

MODULHANDBUCH

Master of Science

Master Informatik (FS-MAINF-60)

60 ECTS

Fernstudium

Klassifizierung: weiterbildend

Inhaltsverzeichnis

1. Semester

Modul DLIMITSS: IT-Systeme: Software

Modulbeschreibung	9
Kurs DLIMITSS01: IT-Systeme: Software	11

Modul DLIMITSH: IT-Systeme: Hardware

Modulbeschreibung	15
Kurs DLIMITSH01: IT-Systeme: Hardware	17

Modul DLMDWPMP: Programmieren mit Python

Modulbeschreibung	21
Kurs DLMDWPMP01: Programmieren mit Python	23

Modul DLMCSITSDS: IT Sicherheit und Datenschutz

Modulbeschreibung	27
Kurs DLMCSITSDS01: IT Sicherheit und Datenschutz	29

Modul DLMCSA_D: Algorithmik

Modulbeschreibung	33
Kurs DLMCSA01_D: Algorithmik	35

Modul DLMMET-01: Forschungsmethodik

Modulbeschreibung	39
Kurs MMET01-01: Forschungsmethodik	41

2. Semester

Modul DLMCSSCTCS_D: Seminar: Aktuelle Themen der Informatik

Modulbeschreibung	51
Kurs DLMCSSCTCS01_D: Seminar: Aktuelle Themen der Informatik	53

Modul DLMAIEUIUX_D: UI/UX Expert

Modulbeschreibung	55
Kurs DLMAIEUIUX01_D: User Interface und Experience	57
Kurs DLMAIEUIUX02_D: Projekt: Mensch-Maschine-Interaktion	60

Modul DLMCSEBCQC: Blockchain and Quantum Computing

Modulbeschreibung	63
-------------------------	----

Kurs DLMCSEBCQC01: Blockchain	65
Kurs DLMCSEBCQC02: Quantum Computing	69
Modul DLMCSEAITSC: Advanced Cyber Security and Cryptology	
Modulbeschreibung	73
Kurs DLMCSEAITSC01: Seminar: Advanced Cyber Security	75
Kurs DLMCSEAITSC02: Cryptology	77
Modul DLMIWMB: Mobile Software Engineering	
Modulbeschreibung	81
Kurs DLMIWMB01: Mobile Software Engineering I	83
Kurs DLMIWMB02: Mobile Software Engineering II	87
Modul DLMWIWCC: Cloud Computing	
Modulbeschreibung	91
Kurs DLMWIWCC01: Einführung in Cloud Architekturen und Serverless Computing	93
Kurs DLMWIWCC02: Projekt: Cloud Computing	97
Modul DLMIMWCK: Computerkriminalität	
Modulbeschreibung	101
Kurs DLMIMWCK01: Angriffsszenarien und Vorfallreaktion	103
Kurs DLMIMWCK02: Projekt: Cyber-Forensik	107
Modul DLMINFWDA: Data Architectures	
Modulbeschreibung	111
Kurs DLMDMDWPO01: Data Warehousing, Pipelines and Orchestration	114
Kurs DLMDMMDP01: Managing Data Projects	118
Modul DLMINFWDDQ: Database Querying	
Modulbeschreibung	123
Kurs DLMDMDQL01: Data Query Languages	125
Kurs DLMDMNDB01: NoSQL Databases	128
Modul DLMCSEEDSO_D: Sichere Software-Entwicklung	
Modulbeschreibung	133
Kurs DLMCSEEDSO01_D: Sichere Software-Entwicklung	135
Kurs DLMCSEEDSO02_D: Projekt: Sichere Software-Implementierung	138
Modul DLMMAB: Masterarbeit	
Modulbeschreibung	141
Kurs DLMMAB01: Masterarbeit	143
Kurs DLMMAB02: Kolloquium	146

2023-02-15

1. Semester

IT-Systeme: Software

Modulcode: DLMIMITSS

Modultyp s. Curriculum	Zugangsvoraussetzungen keine	Niveau MA	ECTS 5	Zeitaufwand Studierende 150 h
----------------------------------	--	---------------------	------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Dr. Christian Prause (IT-Systeme: Software)

Kurse im Modul

- IT-Systeme: Software (DLMIMITSS01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Grundlagen der Softwareentwicklung
- Datenformate und Codierung
- Firmware und Betriebssysteme
- Klassifizierung und Anwendungsbereiche von Desktop-Applikationen
- Datenbanken
- Anwendungsspezifische Softwaresysteme im Unternehmen
- Ergonomische Aspekte der Computerarbeitsplatzgestaltung und der Mensch-Maschine-Interaktion

Qualifikationsziele des Moduls**IT-Systeme: Software**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Grundlagen der Softwareentwicklung zu verstehen.
- Datenformate und ihre Anwendung in unterschiedlichen Szenarien zu beurteilen.
- die Speicherung und Verarbeitung komplexer Daten und Information zu verstehen.
- Betriebssysteme und deren konzeptionelle Unterschiede für Anwendung und Sicherheit zu beurteilen.
- Einsatzgebiete typischer Desktop-Applikationen zu verstehen und deren Grenzen zu beurteilen.
- Datenbank-basierte Unternehmenslösungen zu differenzieren und deren Nutzen für unternehmerische Anwendungsbereiche zu bewerten.
- Anforderungen an Computerarbeitsplätze zu identifizieren und geeignete Lösungen zu implementieren.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung.

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme im Bereich IT & Technik.

IT-Systeme: Software

Kurscode: DLMIMITSS01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Der Kurs führt in die Funktion und die Anwendungsbereiche typischer Softwaresysteme ein, die in Unternehmen zum Einsatz kommen. Dazu bilden Konzepte der Softwareentwicklung und der Programmiersprachen die Grundlage. Der Kurs vermittelt notwendige Kenntnisse über Datenformate, deren Konversion, Komprimierung und Transformation, um diese auf die Repräsentation komplexer Daten anzuwenden. Er beschreibt Betriebssysteme für lokale und mobile Computer und deren konzeptionelle Unterschiede und Anwendungsbereiche. Darauf aufbauend werden typische Desktop-Applikationen von Text- bis zur Grafikverarbeitung vorgestellt und deren Einsatzgebiet erläutert. Nach einer Einführung in das Konzept der Datenbanken werden typische Server-basierte Lösungen für das Informationsmanagement behandelt. Der Kurs schließt mit einer Betrachtung ergonomischer Software-Aspekte und der Mensch-Maschine-Interaktion ab.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Grundlagen der Softwareentwicklung zu verstehen.
- Datenformate und ihre Anwendung in unterschiedlichen Szenarien zu beurteilen.
- die Speicherung und Verarbeitung komplexer Daten und Information zu verstehen.
- Betriebssysteme und deren konzeptionelle Unterschiede für Anwendung und Sicherheit zu beurteilen.
- Einsatzgebiete typischer Desktop-Applikationen zu verstehen und deren Grenzen zu beurteilen.
- Datenbank-basierte Unternehmenslösungen zu differenzieren und deren Nutzen für unternehmerische Anwendungsbereiche zu bewerten.
- Anforderungen an Computerarbeitsplätze zu identifizieren und geeignete Lösungen zu implementieren.

Kursinhalt

1. Grundlagen der Softwareentwicklung
 - 1.1 Grundlagen der Programmierung und Programmiersprachen
 - 1.2 Software-Lebenszyklus
 - 1.3 Software-Lizensierungsmodelle und Patentierung

2. Datenformate
 - 2.1 ASCII-Code, Unicode und Auszeichnungssprachen
 - 2.2 Seitenbeschreibungssprachen (HTML, XHTML, HTML5)
 - 2.3 Scriptsprachen für Webapplikationen
 - 2.4 Textformate
 - 2.5 Raster-, Vektor-, und Metagrafikformate (PNG, TIFF, JPEG, SVG, WMF)
3. Konversion, Komprimierung und Transformation von Daten
 - 3.1 Datenkonversion (XMI, Transcoding)
 - 3.2 Datenkomprimierung
 - 3.3 Datentransformation
 - 3.4 Anwendung auf audiovisuelle Daten
4. System-Software
 - 4.1 Firmware, BIOS, UEFI
 - 4.2 Betriebssysteme für Endanwender
 - 4.3 Serverbasierte Betriebssysteme
 - 4.4 Mobile Betriebssysteme
5. Desktop-Applikationen
 - 5.1 Office-Software
 - 5.2 Grafik- und Bildbearbeitungsprogramme
 - 5.3 Software für Mathematik und Statistik
 - 5.4 Destop-Publishing und Visualisierung
 - 5.5 Audio- und Videosysteme
6. Datenbanksysteme
 - 6.1 Relationale Datenbanken und SQL
 - 6.2 NoSQL und nicht-relationale Datenbanken
 - 6.3 In-Memory-Datenbanken
 - 6.4 Data Warehousing
7. Business-Systeme
 - 7.1 Webbasierte Systeme und Cloud-Lösungen
 - 7.2 Dokumenten- und Content-Management
 - 7.3 Ressourcenbasiertes Informationsmanagement
 - 7.4 Knowledge-Management, Dashboards und Expertensysteme

8. Ergonomie am Computerarbeitsplatz
 - 8.1 Anthropometrie und Systemergonomie
 - 8.2 Produkt- und Produktionsergonomie
 - 8.3 Computer-Arbeitsplatzergonomie
 - 8.4 Software-Ergonomie
 - 8.5 Designaspekte der grafischen Benutzerschnittstelle

