlägige Softwarewerkzeuge anwenden,
- können sie den einschlägigen Stand der Wissenschaft, Technik und Praxis beurteilen.

Lehrinhalte:

- Einführung, Motivation, Geschichte, Anforderungen
- Beschlagnahme und Auswertung von Beweismitteln
- Methoden und Prozesse der digitale Forensik (Analyse von Betriebssystemen, Dateisystemen, Dateien und Datenbanken, Erkennen von Eindringlingen)
- Fallstudien
- Überblick zu forensischen Software Tools (kommerziell, Open Source)
- Erstellung von Berichten und Gutachten
- Digitale Forensik mobiler Endgeräte

Literatur:

Wird zeitnah und aktuell in der ersten Vorlesungsstunde bekannt gegeben.

Modellbasierte Entwicklung I

AIF312

Modulverantwortlicher:	Prof. Dr. Martin Pellkofer
Dozent:	Prof. Dr. Martin Pellkofer
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich AIF
Sprache:	Deutsch
Angebot:	im dritten Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	Programmieren I
Voraussetzungen:	Zulassung zum Praktikum erfolgt bei bestandener Modulprüfung in Programmieren I oder Programmieren II
Leistungspunkte:	7
Arbeitsaufwand:	60 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 120 Stunden Selbststudium
Lehrformen:	4 SWS seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	schriftliche Prüfung von 90 Minuten am Semesterende

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden verstehen die Methode der modellbasierten Softwareentwicklung und die Unterschiede zur traditionellen Vorgehensweise. Sie sind in der Lage, mit Hilfe der Werkzeugkette Matlab/Simulink/Stateflow modellbasiert komplexe Fahrzeugfunktionen zu entwickeln und die Funktionen in Festkommaarithmetik zu formulieren. Sie können die Modelle der Fahrzeugfunktionen auf verschiedenen generischen Plattformen in Echtzeit ablaufen lassen und über die I/O-Kanäle mit einem äußeren technischen Prozess verbinden. Die Studenten sind in der Lage, aus den Modellen der Fahrzeugfunktionen Quelltext für die Sprache C zu generieren. Sie können dabei den Generierungsprozess so anpassen, dass der Quelltext sich in eine vorgegebene Software-Umgebung auf einem eingebetteten System einfügt.

Lehrinhalte:

- traditioneller und modellbasierter Entwicklungsprozess
- Anforderungen an Modelle und Modellierungstechniken
- Modellierungssprachen und ihre Eigenschaften
- Matlab™: Datentypen, Matrix- und Feldoperationen, Prozeduren und Funktionen, numerisches Lösen von Differentialgleichungen;
- Simulink™: Modellierung dynamischer Systeme durch hierarchische Blockschaltbilder, Stapelverarbeitung von Simulationen mit Variation der Parameter, Erstellen eigener Blockbibliotheken und S-Funktionen, Einbinden von handgeschriebenem Quellcode in das Modell;
- Stateflow™: Ereignisdiskrete Modellierung mit hierarchischen Zustandsautomaten
- Automatische Code-Generierung mit Matlab-, Simulink- und Embedded-Coder™
- Reversibles Umschalten zwischen Gleitkommaarithmetik und Festkommaarithmetik
- Rapid Prototyping mit verschiedenen Plattformen

Literatur:

- A. Angermann, M. Beuschel, M. Rau, W. Wohlfarth: Matlab-Simulink-Stateflow, Oldenbourg, 6. Auflage, München 2009
- P. Marwedel: Eingebettete Systeme, Springer Verlag, Heidelberg, 2008
- User Manuals der Matlab-Toolboxen Matlab Coder™, Simulink Coder™, Embedded Coder™, Fix- Point Designer™ von The Mathworks
- K. Berns, B. Schürmann, M. Trapp: Eingebettete Systeme, Vieweg+Teubner, 1. Auflage, Wiesbaden, 2010