Literatur

Pflichtliteratur

Weiterführende Literatur

- Biesel, H./Hame, H. (2018): Vertrieb Und Marketing in Der Digitalen Welt – So Schaffen Unternehmen Die Business Transformation in Der Praxis. Springer Gabler, Wiesbaden.
- Bourke, P./Fairley, R.E. (Hrsg.) (2014): SWEBOOK V3.0 –Guide to the Software Engineering Body of Knowledge. IEEE Computer Society.
- Chambers, J.M. (2014): Object-Oriented Programming, Functional Programming and R. Statistical Science. 29. Jg., Heft 2, S.167–180.
- Dankmeier, W. (2017): Grundkurs Codierung. Verschlüsselung, Kompression und Fehlerbeseitigung. 4. Auflage, Springer Vieweg, Wiesbaden.
- Geisler, F. (2014): Datenbanken: Grundlagen und Design. 5. Auflage, MIT Press, Heidelberg.
- Groll, T. (2015): 1x1 des Lizenzmanagements, Praxisleitfaden für Lizenzmanager. 3. Auflage, Hanser Verlag.
- Gumm, H.P./Sommer, M. (2012): Einführung in die Informatik. 10. Auflage, Oldenbourg Verlag, München.
- Meier, A. (2017): Werkzeuge der digitalen Wirtschaft: Big Data, NoSQL & Co. Springer, Wiesbaden.
- Schlick, C./Bruder, R./Luczak, H. (2018): Arbeitswissenschaft. 4. Auflage, Springer, Berlin/Heidelberg.
- Strutz, T. (2017): Bilddatenkompression. Grundlagen, Codierung, Wavelets, JPEG, MPEG, H.264, HVEC. 4. Auflage, Springer Vieweg, Wiesbaden.
- Tanenbaum, A.S./Bos, H. (2016): Moderne Betriebssysteme. 4. Auflage, Pearson Deutschland, Hallbergmoos.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

IT-Systeme: Hardware

Modulcode: DLMIMITSH

Modultyp s. Curriculum	Zugangsvoraussetzungen keine	Niveau MA	ECTS 5	Zeitaufwand Studierende 150 h
----------------------------------	--	---------------------	------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Damir Ismailovic (IT-Systeme: Hardware)

Kurse im Modul

- IT-Systeme: Hardware (DLMIMITSH01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Rechnerarithmetik
- Integrierte Schaltkreise
- Speichersysteme
- Ein-/Ausgabesysteme
- Grundlagen der Datenübertragung
- Computernetze
- Server und Rechenzentren

Qualifikationsziele des Moduls**IT-Systeme: Hardware**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Rechnerarithmetik zu verstehen und auf logische Problemstellungen anzuwenden.
- Bestandteile von Rechner-Systemen zu kennen und deren Funktionsprinzipien zu erklären.
- Methoden der Datenübertragung zu differenzieren und deren konzeptionelle Unterschiede in der Anwendung zu bewerten.
- Computernetztechnologien und deren Einsatzgebiete zu beurteilen.
- Anforderungen für den Aufbau und den Betrieb von Rechenzentren zu kennen und zu beurteilen.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung.

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme im Bereich IT & Technik.

IT-Systeme: Hardware

Kurscode: DLMIMITSH01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Der Kurs vermittelt das Verständnis der Funktionsweise computerbasierter Systeme und dient als Basis für die Kommunikation und Führung entsprechender Fachkräfte der Informationstechnologie. Er beschreibt die Logik, mit der digitale Computer arbeiten und die Technik der Herstellung digitaler Schaltkreise. Weiterhin erläutert er den Aufbau typischer Computersysteme und die Funktionsweise von Prozessoren, Speicherbausteinen und peripherer Ein- und Ausgabegeräte. Er vermittelt die Grundlagen der Nachrichtentechnik und stellt die Einsatzkriterien kabelgebundener und kabelloser Datenübertragungstechniken gegenüber. Auf dieser Grundlage werden kleine Server-Infrastrukturen, Großrechner und Supercomputer vorgestellt und Kenntnisse zum Aufbau und Betrieb von Rechenzentren vermittelt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Rechnerarithmetik zu verstehen und auf logische Problemstellungen anzuwenden.
- Bestandteile von Rechner-Systemen zu kennen und deren Funktionsprinzipien zu erklären.
- Methoden der Datenübertragung zu differenzieren und deren konzeptionelle Unterschiede in der Anwendung zu bewerten.
- Computernetztechnologien und deren Einsatzgebiete zu beurteilen.
- Anforderungen für den Aufbau und den Betrieb von Rechenzentren zu kennen und zu beurteilen.

Kursinhalt

1. Grundlagen der Rechnerarithmetik
 - 1.1 Stellenwertarithmetik, Zahlensysteme
 - 1.2 Aussagenlogik und boolesche Operatoren
 - 1.3 Rechnerarithmetik
2. Integrierte Schaltkreise
 - 2.1 Integrierte Schaltkreise und Halbleiterproduktion
 - 2.2 Parallele und serielle Schnittstellen
 - 2.3 Komponenten der Hauptplatine
 - 2.4 Prozessoren und Speicher

3. Speichersysteme
 - 3.1 Festplattenspeicher
 - 3.2 Optische Speichermedien
 - 3.3 Magnetische Wechseldatenträger
 - 3.4 Solid State Disk
4. Ein-/Ausgabesysteme
 - 4.1 Eingabegeräte
 - 4.2 Sensorbildschirmssysteme
 - 4.3 Bildausgabegeräte
 - 4.4 Druckersysteme
5. Grundlagen der Datenübertragung
 - 5.1 Kabelgebundene Datenübertragung und Modulation
 - 5.2 Übertragung mittels Licht
 - 5.3 Antennen und Satellitentechnik
 - 5.4 Mobilfunknetze
 - 5.5 RFID und Near-Field Communication
6. Computernetze
 - 6.1 Netzwerk-Topologie
 - 6.2 Ethernet-Frame und Netzwerkprotokolle
 - 6.3 Switching, Routing und Datenflusssteuerung
 - 6.4 Netzwerkd Diagnose
7. Server und Rechenzentren
 - 7.1 Multi-Tier-Architekturen
 - 7.2 Server-Systeme, Großrechner und Supercomputer
 - 7.3 Aufbau von Rechenzentren
 - 7.4 Aspekte der Sicherheit und des Betriebs von Rechenzentren
 - 7.5 Prinzipien der Virtualisierung

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Beetz, J. (2019): Digital – Wie Computer denken. Springer, Berlin.
- Dürr, B. (2018): IT-Räume und Rechenzentren planen und betreiben: Handbuch der Bautechnik und Technischen Gebäudeausrüstung. Verlag Bau+Technik, Erkrath.
- Geng, H. (2015): Data Center Handbook. Wiley, New York.
- Hoffmann, D.W. (2016): Grundlagen der Technischen Informatik. Carl Hanser Verlag, München.
- Schiffmann, W./Bähring, H./Hönig, U. (2011): Technische Informatik 3 - Grundlagen der PC-Technologie. Springer, Berlin.
- Tanenbaum, A. S./Wetherall, D. J. (2012): Computernetzwerke. Pearson, München.
- Werner, M. (2017): Nachrichtentechnik: Eine Einführung für alle Studiengänge. Springer Vieweg, Wiesbaden.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Programmieren mit Python

Modulcode: DLMDWPMP

Modultyp	Zugangsvoraussetzungen	Niveau	ECTS	Zeitaufwand Studierende
s. Curriculum	keine	MA	5	150 h

Semester	Dauer	Regulär angeboten im	Unterrichtssprache
s. Curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	Deutsch

Modulverantwortliche(r)

Prof. Dr. Kamal Bhattacharya (Programmieren mit Python)

Kurse im Modul

- Programmieren mit Python (DLMDWPMP01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Schriftliche Ausarbeitung: Hausarbeit

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Einführung in die Programmiersprache Python
- Objektorientierte Konzepte in Python
- Behandlung von Ausnahmen und Fehlern
- Das Ökosystem der Python-Bibliothek
- Umgebungen und Paketmanagement
- Dokumentation und Prüfung
- Versionskontrolle

Qualifikationsziele des Moduls**Programmieren mit Python**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die grundlegende Python-Syntax und die Programmierkonzepte zu verstehen.
- sich an objektorientierte Konzepte in Python zu erinnern.
- verschiedene Methoden zur Fehlerbehandlung in Python zu analysieren und anzuwenden.
- gängige und wichtige Python-Bibliotheken zu kennen und wissen, wie man sie auf bestimmte Programmieraufgaben anwendet.
- Konzepte wie Umgebungen und Versionskontrolle zu verstehen.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich
Data Science & Artificial Intelligence

**Bezüge zu anderen Studiengängen der IU
Internationale Hochschule (IU)**

Alle Master-Programme im Bereich IT & Technik

Programmieren mit Python

Kurscode: DLMDWPMP01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Python ist eine der vielseitigsten und am weitesten verbreiteten Skriptsprachen. Seine klare und übersichtliche Syntax sowie sein geradliniges Design tragen wesentlich zu diesem Erfolg bei und machen ihn zu einer idealen Sprache für die Programmierausbildung. Die Anwendungsgebiete reichen von der Webentwicklung bis hin zum wissenschaftlichen Rechnen. Insbesondere in den Bereichen Datenwissenschaft und künstliche Intelligenz ist sie die gebräuchlichste Programmiersprache, die von allen wichtigen Datenverarbeitungs- und Analyseframeworks unterstützt wird. Dieser Kurs bietet eine gründliche Einführung in die Sprache und ihre Hauptfunktionen sowie Einblicke in die Begründung und Anwendung wichtiger angrenzender Konzepte wie Umgebungen, Tests und Versionskontrolle.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die grundlegende Python-Syntax und die Programmierkonzepte zu verstehen.
- sich an objektorientierte Konzepte in Python zu erinnern.
- verschiedene Methoden zur Fehlerbehandlung in Python zu analysieren und anzuwenden.
- gängige und wichtige Python-Bibliotheken zu kennen und wissen, wie man sie auf bestimmte Programmieraufgaben anwendet.
- Konzepte wie Umgebungen und Versionskontrolle zu verstehen.