Autonome Fahrzeuge

AIF630

Modulverantwortlicher:	Prof. Dr. Martin Pellkofer
Dozent:	Prof. Dr. Martin Pellkofer
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich AIF
Sprache:	Deutsch
Angebot:	im sechsten Studiensemester, erstmalig im Sommersemester 2022
Dauer:	ein Semester
Vorkenntnisse:	Programmieren I (Programmierkenntnisse in C/C++), Modellbasierte Entwicklung I (Grundkenntnisse in Matlab/Simulink)
Voraussetzungen:	
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 90 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS Praktikum in kleinen Gruppen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	schriftliche Prüfung von 90 Minuten am Semesterende

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden kennen den Stand der Technik bei hoch automatisierten und autonom fahrenden Landfahrzeugen. Dies beinhaltet die eingesetzte Sensorik, Aktuatorik, Algorithmik, Navigation und Entscheidungsfindung, sowie das Systemdesign. Die Studierenden haben sich ferner mit den ethischen und rechtlichen Fragen auseinandergesetzt, welche autonome Fahrzeuge aufwerfen.

Lehrinhalte:

- Stand der Technik bei hoch automatisierten und autonomen Landfahrzeugen
- Sensoren autonomer Fahrzeuge: z. B. Inertialsensoren, Ultraschallsensoren, Radar, 3D Time-of-Flight-Kamera, Lidar, Monokameras, Stereokamera
- Maschinelles Sehen:
 - Projektion, Bildvorverarbeitung, Glättungsfiler
 - Einzelbildmerkmale und Korrespondenzmerkmale
 - Stereoskopie: Rektifikation, Epipolarbedingung, Disparität, Motion Stereo
- Sensordatenfusion und Zustandsschätzung:
 - Prädiktion und Innovation, Erweiterte Kalmanfilter, Partikelfilter,
 - Positions- und Lagebestimmung mittels Magnetometer, Beschleunigungssensor und Kreisel
 - Schätzung der Pose durch Fusion von IMU- und GPS-Daten mittels Kalman-Filter
 - Schätzung der Zustandsgrößen der Fahrspur und der Position des Ego-Fahrzeugs relativ zur Fahrspur mit einer Monokamera

- Objektverfolgung:
 - Single Object Tracking: kooperatives und nicht-kooperatives Tracking, Interagierende Multi-Modell-Filter
 - Multi Object Tracking: Lösung des Datenassoziationsproblems mit GNN und JPDA, Track-Verwaltung
 - Track-Level-Fusion: Vor- und Nachteile, Problematik bei korreliertem Rauschen
 - Verfolgung von nicht-punktförmigen Objekten: Datenassoziationsproblem bei ausgedehnten Objekten, DBSCAN
 - Schätzung der Zustandsgrößen von Fremdfahrzeugen mit Kamera, Lidar und Radar
- Autonome Navigation:
 - Positionsbestimmung mit Partikelfilter
 - Simultane Positionsbestimmung und Kartierung (SLAM): Pose Graph Optimization (PGO)
 - Pfadfindung und Bewegungsplanung: A*, RRT, RRT*
- 4D-Ansatz
 - Dynamische Objektdatenbank: Lagebeschreibung durch homogene Transformationsmatrizen, Szenenbaum
 - Repräsentation der Fähigkeiten des autonomen Systems
 - Wissensrepräsentation und Entscheidungsfindung
 - Steuerung der ablaufenden Aktionen und Vorhalten von Alternativen
- Aktives Sehen:
 - Der Sehprozess von Wirbeltieren als Vorbild
 - Steuerung der Wahrnehmungsprozesse und der Aufmerksamkeit
 - Blickrichtungssteuerung für Zweiachsen-Kameraplattformen
- Anwendungen von Methoden aus den Bereichen künstliche Intelligenz und maschinelles Lernen zur Objekterkennung:
 - Cascade Classifiers: "Haar-like"-Merkmale, schwache und starke Klassifikatoren, Boosting
 - Beispiel: Mustererkennung und Klassifikation zur Erkennung von Verkehrszeichen und Fremdfahrzeugen
 - Klassifikation mit Convolutional Neural Networks (CNN) und YOLO-Netze
- Ethische und rechtliche Fragen beim autonomen Fahren
- Entwicklungsplattformen:
 - Sensor-in-the-Loop-Simulationen mit CarMaker (Fa. IPG) zur Entwicklung von Wahrnehmungsprozessen
 - Entwicklungsarbeiten und Experimente mit autonom fahrenden 1:10-Modellfahrzeugen
 - autonome Navigation mit Robotinos