Kursinhalt

1. Einführung in Python
 - 1.1 Datenstrukturen
 - 1.2 Konditionaler Code
 - 1.3 Funktionen
 - 1.4 Schleifen
 - 1.5 Eingabe/Ausgabe
 - 1.6 Programme protokollieren
 - 1.7 Module und Pakete

2. Klassen und Vererbung
 - 2.1 Namespaces und Scopes
 - 2.2 Klassen und Vererbung
 - 2.3 Iteratoren und Generatoren
3. Fehler und Ausnahmen
 - 3.1 Syntaxfehler
 - 3.2 Behandlung und Auslösung von Ausnahmen
 - 3.3 Benutzerdefinierte Ausnahmen
4. Wichtige Python-Bibliotheken
 - 4.1 Standard-Python-Bibliothek
 - 4.2 Wissenschaftliche Berechnungen
 - 4.3 Beschleunigung von Python
 - 4.4 Visualisierung
 - 4.5 Zugriff auf Datenbanken
5. Arbeiten mit Python
 - 5.1 Virtuelle Umgebungen
 - 5.2 Verwaltung von Paketen
 - 5.3 Unit- und Integrationstests
 - 5.4 Dokumentation des Codes
6. Versionskontrolle
 - 6.1 Einführung in die Versionskontrolle
 - 6.2 Versionskontrolle mit GIT

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Lutz, M. (2013): Learning Python. 5. Auflage, O'Reilly, Sebastopol.
- Matthes, E. (2019): Python Crash Course. 2. Auflage, No Starch Press, San Francisco.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Hausarbeit

Zeitaufwand Studierende					
Selbststudium 110 h	Präsenzstudium 0 h	Tutorium 20 h	Selbstüberprüfung 20 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

DLMDWPMP01

IT Sicherheit und Datenschutz

Modulcode: DLMCSITSDS

Modultyp s. Curriculum	Zugangsvoraussetzungen keine	Niveau MA	ECTS 5	Zeitaufwand Studierende 150 h
----------------------------------	--	---------------------	------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Ralf Kneuper (IT Sicherheit und Datenschutz)

Kurse im Modul

- IT Sicherheit und Datenschutz (DLMCSITSDS01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Fachpräsentation

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Datenschutz und Privatsphäre
- Bausteine der IT-Sicherheit
- IT-Sicherheitsmanagement
- Kryptographiekonzepte
- Kryptographie-Anwendungen

Qualifikationsziele des Moduls**IT Sicherheit und Datenschutz**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Kernkonzepte von IT-Sicherheit, Datenschutz und Kryptographie einschließlich ihrer Unterschiede und Beziehungen zu erklären.
- die Ansätze zum Datenschutz in verschiedenen Rechtsordnungen zu vergleichen.
- Datenschutzkonzepte auf die Datenwissenschaft und andere Anwendungsszenarien anzuwenden
- eine Analyse von Anwendungsszenarien durchzuführen, um die geeigneten Maßnahmen für das IT-Sicherheitsmanagement zu identifizieren, die umgesetzt werden sollten.
- Anwendungsszenarien zu untersuchen, um die geeigneten kryptografischen Konzepte zu identifizieren.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme im Bereich IT & Technik

IT Sicherheit und Datenschutz

Kurscode: DLMCSITSDS01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Mit der zunehmenden Digitalisierung und Vernetzung von IT-Systemen ist der Bedarf gestiegen, Systeme und die von diesen Systemen verarbeiteten Daten zu schützen. Ziel dieses Moduls ist es, ein Verständnis für die erforderlichen Sicherheitsmaßnahmen, die IT-Sicherheit einschließlich Kryptographie und den Datenschutz zu vermitteln. Während der Bedarf an IT-Sicherheit weltweit ähnlich ist, haben verschiedene Kulturen unterschiedliche Erwartungen an Datenschutz und Privatsphäre. Dennoch werden personenbezogene Daten oft außerhalb des Landes verarbeitet, in dem die betroffenen Personen leben. Daher müssen die kulturellen Aspekte des Datenschutzes bei der Verarbeitung der Daten berücksichtigt werden. Dieser Kurs gibt einen Überblick über die wichtigsten IT-Sicherheitsmaßnahmen in verschiedenen Anwendungsszenarien sowie deren Integration in ein Informationssicherheitsmanagementsystem mit besonderem Fokus auf die relevante Normenfamilie ISO/IEC 270xx. Die Kryptographie stellt ein wichtiges Werkzeug für die IT-Sicherheit dar und wird in vielen verschiedenen Anwendungsszenarien wie sicheren Internetprotokollen und Block Chain eingesetzt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Kernkonzepte von IT-Sicherheit, Datenschutz und Kryptographie einschließlich ihrer Unterschiede und Beziehungen zu erklären.
- die Ansätze zum Datenschutz in verschiedenen Rechtsordnungen zu vergleichen.
- Datenschutzkonzepte auf die Datenwissenschaft und andere Anwendungsszenarien anzuwenden
- eine Analyse von Anwendungsszenarien durchzuführen, um die geeigneten Maßnahmen für das IT-Sicherheitsmanagement zu identifizieren, die umgesetzt werden sollten.
- Anwendungsszenarien zu untersuchen, um die geeigneten kryptografischen Konzepte zu identifizieren.

Kursinhalt

1. Grundlagen von Datenschutz und IT-Sicherheit
 - 1.1 Terminologie und Risikomanagement
 - 1.2 Kernkonzepte der IT-Sicherheit
 - 1.3 Kernkonzepte von Datenschutz und Privatsphäre
 - 1.4 Kernkonzepte der Kryptografie
 - 1.5 Rechtliche Aspekte

2. Datenschutz
 - 2.1 Grundbegriffe des Datenschutzes (ISO/IEC 29100, Privacy by Design)
 - 2.2 Datenschutz in Europa: die DSGVO
 - 2.3 Datenschutz in den USA
 - 2.4 Datenschutz in Asien
3. Anwendung des Datenschutzes
 - 3.1 Anonymität und Pseudonyme
 - 3.2 Datenschutz in der Datenwissenschaft und Big Data
 - 3.3 Benutzer-Tracking im Online-Marketing
 - 3.4 Cloud Computing
4. Bestandteile der IT-Sicherheit
 - 4.1 Authentifizierung, Zugriffsverwaltung und -kontrolle
 - 4.2 Endgerätesicherheit
 - 4.3 IT-Sicherheit in Netzwerken
 - 4.4 Entwicklung sicherer IT-Systeme
5. IT-Sicherheitsmanagement
 - 5.1 Sicherheitsrichtlinien
 - 5.2 Sicherheits- und Risikoanalyse
 - 5.3 Die ISO 27000-Reihe
 - 5.4 IT-Sicherheit und IT-Governance
 - 5.5 Beispiel: IT-Sicherheit für Kreditkarten (PCI DSS)
6. Kryptografie
 - 6.1 Grundbegriffe der Kryptografie
 - 6.2 Symmetrische Kryptografie
 - 6.3 Asymmetrische Kryptografie
 - 6.4 Kryptografie mit elliptischer Kurve
 - 6.5 Hash-Funktionen
 - 6.6 Sicherer Datenaustausch
7. Kryptografische Anwendung
 - 7.1 Digitale Signaturen
 - 7.2 Sichere Internet-Protokolle
 - 7.3 Blockchain
 - 7.4 Elektronisches Geld

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Bowman, C. et al. (2015): The architecture of privacy. On engineering technologies that can deliver trustworthy safeguards. O'Reilly, Sebastopol, CA.
- Hintzbergen, J. et al. (2015): Foundations of information security (3rd ed.). Van Haren Publishing, Zaltbommel.
- ISO/IEC 29100 (2011): Information technology — Security techniques — Privacy framework. ISO. (URL: https://standards.iso.org/ittf/PubliclyAvailableStandards/c045123_ISO_IEC_29100_2011.zip [Retrieved: 11.3.2020]).
- Paar, C./Pelzl, J. (2011). Understanding cryptography: A textbook for students and practitioners. Springer, Heidelberg.
- The Open Web Application Security Project (OWASP) (2005): A guide to building secure web applications and web services. OWASP. (URL: <https://www.um.es/atika/documentos/OWASPGuide2.0.1.pdf> [Retrieved: 11.3.2020]).

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Fachpräsentation

Zeitaufwand Studierende					
Selbststudium 110 h	Präsenzstudium 0 h	Tutorium 20 h	Selbstüberprüfung 20 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Algorithmik

Modulcode: DLMCSA_D

Modultyp	Zugangsvoraussetzungen	Niveau	ECTS	Zeitaufwand Studierende
s. Curriculum	Keine	MA	5	150 h

Semester	Dauer	Regulär angeboten im	Unterrichtssprache
s. Curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	Deutsch

Modulverantwortliche(r)

Prof. Dr. Paul Libbrecht (Algorithmik)

Kurse im Modul

- Algorithmik (DLMCSA01_D)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Algorithmenentwurf
- Wichtig Klassen von Algorithmen
- Korrektheit und Vollständigkeit von Algorithmen
- Berechenbarkeit und die theoretischen Grenzen von Algorithmen
- Effizienz der Algorithmen

Qualifikationsziele des Moduls**Algorithmik**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- das Konzept eines Algorithmus und seine Bedeutung in der Informatik zu erklären.
- die Korrektheit von Softwareprogrammen zu bewerten.
- die theoretischen und praktischen Grenzen von Softwareprogrammen zu diskutieren.
- Algorithmen auszuwählen, um bestimmte Anwendungsprobleme zu lösen.
- neue Algorithmen auf Basis von Standardmethoden zu entwerfen, um einfache Anwendungsprobleme zu lösen.
- Algorithmen zu analysieren und zu vergleichen sowie ihre Stärken und Schwächen zu skizzieren.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für alle weiteren Module aus dem Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme aus dem Bereich IT & Technik

Algorithmik

Kurscode: DLMCSA01_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	Keine

Beschreibung des Kurses

Eine Kernaktivität der Informatik und ähnlicher Bereiche, ist der Entwurf, die Verwendung und die Anwendung von Algorithmen zur Lösung von Problemen. Dieser Kurs führt in gängige Ansätze zum Entwurf von Algorithmen sowie in wichtige Klassen von Algorithmen ein, die zur Lösung häufiger Probleme verwendet werden können. Die erfolgreiche Durchführung dieser Aufgaben erfordert ein gründliches Verständnis der Qualitätsmerkmale von Algorithmen: (partielle und totale) Korrektheit, Genauigkeit, Vollständigkeit und Effizienz. Gleichzeitig gibt es Grenzen für das, was ein Algorithmus theoretisch und praktisch erreichen kann und sollte, und es ist wichtig, diese Grenzen zu erkennen und zu berücksichtigen. Neben Algorithmen, die auf Standardprogrammierparadigmen basieren, gibt es auch verschiedene andere Programmierparadigmen, die zu anderen Arten von Algorithmen führen; daher bietet dieser Kurs auch eine kurze Einführung in Algorithmen für paralleles Rechnen, probabilistische Algorithmen und Quantenalgorithmen.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- das Konzept eines Algorithmus und seine Bedeutung in der Informatik zu erklären.
- die Korrektheit von Softwareprogrammen zu bewerten.
- die theoretischen und praktischen Grenzen von Softwareprogrammen zu diskutieren.
- Algorithmen auszuwählen, um bestimmte Anwendungsprobleme zu lösen.
- neue Algorithmen auf Basis von Standardmethoden zu entwerfen, um einfache Anwendungsprobleme zu lösen.
- Algorithmen zu analysieren und zu vergleichen sowie ihre Stärken und Schwächen zu skizzieren.

Kursinhalt

1. Einführung in die Algorithmik
 - 1.1 Grundlegende Konzepte und historischer Überblick
 - 1.2 Algorithmen, Programmiersprachen und Datenstrukturen
 - 1.3 Qualität von Algorithmen: Korrektheit, Genauigkeit, Vollständigkeit, Effizienz
 - 1.4 Die Rolle von Algorithmen in der Gesellschaft

2. Algorithmenentwurf
 - 2.1 Datenstrukturen
 - 2.2 Rekursion und Iteration
 - 2.3 Divide-and-Conquer
 - 2.4 Algorithmenstrategien: Ausgleichendes, gieriges und dynamisches Programmieren
3. Wichtige Algorithmen
 - 3.1 Sortieren und Suchen
 - 3.2 Mustererkennung
 - 3.3 Der RSA-Algorithmus
 - 3.4 Der k-Means-Algorithmus für Daten-Clustering
4. Korrektheit, Genauigkeit und Vollständigkeit von Algorithmen
 - 4.1 Partielle Korrektheit
 - 4.2 Totale Korrektheit
 - 4.3 Sicherstellung der Korrektheit im Programmieralltag
 - 4.4 Genauigkeit, Approximation und Fehleranalyse
5. Berechenbarkeit
 - 5.1 Modelle der Datenverarbeitung
 - 5.2 Das Halteproblem
 - 5.3 Unentscheidbare Probleme
6. Effizienz von Algorithmen: Komplexitätstheorie
 - 6.1 Modelle der Komplexität
 - 6.2 NP-Vollständigkeit
 - 6.3 $P=NP?$
7. Erweiterte Algorithmik
 - 7.1 Parallele Rechnern
 - 7.2 Probabilistische Algorithmen

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C., Molitor, P. (2013). Algorithmen – Eine Einführung, Oldenbourg Verlag
- Dewdney, A. K. (1995). Der Turing Omnibus. Springer Verlag.
- Harel, D. (2014). Algorithmics: The spirit of computing (3rd ed.). Berlin, Heidelberg: Springer.
- Sedgewick, A., & Wayne, K. (2011). Algorithms (4th ed.). Boston, MA: Pearson Education.
- Skiena, S. S. (2012). The algorithm design manual (2nd ed.). London: Springer.
- Weicker, K., Weicker, (2013) N., Algorithmen und Datenstrukturen, Springer Verlag

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Forschungsmethodik

Modulcode: DLMMET-01

Modultyp s. Curriculum	Zugangsvoraussetzungen keine	Niveau MA	ECTS 5	Zeitaufwand Studierende 150 h
----------------------------------	--	---------------------	------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Julia Pitters (Forschungsmethodik)

Kurse im Modul

- Forschungsmethodik (MMET01-01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Kombistudium
Klausur, 90 Minuten

Studienformat: Fernstudium
Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Einführung in Wissenschaftstheorien
- Voraussetzungen für quantitatives Messen und Testen
- Grundlagen der qualitativen Forschung

Qualifikationsziele des Moduls

Forschungsmethodik

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- unterschiedliche Annahmen und Herangehensweisen qualitativer und quantitativer Forschung zu kategorisieren.
- die methodologischen Voraussetzungen zu bestimmen, die bei der quantitativen Messung und Testung spezifischer Konstrukte gegeben sein müssen.
- die jeweiligen quantitativen Skalen und Indikatoren zielgerichtet in eigener Forschung einzusetzen.
- verschiedene qualitative Erhebungs- und Auswertungsverfahren voneinander zu differenzieren und in eigener Forschung anzuwenden.
- spezielle Probleme bei der Durchführung von Forschungsstudien zu analysieren und kennen diesbezügliche Lösungsmöglichkeiten, um eine optimale Durchführung von Forschung realisieren zu können.
- die Qualität von Forschungsvorhaben hinsichtlich quantitativer und qualitativer Gütekriterien bewerten zu können.
- Konzeptionen der Forschung im Hinblick auf Forschungsphilosophie, Forschungsansatz und ethischen Aspekten zu bewerten.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für alle weiteren Module im Bereich Methoden

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme im Bereich Wirtschaft & Management

Forschungsmethodik

Kurscode: MMET01-01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Der Kurs vermittelt in kritischer Weise zuerst den wissenschaftstheoretischen Hintergrund und die Terminologie der entsprechenden forschungstheoretischen Paradigmen, um den Studierenden die unterschiedliche Herangehensweise qualitativer und quantitativer Methodik verständlich zu machen. Dabei werden die unterschiedlichen Perspektiven der Wissenschaftstheorie in die Betrachtung einbezogen. Aufbauend auf die Skalenniveaus, lernen die Studierenden die Annahmen der klassischen sowie der probabilistischen Testtheorie kennen, um auf deren Basis die Anforderungen an Forschungsmethoden im Sinne der Qualitätskriterien sowie die Notwendigkeit der Bildung verschiedener Skalentypen und Indikatoren nachvollziehen zu können. Die wichtigen Aspekte der Konzeption der Forschung, ausgehend von der Forschungsphilosophie bis hin zu ethischen Dimensionen der Forschung werden verknüpft mit der Betrachtung von quantitativer und qualitativer Forschung um letztendlich deren Verbindung der Triangulation aufzuzeigen. Wichtig bei den Untersuchungsdesigns ist es, deren Güte in der Umsetzung festzustellen, sodass Gütekriterien sowohl bei qualitativer als auch bei quantitativer Forschung im Fokus stehen. Den Abschluss bilden Methoden der Datengenerierung und Methoden der Datenanalyse von qualitativer Forschung. Dabei werden die bedeutsamen Methoden der Datenanalyse wie die Inhaltsanalyse, Grounded Theorie und die Diskursanalyse sowohl theoretisch als auch praxisorientiert näher gebracht und den Studierenden die Möglichkeit eingeräumt, besondere Interviewformen – wie das fokussierte Interview oder das narrative Interview – neben der theoretischen Beschäftigung auch in der konkreten Umsetzung wahrzunehmen, aber auch Beobachtung und Feldnotizen zu betrachten.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- unterschiedliche Annahmen und Herangehensweisen qualitativer und quantitativer Forschung zu kategorisieren.
- die methodologischen Voraussetzungen zu bestimmen, die bei der quantitativen Messung und Testung spezifischer Konstrukte gegeben sein müssen.
- die jeweiligen quantitativen Skalen und Indikatoren zielgerichtet in eigener Forschung einzusetzen.
- verschiedene qualitative Erhebungs- und Auswertungsverfahren voneinander zu differenzieren und in eigener Forschung anzuwenden.
- spezielle Probleme bei der Durchführung von Forschungsstudien zu analysieren und kennen diesbezügliche Lösungsmöglichkeiten, um eine optimale Durchführung von Forschung realisieren zu können.
- die Qualität von Forschungsvorhaben hinsichtlich quantitativer und qualitativer Gütekriterien bewerten zu können.
- Konzeptionen der Forschung im Hinblick auf Forschungsphilosophie, Forschungsansatz und ethischen Aspekten zu bewerten.

Kursinhalt

1. Wissenschaftliche Grundlagen
 - 1.1 Grundlegende Vorstellungen in der Wissenschaft
 - 1.2 Von der Idee zum Forschungsvorhaben
 - 1.3 Erklärungsansätze in der Wissenschaft
2. Perspektiven in der Wissenschaftstheorie
 - 2.1 Vom logischen Empirismus zum kritischen Rationalismus
 - 2.2 Konstruktivismus
 - 2.3 Methodischer Anarchismus
3. Quantitatives Messen mit der klassischen und probabilistischen Testtheorie
 - 3.1 Skalenniveaus und die Unterscheidung manifester und latenter Merkmale
 - 3.2 Klassische Testtheorie
 - 3.3 Probabilistische Testtheorie
4. Grundlegende Konzepte der Itembildung
 - 4.1 Skalierungsverfahren
 - 4.2 Indexbildung
5. Konzeption der Forschung
 - 5.1 Wissenschaftstheorie und Forschungsprozess
 - 5.2 Ethische Aspekte der Forschung – Forschungsethik

6. Untersuchungsdesign
 - 6.1 Der qualitative und der quantitative Ansatz
 - 6.2 Die Dichotomie von „quantitativ versus qualitativ“ – eine Begriffsbestimmung
7. Prüfung der Gütekriterien in der quantitativen und qualitativen Forschung
 - 7.1 Das Gütekriterium Objektivität
 - 7.2 Das Gütekriterium Reliabilität
 - 7.3 Das Gütekriterium Validität
8. Durchführen qualitativer Forschung
 - 8.1 Methoden der Datengenerierung
 - 8.2 Besondere Interviewformen
9. Methoden der qualitativen Analyse
 - 9.1 Inhaltsanalyse
 - 9.2 Grounded Theory
 - 9.3 Diskursanalyse

Literatur

Pflichtliteratur

Weiterführende Literatur

- Bortz, J./Döring, N. (2006): Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler. 4. Auflage, Springer, Heidelberg.
- Diekmann, A. (2007): Empirische Sozialforschung. Grundlagen, Methoden, Anwendungen. 4. Auflage, Rowohlt, Reinbek.
- Kromrey, H. (2009): Empirische Sozialforschung. 12. Auflage, UTB, Stuttgart.
- Lamnek, S. (2010): Qualitative Sozialforschung. 5. Auflage, Beltz, Weinheim.
- Mayring, P. (2002): Einführung in die Qualitative Sozialforschung. 5. Auflage, Beltz, Weinheim.
- Mayring, P. (2010): Qualitative Inhaltsanalyse. Grundlagen und Techniken. 11. Auflage, Beltz, Weinheim.
- Schnell, R./Hill, P. B./Esser, E. (2008): Methoden der empirischen Sozialforschung. 8. Auflage, Oldenbourg, München.
- Sedlmeier, P./Renkewitz, F. (2007): Forschungsmethoden und Statistik in der Psychologie. Pearson Studium, München.