Literatur:

- H. Winner, S. Hakuli, F. Lotz, C. Singer: Handbuch Fahrerassistenzsysteme, 3. Auflage, Springer, 2015
E. D. Dickmanns: Dynamic Vision for Perception and Control of Motion, Springer, 2007
M. Maurer, J. Ch. Gerdes, B. Lenz, H. Winner (Hrsg.): Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte, Springer, 2015
Ethik-Kommission des Bundesministers für Verkehr und digitale Infrastruktur: Automatisiertes und Vernetztes Fahren, Bericht, 2017
A. Herrmann, W. Brenner: Die autonome Revolution, Frankfurter Allgemeine Buch, 1. Auflage, 2018
R. Henze: Vom Assistierten zum Hoch-Automatisierten Fahren, Dissertation, TU Braunschweig, 2018
H. Cheng: Autonomous Intelligent Vehicles: Theory, Algorithms, and Implementation, Springer, 2011
Dokumentation und Webinare der relevanten Toolboxes von Matlab/Simulink (Fa. The MathWorks)

Geschäftsprozesse und Organisation

WIF360

Modulverantwortlicher:	Prof. Dr. Matthias Dorfner
Dozent:	Thomas Roidl
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich WIF
Sprache:	Deutsch
Angebot:	im dritten Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	-
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit im Unterricht 90 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht mit Übungen 2 SWS Übungen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	Schriftliche Prüfung, 90 Min. Leistungsnachweise im Praktikum.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden entwickeln ein Verständnis für die wesentlichen Fragestellungen der Aufbau- und Ablauforganisation. Sie verstehen die Aufgaben und Vorgehensweisen des Geschäftsprozessmanagements und sind in der Lage, Geschäftsprozesse auf Basis verschiedener Ansätze und Methoden systematisch zu analysieren, zu modellieren, zu optimieren, wobei sie zusätzlich einen Einblick in die Herausforderungen der IT-technischen Implementierung von Geschäftsprozessen gewinnen.

Lehrinhalte:

- Einführung in die Organisationslehre
- Aufbau- und Ablauforganisation
- Organisationsanalyse und -gestaltung
- Bedeutung des Geschäftsprozessmanagements
- Vorgehen beim Geschäftsprozessmanagement
- Dokumentation von Geschäftsprozessen
- Modellierung, Optimierung und Bewertung von Geschäftsprozessen
- IT-technische Implementierung von Geschäftsprozessen

Literatur:

J. Becker, C. Mathas, A. Winkelmann: Geschäftsprozessmanagement, Springer, Berlin 2009
 T. Allweyer: Geschäftsprozessmanagement - Strategie, Entwurf, Implementierung, Controlling. 3. Nachdruck, W3L GmbH, Herdecke 2009
 T. Allweyer: BPMN 2.0 - Business Process Model and Notation: Einführung in den Standard für die Geschäftsprozessmodellierung, 3. aktualisierte Auflage, Berlin: Books on Demand 2015
 A. Gadatsch: Grundkurs Geschäftsprozess-Management – Methoden und Werkzeuge für die IT-Praxis. 7. Auflage, Vieweg-Verlag, Wiesbaden 2013
 A. Gadatsch: Geschäftsprozesse analysieren und optimieren, Wiesbaden: Springer Vieweg 2015
 H. J. Schmelzer, W. Sesselmann: Geschäftsprozessmanagement in der Praxis – Kunden zufrieden stellen, Produktivität steigern, Wert erhöhen. 7. Auflage, Hanser-Verlag, München 2010

Operations Research

WIF460

Modulverantwortlicher:	Prof. Dr. Michael Sagraloff
Dozent:	Prof. Dr. Michael Sagraloff
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich WIF
Sprache:	Deutsch
Angebot:	im vierten Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	Mathematik I und II
Voraussetzungen:	Zulassung zum Praktikum erfolgt bei bestandener Prüfung in Mathematik I
Leistungspunkte:	5
Arbeitsaufwand:	60 Stunden Präsenzzeit im Unterricht 90 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht 2 SWS Übungen (14tägig 4 Stunden)
Leistungsnachweise und Prüfung:	Schriftliche Prüfung, 90 Min. Leistungsnachweis im Praktikum.