Studienformat Kombistudium

Studienform Kombistudium	Kursart Vorlesung
------------------------------------	-----------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input checked="" type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input checked="" type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

MMET01-01

2. Semester

Seminar: Aktuelle Themen der Informatik

Modulcode: DLMCSSCTCS_D

Modultyp s. Curriculum	Zugangsvoraussetzungen Keine	Niveau MA	ECTS 5	Zeitaufwand Studierende 150 h
----------------------------------	--	---------------------	------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. André Köhler (Seminar: Aktuelle Themen der Informatik)

Kurse im Modul

- Seminar: Aktuelle Themen der Informatik (DLMCSSCTCS01_D)

Art der Prüfung(en)

Modulprüfung Studienformat: Fernstudium Schriftliche Ausarbeitung: Seminararbeit	Teilmodulprüfung
---	-------------------------

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

Dieses Seminar beschäftigt sich mit aktuellen Themen der Informatik. Die Studierenden tauchen tief in ein bestimmtes Thema innerhalb eines Teilbereiches ihrer Wahl ein. Mögliche Bereiche sind unter anderem: Künstliche Intelligenz, Big-Data-Technologien, IT-Sicherheit und Datenschutz, Algorithmen, Data Science und Programmierung.

Qualifikationsziele des Moduls**Seminar: Aktuelle Themen der Informatik**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- aktuelle und relevante Themen aus den Teilbereichen der Informatik zu nennen.
- Beispiele für den Einfluss der Informatik auf aktuelle Entwicklungen zu nennen.
- das theoretische Wissen auf reale Fallbeispiele zu übertragen.
- die gelernten Theorien in die praktische Anwendung im Bereich der Informatik umzusetzen.
- über ein ausgewähltes Informatik-Thema wissenschaftlich zu schreiben.
- aktuelle Probleme der Informatik kritisch zu hinterfragen und zu diskutieren.
- Modelle und Frameworks aus Teildisziplinen der Informatik zur Lösung von praktischen Problemen anzupassen.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für alle weiteren Module aus dem Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme aus dem Bereich IT & Technik

Seminar: Aktuelle Themen der Informatik

Kurscode: DLMCSSCTCS01_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	Keine

Beschreibung des Kurses

Dieses Seminar ist eine Gelegenheit für die Studierenden, das breite Wissen zu vertiefen, das sie in den vorangegangenen Semestern erworben haben. Die Studierenden wählen aus einem Teilbereich der Informatik ein Thema, welches sie speziell interessiert. Interessiert sich ein Student zum Beispiel für die Anwendung von künstlicher Intelligenz in einem bestimmten Kontext, kann die Ausarbeitung kontextspezifischer Anwendungsfälle aus einer Literaturübersicht das Thema der schriftlichen Ausarbeitung sein. Das Feedback des Betreuers hilft den Studierenden, eventuelle Schwächen im wissenschaftlichen Schreiben und akademischen Arbeiten zu beheben und bereitet sie auf das Schreiben der Masterarbeit vor.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- aktuelle und relevante Themen aus den Teilbereichen der Informatik zu nennen.
- Beispiele für den Einfluss der Informatik auf aktuelle Entwicklungen zu nennen.
- das theoretische Wissen auf reale Fallbeispiele zu übertragen.
- die gelernten Theorien in die praktische Anwendung im Bereich der Informatik umzusetzen.
- über ein ausgewähltes Informatik-Thema wissenschaftlich zu schreiben.
- aktuelle Probleme der Informatik kritisch zu hinterfragen und zu diskutieren.
- Modelle und Frameworks aus Teildisziplinen der Informatik zur Lösung von praktischen Problemen anzupassen.

Kursinhalt

Literatur

Pflichtliteratur

Weiterführende Literatur

- Eckert, C. (2018). IT-Sicherheit: Konzepte - Verfahren – Protokolle. 10. Aufl. De Gruyter
- Sedgewick, R., Wayne, K. (2011). Algorithms. 4th Ed. Addison Wesley
- Russell, S., Norvig, P. (2012). Künstliche Intelligenz: Ein moderner Ansatz. 4. Aufl. Pearson Studium
- Kleppmann, M., Langenau, F. (2018). Datenintensive Anwendungen designen. O'Reilly
- Sommerville, I. (2020). Modernes Software-Engineering. Pearson Studium

Studienformat Fernstudium

Studienform Fernstudium	Kursart Seminar
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Seminararbeit

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed

UI/UX Expert

Modulcode: DLMAIEUIUX_D

Modultyp s. Curriculum	Zugangsvoraussetzungen <ul style="list-style-type: none"> ▪ keine ▪ DLMAIEUIUX01_D oder DLMAIEUIUX01 	Niveau MA	ECTS 10	Zeitaufwand Studierende 300 h
----------------------------------	---	---------------------	-------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Adelka Niels (User Interface und Experience) / Prof. Dr. Inga Schlömer (Projekt: Mensch-Maschine-Interaktion)

Kurse im Modul

- User Interface und Experience (DLMAIEUIUX01_D)
- Projekt: Mensch-Maschine-Interaktion (DLMAIEUIUX02_D)

Art der Prüfung(en)

Modulprüfung

Teilmodulprüfung

User Interface und Experience

- Studienformat "Fernstudium": Klausur,
90 Minuten

Projekt: Mensch-Maschine-Interaktion

- Studienformat "Fernstudium": Portfolio

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

User Interface und Experience

- ROI des UX-Designs
- Rolle und Denkweise des UX-Designs in IT-Projekten
- Der UX-Designprozess
- UX-Psychologie: Wie der menschliche Verstand funktioniert
- Benutzerforschung
- UX-Design-Grundlagen

Projekt: Mensch-Maschine-Interaktion

In diesem Kurs sammeln die Studierenden praktische Erfahrungen im User Experience Design. Sie führen Usability-Tests für eine vorgegebene Benutzeroberfläche durch und arbeiten daran Verbesserungen zu entwickeln.

Qualifikationsziele des Moduls

User Interface und Experience

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- zu verstehen, worum es bei Design geht und welche Aspekte für einen gutes Design entscheidend sind.
- die Rolle des UI/UX-Designers innerhalb eines Projekts zu verstehen und zu definieren.
- den UX-Designprozess und die benutzerzentrierte Denkweise zu erklären.
- den Nutzen des UX-Designs für IT-Projekte überzeugend zu vermitteln.
- die grundlegenden Methoden der Benutzerforschung, von Benutzer-Tests und des benutzerzentrierten Designs zu beschreiben.

Projekt: Mensch-Maschine-Interaktion

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Gebrauchstauglichkeit einer Benutzeroberfläche zu bewerten.
- Usability-Tests durchzuführen.
- die Auswirkungen von „User First“ auf die Usability Praxis zu verstehen.
- kleine Änderungen an bestehenden Benutzeroberflächen vornehmen zu können und Situationen zu erkennen, in denen ein User Experience Designer hinzugezogen werden sollte.

Bezüge zu anderen Modulen im Studiengang

Baut auf Modulen aus dem Bereich Data Science & Artificial Intelligence auf

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme aus dem Bereich IT & Technik

User Interface und Experience

Kurscode: DLMAIEUIUX01_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Das UX-Design ist entscheidend für die Entwicklung neuer IT-Dienste und Anwendungen und steigert die Qualität des Ergebnisses. Die Anwendung von UX-Designtechniken kann den Software-Entwicklungsprozess signifikant und positiv verändern. Gutes UX-Design ist das Ergebnis effektiver Teamarbeit. Mit den Inhalten dieses Kurses sollen die Studierenden die Denkweise, die grundlegenden Techniken und die Auswirkungen des UX-Designs auf IT-Projekte verstehen. Sie werden lernen, wie der UX-Designprozess funktioniert und welche Rolle UX-Designer innerhalb von IT-Projekten innehaben. Sie werden auch lernen, durch welche Art der Zusammenarbeit man die besten Ergebnisse erzielt. Anhand ihrer Grundkenntnisse über gutes Design werden die Studierenden erkennen, wann es angebracht ist, kleine Änderungen an UIs selbst vorzunehmen und wann es an der Zeit ist, einen Designer zu konsultieren.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- zu verstehen, worum es bei Design geht und welche Aspekte für einen gutes Design entscheidend sind.
- die Rolle des UI/UX-Designers innerhalb eines Projekts zu verstehen und zu definieren.
- den UX-Designprozess und die benutzerzentrierte Denkweise zu erklären.
- den Nutzen des UX-Designs für IT-Projekte überzeugend zu vermitteln.
- die grundlegenden Methoden der Benutzerforschung, von Benutzer-Tests und des benutzerzentrierten Designs zu beschreiben.

Kursinhalt

1. ROI des UX-Designs
 - 1.1 Effektivität
 - 1.2 Effizienz
 - 1.3 Der Einfluss des Designs auf Anwendungsfehler
2. Rolle und Denkweise des UX-Designs in IT-Projekten
 - 2.1 Die Rolle des UX-Designs: der UX-Designer
 - 2.2 Die UX-Denkweise: der Benutzer steht an erster Stelle

3. Menschenzentrierte Gestaltung im Zusammenspiel mit Vorgehensmodellen der Softwareentwicklung
 - 3.1 Menschenzentrierte Gestaltung in einer Wasserfall-Prozessumgebung
 - 3.2 Menschenzentrierte Gestaltung in einer agilen Prozessumgebung
4. UX Psychologie: Wie der menschliche Verstand funktioniert
 - 4.1 Wahrnehmungspsychologie
 - 4.2 Informationsverarbeitung
 - 4.3 Entscheidungsfindung
 - 4.4 Situationsbewusstsein
 - 4.5 Fehler
5. Nutzerforschung
 - 5.1 Der Wert von Nutzerforschung
 - 5.2 Befragungen
 - 5.3 Beobachtungen
 - 5.4 Prototyping und Usability Testing
 - 5.5 Web-Analytics und A/B-Tests
6. UX-Design-Grundlagen
 - 6.1 Allgemeine Regeln für gute Gestaltung
 - 6.2 Design Patterns und Guidelines
 - 6.3 Barrierefreiheit
 - 6.4 Informationsarchitektur

Literatur

Pflichtliteratur

Weiterführende Literatur

- Cooper, A., Reimann, R., Cronin, D., & Noessel, C. (2014). About face: The essentials of interaction design. New York, NY: Wiley.
- Johnson, J. (2010). Designing with the mind in mind. Burlington, MA: Elsevier.
- Preece, J., Sharp, H., & Rogers, Y. (2015). Interaction design: Beyond human-computer interaction. New York, NY: Wiley.
- Microsoft Windows Dev Center. (2018). Guidelines. [Web page]. Retrieved from <https://docs.microsoft.com/en-us/windows/desktop/uxguide/guidelines>
- Unger, R., & Chandler, C. (2012). A project guide to UX design. Berkeley, CA: New Riders.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Projekt: Mensch-Maschine-Interaktion

Kurscode: DLMAIEUIUX02_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	DLMAIEUIUX01_D oder DLMAIEUIUX01

Beschreibung des Kurses

In diesem Kurs sammeln die Studierenden praktische Erfahrungen im User Experience Design. Sie führen Usability-Tests für eine vorgegebene Benutzeroberfläche durch und arbeiten daran Verbesserungen zu entwickeln.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Gebrauchstauglichkeit einer Benutzeroberfläche zu bewerten.
- Usability-Tests durchzuführen.
- die Auswirkungen von „User First“ auf die Usability Praxis zu verstehen.
- kleine Änderungen an bestehenden Benutzeroberflächen vornehmen zu können und Situationen zu erkennen, in denen ein User Experience Designer hinzugezogen werden sollte.

Kursinhalt

- User Experience Design fokussiert sich auf die Bedürfnisse der Nutzer. Im Rahmen dieses Portfolio-Projekts setzen die Studierenden grundlegende Techniken um, die zu einem guten nutzerzentrierten Design führen. Sie lernen, wie man die User Experience und Usability einer Anwendung durch Benutzertests testet, und sie lernen auch, wie man Ideen für Verbesserungen entwickelt und testet. Die Studierenden werden diesen Kurs abschließen, nachdem sie praktische Erfahrung im Umgang mit der Denkweise, den Benutzer an die erste Stelle zu setzen, gesammelt haben.

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Barnum, C. M. (2020). Usability Testing Essentials. Ready, Set...Test! 2. Auflage, Morgan Kaufmann.
- Cooper, A., Reimann, R., Cronin, D (2010). Interface und Interaction Design. Zielgerichtetes Design. User verstehen und kategorisieren. Visuelles Interface und Information Design. MITP-Verlag
- Geis, T. & Tesch, G. (2019). Basiswissen Usability und User Experience. dpunkt.verlag.
- Krug, S. (2014). Don't make me think! – Web & Mobile Usability: Das intuitive Web. 3. Auflage, mitp Verlag.
- Rubin, J. & Chisnell, D. (2011). Handbook of Usability Testing. How to Plan, Design, and Conduct Effective Tests. Indianapolis, Wiley Publishing.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Projekt
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Portfolio

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed

Blockchain and Quantum Computing

Module Code: DLMCSEBCQC

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	None	MA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Ralf Kneuper (Blockchain) / Dr. Carsten Blank (Quantum Computing)

Contributing Courses to Module

- Blockchain (DLMCSEBCQC01)
- Quantum Computing (DLMCSEBCQC02)

Module Exam Type

Module Exam

Split Exam

Blockchain

- Study Format "Distance Learning": Written Assessment: Written Assignment

Quantum Computing

- Study Format "Distance Learning": Oral Assignment

Weight of Module

see curriculum

Module Contents

Blockchain

- Basic concepts of blockchain and related technologies
- Applications of blockchain and DLT
- Security
- Development of blockchain and DLT applications
- Social and legal aspects

Quantum Computing

- Physics of quantum computing
- Quantum computing models
- Quantum algorithms
- Quantum computing with the IBM framework Qiskit
- Applications, potential for and challenges of quantum computing

Learning Outcomes

Blockchain

On successful completion, students will be able to

- outline the functions provided by and the technology used in blockchains.
- explain important applications of block chains, in particular BitCoin.
- demonstrate the technical architecture of blockchain applications.
- appraise the benefits and challenges of suggested blockchain applications.
- discuss the social and legal aspects of blockchain technology.

Quantum Computing

On successful completion, students will be able to

- outline the basic concepts of quantum mechanics as they relate to quantum computing.
- describe the computation models used in quantum computing.
- demonstrate the role of quantum computing for cryptography and other application areas.
- compare the theoretical and practical potential of quantum computing to classical computing.
- apply the concepts of quantum computing to develop simple programs within the Qiskit framework.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development.

Links to other Study Programs of IU International University of Applied Sciences (IU)

All Bachelor Programmes in the IT & Technology field.

Blockchain

Course Code: DLMCSEBCQC01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	None

Course Description

Started by the cryptocurrency BitCoin, blockchain and related topics such as distributed ledger technologies and smart contracts have become increasingly important over the last few years and are claimed to be a major disruptive technologies. As BitCoin shows, systems that today need a trustworthy central coordinating body may become genuinely distributed systems without the need for such a body in the future. While blockchain has the potential for completely new types of applications, these suggested applications do not always make use of the strengths of the technology; rather, they simply provide a different approach to solving problems that could be solved more easily and efficiently using standard technologies such as database systems. Furthermore, blockchain applications have led to new social challenges and legal questions, such as the legal status of “smart contracts”. Different infrastructures such as Ethereum and Hyperledger have been developed to form the basis for blockchain applications. The goal of this course is to provide an understanding of the technical, as well as social and legal, aspects of blockchain and related technologies.

Course Outcomes

On successful completion, students will be able to

- outline the functions provided by and the technology used in blockchains.
- explain important applications of block chains, in particular BitCoin.
- demonstrate the technical architecture of blockchain applications.
- appraise the benefits and challenges of suggested blockchain applications.
- discuss the social and legal aspects of blockchain technology.

Contents

1. Basic Concepts
 - 1.1 The Functional View: Distributed Ledger Technologies
 - 1.2 The Technical View: Blockchain
 - 1.3 History of Blockchain and DLT
 - 1.4 Consense Mechanisms

2. BitCoin
 - 2.1 The BitCoin Payment System
 - 2.2 The Technology Behind BitCoin
 - 2.3 Security of BitCoin
 - 2.4 Scalability and Other Limitations of BitCoin
 - 2.5 BitCoin Derivatives and Alternatives
3. Smart Contracts and Decentralized Apps
 - 3.1 Smart Contracts
 - 3.2 Decentralized Apps (DApps)
 - 3.3 Ethereum
 - 3.4 Hyperledger
 - 3.5 Alternative Platforms for Smart Contracts and DApps
4. Security of Block Chain and DLT
 - 4.1 Cryptology Used
 - 4.2 Attacks on Blockchain and DLT
 - 4.3 Resolving Bugs and Security Holes
 - 4.4 Long-Term Security
5. Block Chain and DLT Application Scenarios
 - 5.1 Benefits and Limits of Applying Blockchain and DLT
 - 5.2 Registers for Land and Other Property
 - 5.3 Applications in the Supply Chain
 - 5.4 Applications in Insurance
 - 5.5 Initial Coin Offerings for Sourcing Capital
 - 5.6 Examples of Further Applications
6. Development of Blockchain and DLT Applications
 - 6.1 Architecture of Blockchain and DLT Applications
 - 6.2 Platform Selection
 - 6.3 Design of Blockchain and DLT Applications
7. Blockchain and Society
 - 7.1 (Mis-)Trust in Institutions
 - 7.2 Blockchain and the Environment
 - 7.3 Cyber-Currencies in the Darknet
 - 7.4 ICO Fraud

8. Legal Aspects
 - 8.1 DLT and Smart Contracts as Legal Contracts
 - 8.2 Cryptocurrencies as Legal Currencies
 - 8.3 Regulation of ICOs
 - 8.4 Data Protection / Privacy in Blockchains

Literature

Compulsory Reading

Further Reading

- De Filippi, P., & Wright, A. (2018). Blockchain and the law. The rule of code. Cambridge, MA: Harvard University Press.
- Meinel, C., Gayvoronskaya, T. & Schnjakin, M. (2018). Blockchain. Hype or innovation. Potsdam: Universitätsverlag Potsdam.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system [white paper]. Retrieved from <https://bitcoin.org/bitcoin.pdf>
- Tapscott, D., & Tapscott, N. (2018). Blockchain revolution. How the technology behind bitcoin is changing money, business, and the world. New York, NY: Portfolio/Penguin.
- Xu, W., Weber, I., & Staples, M. (2019). Architecture for blockchain applications. Cham: Springer.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Written Assessment: Written Assignment

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

Quantum Computing

Course Code: DLMCSEBCQC02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	none

Course Description

Quantum computing is a completely new paradigm for the architecture of computers. It currently is in the early stage of development but has the potential to speed up certain kinds of computations, not just by orders of magnitude but by moving them from exponential to linear growth. One of the issues that will be affected is the prime factorization of large numbers which currently forms the basis for important cryptographic algorithms, in particular the RSA algorithm which would in that case would no longer be secure. This course gives an introduction to the physics behind quantum computing and the computation models used. Students are familiarized with the most important algorithms for quantum computing and write a few programs for quantum computers. The application potential and challenges of quantum computing are also discussed.

Course Outcomes

On successful completion, students will be able to

- outline the basic concepts of quantum mechanics as they relate to quantum computing.
- describe the computation models used in quantum computing.
- demonstrate the role of quantum computing for cryptography and other application areas.
- compare the theoretical and practical potential of quantum computing to classical computing.
- apply the concepts of quantum computing to develop simple programs within the Qiskit framework.