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Die Studierenden sind mit den wichtigsten Themengebieten des Operations Research wie (nicht) lineare (ganzzahlige) Optimierung, Optimierung in Graphen, Netzplantechnik, sowie heuristische und probabilistische Verfahren vertraut. Sie sind nach der Vorlesung in der Lage, neue Algorithmen leicht zu verstehen, an eingeführten Verfahren Modifikationen vorzunehmen oder und auch selbst Verfahren zu entwickeln. Zudem können sie für Standardprobleme der industriellen Praxis das richtige OR-Verfahren auswählen und anwenden.

Lehrinhalte:

- Einführung und Grundbegriffe des Operations Research
- Lineare Optimierung (Simplex Algorithmus, Dualität, Sensitivitätsanalyse)
- Ganzzahlige lineare Optimierung (Branch and Bound-Algorithmus, Gomory Verfahren)
- Nichtlineare Optimierung (Newton Verfahren, Lagrange Verfahren, Gradientenverfahren, Simulated Annealing)
- Optimierung in Graphen (Algorithmen von Dijkstra, Kruskal, und Prim)
- Netzplantechnik (Modellierung, Berechnung kritischer Pfade, Pufferzeiten)
- Transport- und Tourenplanung als Beispiel für Standard-Probleme der industriellen Praxis

Literatur:

Domschke W., Drexl A.: „Einführung in Operations Research“, 7. Auflage, Springer, Berlin, 2007
 Hillier F.S., Lieberman G.J.: „Introduction to Operations Research“, 9. Auflage, McGraw Hill, 2012
 Heinrich G., Grass J.: „Operations Research in der Praxis“, Oldenbourg Verlag, 2006
 Neumann K., Morlock M.: „Operations Research“, 2. Auflage, Hanser Verlag, 2004
 Zimmermann H.-J.: „Methoden und Modelle des Operations Research für Ingenieure, Ökonomen und Informatiker“, 2. Auflage, Vieweg Verlag, 2008
 Zimmermann W.: „Operations Research - Quantitative Methoden zur Entscheidungsvorbereitung“, Oldenbourg Verlag, 1999
 Ulrich Kathöfer und Ulrich Müller-Funk: „Operations Research“, 2017, 3. Auflage, 256 Seiten, UVK Verlagsgesellschaft mbH

Software Engineering III

(Secure Software Engineering)

WIF620

Modulverantwortlicher:	Prof. Dr. Peter Scholz
Dozent:	Prof. Dr. Peter Scholz
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich WIF
Sprache:	Folien/Unterlagen in Englisch, Vorlesung in Deutsch oder Englisch
Angebot:	im sechsten Studiensemester
Dauer:	ein Semester
Vorkenntnisse:	Software Engineering I (Überblick über alle Phasen der Softwareentwicklung und die dort eingesetzten Methoden und Verfahren); Software Engineering II (Objektorientierte Analyse und Design von Software, UML), Informationssicherheit
Voraussetzungen:	Zulassung erfolgt bei bestandener Prüfung in Programmieren I oder Programmieren II
Leistungspunkte:	5
Arbeitsaufwand:	45 Stunden Präsenzzeit im Unterricht 105 Stunden Selbststudium
Lehrformen:	2 SWS seminaristischer Unterricht 1 SWS Übung zur Erstellung einer Hausarbeit oder Projekt(gruppen)arbeit 1 SWS Erstellung einer Hausarbeit oder Projekt(gruppen)arbeit
Leistungsnachweise und Prüfung:	Leistungsnachweis, schriftliche Prüfung 90 Minuten

Qualifikationsziele und Inhalte:

Qualifikationsziele:

Aufbauend auf den Grundlagen des Software Engineerings in den Modulen WIF210 und WIF310 haben die Studierenden vertieften Einblick in ausgewählte spezielle Themengebiete des Software Engineering. Insbesondere haben sie verstanden und eingeübt, wie sichere Software entwickelt werden kann. Sichere Software ist gegen absichtliche Angriffe geschützt. Die Studierenden lernen, wie Sicherheit im Entwicklungsprozess verankert wird.