Contents

1. Basic concepts
 - 1.1 Quantum physics as a basis for computing
 - 1.2 Types of quantum computers
 - 1.3 Qbits
 - 1.4 Linear algebra

2. The physics of quantum computers
 - 2.1 Basic concepts of quantum mechanics
 - 2.2 Spin and entanglement
 - 2.3 Architecture of quantum computers
 - 2.4 Noise and error correction
 - 2.5 Current state and outlook
3. Quantum computing models
 - 3.1 Quantum gates and circuits
 - 3.2 Single qubit quantum systems
 - 3.3 Multiple qubit quantum systems
4. Quantum algorithms
 - 4.1 Computability and complexity in quantum computing
 - 4.2 Quantum Fourier transform
 - 4.3 The Shor algorithm
 - 4.4 The Grover algorithm
5. Quantum computing with the IBM framework Qiskit
 - 5.1 Overview of Qiskit and the IBM Q Provider
 - 5.2 Quantum circuits in Qiskit
 - 5.3 First steps in programming with Qiskit
6. Applications, potential and challenges of quantum computing
 - 6.1 Applications of quantum computing
 - 6.2 Quantum cryptography and post-quantum cryptography
 - 6.3 Quantum supremacy

Literature**Compulsory Reading****Further Reading**

- Mermin, N. D. (2007). Quantum computer science: An introduction. Cambridge University Press.
- Nielsen, M. A., & Chuang, I. L. (2000). Quantum computation and quantum information. Cambridge University Press.
- Rieffel, E. G., & Polak, W. H. (2011). Quantum computing: A gentle introduction. MIT Press.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Oral Assignment

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

DLMCSEBCQC02

Advanced Cyber Security and Cryptology

Module Code: DLMCSEAITSC

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	<ul style="list-style-type: none"> ▪ DLMCSITSDP01 or DLMCSITSDS01 ▪ DLMCSEAITSC01; DLMCSITSDP01 or DLMCSITSDS01 	MA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Alexander Lawall (Seminar: Advanced Cyber Security) / Dr. Ralf Kueres (Cryptology)

Contributing Courses to Module

- Seminar: Advanced Cyber Security (DLMCSEAITSC01)
- Cryptology (DLMCSEAITSC02)

Module Exam Type

Module Exam

Split Exam

Seminar: Advanced Cyber Security

- Study Format "Distance Learning": Written Assessment: Research Essay

Cryptology

- Study Format "Distance Learning": Oral Assignment

Weight of Module

see curriculum

Module Contents

Seminar: Advanced Cyber Security

- This course covers selected advanced topics in cyber security, including the closely related topics of data protection and cryptology, and discusses them in detail. Based on a list of topics updated regularly, students select or are assigned a specific topic about which they write a scientific research essay.

Cryptology

- Symmetric and asymmetric cryptosystems
- Authentication
- Cryptanalysis
- Cryptology in the internet
- Applications

Learning Outcomes

Seminar: Advanced Cyber Security

On successful completion, students will be able to

- analyze and describe one aspect of cyber security in detail.
- independently analyze selected topics in cyber security and link them with well-known concepts, as well as critically question and discuss them.
- transfer theoretically-acquired knowledge to a specific context.
- write and edit a scientific essay on a relevant select topic.

Cryptology

On successful completion, students will be able to

- discuss the main cryptographic systems and algorithms and their relevance in IT today.
- discuss the security of internet-based applications.
- evaluate different cryptographic systems and algorithms to select an appropriate solution for real-world problems in IT.
- apply standard cryptographic systems and algorithms to solve real-world problems in IT.
- appraise existing cryptographic solutions to real-world problems and identify major weaknesses where relevant.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development

Links to other Study Programs of IU International University of Applied Sciences (IU)

All Master Programmes in the IT & Technology field

Seminar: Advanced Cyber Security

Course Code: DLMCSEAITSC01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	DLMCSITSDP01 or DLMCSITSDS01

Course Description

This seminar covers advanced topics in cyber security. With the growth of the internet and digitization, cyber security has become an increasingly important topic and needs to be taken into account in the development and setup of software and IT systems. Typical topics that may be addressed include the analysis of selected aspects of information security management systems according to the ISO 27000 series; the use of cyber security to support data protection; and the detailed analysis and description of certain algorithms or cryptosystems.

Course Outcomes

On successful completion, students will be able to

- analyze and describe one aspect of cyber security in detail.
- independently analyze selected topics in cyber security and link them with well-known concepts, as well as critically question and discuss them.
- transfer theoretically-acquired knowledge to a specific context.
- write and edit a scientific essay on a relevant select topic.

Contents

- The seminar covers different advanced topics regarding cyber security. Each participant must prepare a research essay on a topic assigned to him/her.

Literature

Compulsory Reading

Further Reading

- Turabian, K. L. (2013). A manual for writers of research papers, theses, and dissertations. Chicago: University of Chicago Press.
- Swales, J. M., & Feak, C. R. (2012). Academic writing for graduate students, essential tasks and skills. Michigan: University of Michigan Press.
- Bailey, S. (2011). Academic writing for international students of business. New York, NY: Routledge.

Study Format Distance Learning

Study Format Distance Learning	Course Type Seminar
--	-------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Research Essay

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed

Cryptology

Course Code: DLMCSEAITSC02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	DLMCSEAITSC01; DLMCSITSDP01 or DLMCSITSDS01

Course Description

The focus of this course is to provide a thorough introduction to cryptology and its main sub-disciplines cryptography and cryptanalysis. Particular emphasis is put on the use of cryptology to support the security of IT systems. In the first part of the courses, students gain a solid understanding of the basic concepts of cryptology, in particular symmetric and asymmetric cryptosystems, authentication, and common approaches to break these cryptosystems using cryptanalysis. Based on this foundational understanding, the course goes on to cover the practical use of cryptology, starting with an introduction to the standard protocols and techniques used to ensure the security of communication via the internet. Next, practical aspects of applying cryptographic techniques and algorithms are covered, such as their long-term security. Finally, some application examples show how the concepts of cryptology are commonly used and can be used to solve challenges such as online banking.

Course Outcomes

On successful completion, students will be able to

- discuss the main cryptographic systems and algorithms and their relevance in IT today.
- discuss the security of internet-based applications.
- evaluate different cryptographic systems and algorithms to select an appropriate solution for real-world problems in IT.
- apply standard cryptographic systems and algorithms to solve real-world problems in IT.
- appraise existing cryptographic solutions to real-world problems and identify major weaknesses where relevant.

Contents

1. Basic concepts of cryptology
 - 1.1 Introduction and terminology
 - 1.2 IT security, threats and common attacks
 - 1.3 Historical overview
 - 1.4 Kerckhoffs's principle

2. Symmetric cryptosystems
 - 2.1 Substitution and transposition
 - 2.2 Stream and block ciphers
 - 2.3 Digital encryption standard (DES)
 - 2.4 Advanced encryption standard (AES)
3. Asymmetric cryptosystems
 - 3.1 The RSA algorithm
 - 3.2 Elliptic curves
 - 3.3 Cryptographic hash functions
 - 3.4 Signatures and MACs
 - 3.5 Key exchange and public key infrastructures
4. Authentication
 - 4.1 Passwords
 - 4.2 Challenge-response and zero-knowledge
 - 4.3 Biometrics-based authentication
 - 4.4 Authentication in distributed systems
 - 4.5 Smartcards
 - 4.6 Identity and anonymity
5. Cryptanalysis – how to break encryption
 - 5.1 Frequency analysis
 - 5.2 Brute-force attacks
 - 5.3 Rainbow tables
 - 5.4 Known/chosen plaintext
 - 5.5 Side-channel attacks
6. Cryptology and the internet
 - 6.1 Basic setup of the Internet and its protocols
 - 6.2 IPsec
 - 6.3 Transport Layer Security
 - 6.4 Secure E-Mail (TLS, S/MIME and PGP)
 - 6.5 Secure DNS

7. Practical aspects of cryptology
 - 7.1 Random number generation
 - 7.2 Long-term security (key lengths, perfect forward security, quantum computing)
 - 7.3 Incorporating cryptography into application development
 - 7.4 Legal and regulatory aspects

8. Applications
 - 8.1 Online banking
 - 8.2 Blockchain
 - 8.3 Voting
 - 8.4 Steganography and watermarks
 - 8.5 The Tor Project

Literature**Compulsory Reading****Further Reading**

- Esslinger, B. (2010). The CrypTool script: Cryptography, mathematics, and more (10th ed.). CrypTool Development Team.
- Katz, J., & Lindell, Y. (2014). Introduction to modern cryptography (2nd ed.). Chapman and Hall/CRC.
- Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (2015). Handbook of applied cryptography. CRC Press.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Oral Assignment

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

Mobile Software Engineering

Modulcode: DLMIWMB

Modultyp s. Curriculum	Zugangsvoraussetzungen keine	Niveau MA	ECTS 10	Zeitaufwand Studierende 300 h
----------------------------------	--	---------------------	-------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Marian Benner-Wickner (Mobile Software Engineering I) / Prof. Dr. Marian Benner-Wickner (Mobile Software Engineering II)

Kurse im Modul

- Mobile Software Engineering I (DLMIWMB01)
- Mobile Software Engineering II (DLMIWMB02)

Art der Prüfung(en)

Modulprüfung	Teilmodulprüfung
	<p><u>Mobile Software Engineering I</u></p> <ul style="list-style-type: none"> • Studienformat "Fernstudium": Schriftliche Ausarbeitung: Fallstudie <p><u>Mobile Software Engineering II</u></p> <ul style="list-style-type: none"> • Studienformat "Fernstudium": Schriftliche Ausarbeitung: Projektbericht

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

Mobile Software Engineering I

- Grundlagen der mobilen Software-Entwicklung
- Android-Systemarchitektur
- Entwicklungsumgebung
- Kernkomponenten einer Android-App
- Interaktion zwischen Anwendungskomponenten
- Fortgeschrittene Techniken

Mobile Software Engineering II

Konzeption, Umsetzung und Dokumentation von kleinen, mobilen Anwendungen auf Basis einer konkreten Aufgabenstellung.

Qualifikationsziele des Moduls

Mobile Software Engineering I

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Unterschiede und Besonderheiten der SW-Entwicklung für mobile Systeme zu erkennen und diese zu erläutern.
- verschiedene Aktivitäten, Rollen und Risiken bei Erstellung, Betrieb und Wartung von mobilen Software-Systemen zu unterscheiden.
- Architektur und technische Eigenschaften der Android Plattform zu erläutern und zu unterscheiden.
- selbständig mobile Software-Systeme zur Lösung von konkreten Problemen für die Plattform „Android“ zu erstellen.

Mobile Software Engineering II

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- selbständig eine kleine mobile Anwendung zu konzipieren und prototypisch zu erstellen, um eine gezielte Aufgabe zu lösen.
- typische Probleme und Herausforderungen in der praktischen Umsetzung kleiner mobiler Anwendungen zu erkennen.
- die Konzeption und die Umsetzung von kleinen, eigenständig konzipierten und umgesetzten mobilen Anwendungen zu dokumentieren.