Lehrinhalte:

- Angriffe auf Software
- Softwaresicherheit aus Nutzer- und Angreifersicht
- Formulierung von Sicherheitsanforderungen
- Modellierung von Bedrohungen
- Sicherer Softwareentwurf
- Sicheres Programmieren
- Qualitätssicherung von sicherer Software

Literatur:

Wird zeitnah und aktuell in der ersten Vorlesungsstunde bekannt gegeben. Darüber hinaus:
Sachar Paulus: „Basiswissen Sichere Software“, dpunkt.verlag, Heidelberg, 2011.
Walter Kriha, Roland Schmitz: „Sichere Systeme“, Springer-Verlag, Berlin, Heidelberg, 2009.

Text Mining

WIF725

Modulverantwortlicher:	Prof. Dr. Johannes Busse
Dozent:	Prof. Dr. Johannes Busse
Studiengang:	Bachelor
Modultyp:	FWP aus dem Bereich WIF
Sprache:	Deutsch
Angebot:	im Sommersemester
Dauer:	ein Semester
Vorkenntnisse:	-
Voraussetzungen:	-
Leistungspunkte:	5
Arbeitsaufwand:	30 Stunden Präsenzzeit im Unterricht 30 Stunden Präsenzzeit im Praktikum 90 Stunden Selbststudium
Lehrformen:	4 SWS seminaristischer Unterricht
Leistungsnachweise und Prüfung:	Studienarbeit;

Qualifikationsziele und Inhalte:**Qualifikationsziele:**

Die TN können unter Linux in Python mit einschlägigen Bibliotheken (wie z.B. scikit-learn, SpaCy, Gensim, NLTK) schwach strukturierte Texte sowie Tabellendaten aus dem Bereich der Wirtschaftsinformatik mit Verfahren des Machine Learning analysieren, Textähnlichkeit feststellen, klassifizieren, korrelierte Daten vorhersagen.

Praktisch beschäftigen wir uns mit der Bepreisung von Immobilien (Boston Housing Dataset), der Text-Klassifikation (20 Newsgroups Dataset) oder der Sentiment Analysis aufgrund von Produktbewertungen. An weiteren Anwendungsfällen diskutieren wir exemplarisch (Weiss 2015): 8.1 Market Intelligence from the Web — 8.3 Generating Model Cases for Help Desk Applications — 8.8 Mining Social Media — 8.9 Customized Newspapers

Die hier vermittelte Technologie bildet eine Grundlage für weiterführende KI-Anwendungen in der Wirtschaftsinformatik.

Lehrinhalte:

- Grundlagen des dsc-lab: Linux, bash, Jupyter Notebook, Publizieren mit Jupyterbook etc.
- Grundlagen des Machine Learning : Klassifikation, Regression, Modellevaluation, Confusion Matrix etc.
- Grundlagen der Informationsextraktion aus Text: Regex, NLP mit Spacy etc.
- Theorie des Information Retrieval (IR) from text

Die Veranstaltung beruht auf einem virtuellen Data Science Laboratory <http://jbusse.de/dsci-lab/>, das den Studierenden unter VirtualBox als virtuelle Xubuntu-Maschine zur Verfügung gestellt wird.

Literatur:

Bücher:

- Tobias Roelen-Blasberg: Automatisierte Präferenzmessung: Extraktion und Evaluation von Produktattributen auf Basis von Online-Rezensionen. Springer 2019.
- Winfried Gödert, Jessica Hubrich und Matthias Nagelschmidt: Semantic Knowledge Representation for Information Retrieval. De Gruyter Saur 2014.
- Weiss, Sholom M.: Fundamentals of Predictive Text Mining. Springer 2nd ed. 2015
- Aggarwal, Charu C.: Machine learning for text (2018)

Online:

- ausgewählte Einführungs-Lectures aus <https://www.kaggle.com/learn/overview>
- SpaCy <https://spacy.io/usage/spacy-101>
- Beautiful Soup <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
- RegEx online zum Üben: <https://regex101.com/> > Python flavor