Bezüge zu anderen Modulen im Studiengang

Baut auf Modulen aus dem Bereich Informatik & Software-Entwicklung auf

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme im Bereich IT & Technik

Mobile Software Engineering I

Kurscode: DLMIWMB01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Am Beispiel der mobilen Plattform „Android“ wird vermittelt, wie sich die Programmierung von mobilen Anwendungen (Apps) von der Entwicklung von Browser-basierten Informationssystemen unterscheidet, welche Technologien und Programmierkonzepte typischerweise dabei zum Einsatz kommen und welche typischen Herausforderungen es bei der App-Entwicklung für industrielle Anwendungen gibt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Unterschiede und Besonderheiten der SW-Entwicklung für mobile Systeme zu erkennen und diese zu erläutern.
- verschiedene Aktivitäten, Rollen und Risiken bei Erstellung, Betrieb und Wartung von mobilen Software-Systemen zu unterscheiden.
- Architektur und technische Eigenschaften der Android Plattform zu erläutern und zu unterscheiden.
- selbständig mobile Software-Systeme zur Lösung von konkreten Problemen für die Plattform „Android“ zu erstellen.

Kursinhalt

1. Grundlagen der mobilen Software-Entwicklung
 - 1.1 Besonderheiten von mobilen Endgeräten
 - 1.2 Besonderheiten der mobilen Software-Entwicklung
 - 1.3 Einteilung von mobilen Endgeräten
 - 1.4 Die Android-Plattform
2. Android-Systemarchitektur
 - 2.1 Das Android-System
 - 2.2 Sicherheit
 - 2.3 Kommunikation mit Netzwerken

3. Entwicklungsumgebung
 - 3.1 Android Studio
 - 3.2 Erste App und Emulator-Test
 - 3.3 Anwendungsdeployment
4. Kernkomponenten einer Android-App
 - 4.1 Überblick über die Komponenten einer Android-App
 - 4.2 Activities, Layouts und Views
 - 4.3 Ressourcen
 - 4.4 Zusammenfassung in einer App
 - 4.5 Grafische Gestaltung
5. Interaktion zwischen Anwendungskomponenten
 - 5.1 Intents
 - 5.2 Services
 - 5.3 Broadcast Receive
6. Fortgeschrittene Techniken
 - 6.1 Threading
 - 6.2 Anwendungsspeicher

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Becker, A./Pant, M. (2015): Android 5. Programmieren für Smartphones und Tablets. 4. Auflage, dpunkt.verlag, Heidelberg.
- Eason, J. (2014): Android Studio 1.0. (URL: <https://android-developers.googleblog.com/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Franke, F./Ippen, J. (2012): Apps mit HTML5 und CSS3. Galileo Computing, Bonn.
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>)
- Google Inc. (Hrsg.) (2015): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).
- Hipp, Wyrick & Company, Inc. (Hrsg.) (2015): SQLite Webseite. (URL: <http://sqlite.org/index.html> [letzter Zugriff: 12.06.2015]).
- Künneht, T. (2015): Android 5. Apps entwickeln mit Android Studio. 3. Auflage, Rheinwerk Computing, Bonn.
- Post, U. (2014): Android Apps entwickeln. 4. Auflage, Galileo Computing, Bonn.
- Ross, M. (2013): Phone Gap. Mobile Cross-Plattform-Entwicklung mit Apache Cordova & Co. dpunkt.verlag, Heidelberg.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Fallstudie

Zeitaufwand Studierende					
Selbststudium 110 h	Präsenzstudium 0 h	Tutorium 20 h	Selbstüberprüfung 20 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Mobile Software Engineering II

Kurscode: DLMIWMB02

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

In diesem Kurs erstellen die Studierenden selbständig eine mobile Anwendung und dokumentieren deren Konzeption und Umsetzung.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- selbständig eine kleine mobile Anwendung zu konzipieren und prototypisch zu erstellen, um eine gezielte Aufgabe zu lösen.
- typische Probleme und Herausforderungen in der praktischen Umsetzung kleiner mobiler Anwendungen zu erkennen.
- die Konzeption und die Umsetzung von kleinen, eigenständig konzipierten und umgesetzten mobilen Anwendungen zu dokumentieren.

Kursinhalt

- Konzeption, Umsetzung und Dokumentation von kleinen, mobilen Anwendungen auf Basis einer konkreten Aufgabenstellung. Mögliche Themen sind zum Beispiel:
 - Eine Radio-App, um den Austausch zwischen Hörer und Sender allgemein, aber vor allem zwischen Hörern und Radiomoderatoren zu verbessern
 - Eine App, mit der eine Gruppe von Brettspielfans ihren regelmäßigen abendlichen Spieltermin besser organisieren kann
 - Eine App, mit der die Betreuer von Abschlussarbeiten an der IUBH ihre Betreuungsprozesse verbessern können

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Becker, A./Pant, M. (2015): Android 5. Programmieren für Smartphones und Tablets. 4. Auflage, dpunkt, Heidelberg.
- Eason, J. (2014): Android Studio 1.0. (URL: <http://android-developers.blogspot.de/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Franke, F./Ippen, J. (2012): Apps mit HTML5 und CSS3. Rheinwerk Verlag, Bonn.
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>)
- Google Inc. (Hrsg.) (2015a): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015b): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015c): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).
- Hipp, Wyrick & Company, Inc. (Hrsg.) (2015): SQLite Webseite. (URL: <http://sqlite.org/index.html> [letzter Zugriff: 12.06.2015]).
- Künneht, T. (2016): Android 7. Das Praxisbuch für Entwickler. 4. Auflage, Rheinwerk, Bonn.
- Ross, M. (2013): Apache Cordova. Eine praktische Einführung in die mobile Cross-Plattform-Entwicklung mit PhoneGap. dpunkt Verlag, Heidelberg.
- Post, U. (2014): Android Apps entwickeln. Eigene Spiele-Apps für Leser mit Programmierkenntnissen. 4. Auflage, Galileo Computing, Bonn.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Projekt
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Projektbericht

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed

DLMIWMB02

Cloud Computing

Modulcode: DLMWIWCC

Modultyp	Zugangsvoraussetzungen	Niveau	ECTS	Zeitaufwand Studierende
s. Curriculum	keine	MA	10	300 h

Semester	Dauer	Regulär angeboten im	Unterrichtssprache
s. Curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	Deutsch

Modulverantwortliche(r)

Prof. Dr. Claudia Heß (Einführung in Cloud Architekturen und Serverless Computing) / Prof. Dr. Claudia Heß (Projekt: Cloud Computing)

Kurse im Modul

- Einführung in Cloud Architekturen und Serverless Computing (DLMWIWCC01)
- Projekt: Cloud Computing (DLMWIWCC02)

Art der Prüfung(en)

Modulprüfung	Teilmodulprüfung
	<p><u>Einführung in Cloud Architekturen und Serverless Computing</u></p> <ul style="list-style-type: none"> • Studienformat "Fernstudium": Klausur, 90 Minuten <p><u>Projekt: Cloud Computing</u></p> <ul style="list-style-type: none"> • Studienformat "Fernstudium": Schriftliche Ausarbeitung: Projektbericht

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

Einführung in Cloud Architekturen und Serverless Computing

- Grundlage von Cloud Computing
- Service- und Bereitstellungsmodelle in der Cloud
- Cloud Sicherheit und Datenschutz
- Anbieter im Cloud Markt
- Typische Geschäftsanwendungen in der Cloud

Projekt: Cloud Computing

Identifizierung eines Use Cases, Ideation, Konzeption und Entwicklung einer eigenen Cloud-Anwendung am Beispiel eines typischen Cloud Anbieters wie Amazon AWS oder Microsoft Azure.

Qualifikationsziele des Moduls

Einführung in Cloud Architekturen und Serverless Computing

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Definitionen und Kategorisierungen von Cloud Computing zu kennen und zu verstehen.
- Technologie-Entwicklungen zur Ermöglichung von Cloud Computing auszuführen.
- Service Modelle der Cloud zu erläutern und zu bewerten.
- Sicherheitsrisiken von Cloud Lösungen zu verstehen und für Unternehmen zu beurteilen.
- marktübliche Cloud Anbieter zu unterscheiden und deren Services zu vergleichen.
- Geschäftsanwendungen in der Cloud zu bewerten.

Projekt: Cloud Computing

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- ein prototypisches Cloud Projekt auf Basis eines ausgewählten Cloud Anbieters zu planen, umzusetzen und zu dokumentieren.
- einen passenden Cloud Service Provider zu identifizieren und dessen verfügbare Services für den Einsatz einer rezeptbasierten Cloud Architektur zu beurteilen.
- typische Probleme in verschiedenen Projektphasen der Entwicklung von Cloud Lösungen durch den Einsatz von geeigneten Methoden zu erkennen und zu bewerten.
- geschäftsrelevante Fragestellung zur Bewertung einer Cloud Lösung zu beantworten und zu bewerten.

Bezüge zu anderen Modulen im Studiengang

Baut auf Modulen aus dem Bereich Informatik & Software-Entwicklung auf

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master Programme im Bereich IT & Technik

Einführung in Cloud Architekturen und Serverless Computing

Kurscode: DLMWIWCC01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Cloud Computing steht für jederzeit abrufbare, dezentral über das Internet bereitgestellte technische Dienste, Softwareprodukte und Infrastrukturen, die die steigenden Anforderungen an die digitale Unternehmenswelt durch skalierbare und flexible Lösungen abdecken. Ziel dieses Kurses ist es daher, grundlegende Konzepte sowie Services- und Bereitstellungsmodelle von Cloud Computing zu vermitteln und Studierende in die Lage zu versetzen, passende Anwendungsfälle in der Geschäftswelt für Cloud Lösungen zu identifizieren und zu bewerten. Der Kurs geht zunächst auf grundlegende Begrifflichkeiten und Klassifizierungen des Cloud Computings ein und beschreibt notwendige Technologie-Entwicklungen, die Cloud Computing möglich gemacht haben. Dabei werden Chancen und Risiken der Nutzung von Cloud Lösungen reflektiert und bewertet. Darauf aufbauend werden typische Cloud Servicemodelle behandelt und wesentliche Sicherheitskonzepte für Cloud Lösungen diskutiert. Nach einer Vorstellung von zentralen Playern im Cloud Markt, wird auf typische Anwendungsfälle in der Geschäftswelt eingegangen. Vor diesem Hintergrund werden nicht nur die technologischen Grundlagen und Risiken von Cloud Computing behandelt, sondern es wird insbesondere darauf hingearbeitet, ein grundlegendes Verständnis über Einsatz und Nutzbarkeit von Cloud Lösungen im unternehmerischen Umfeld aufzubauen.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Definitionen und Kategorisierungen von Cloud Computing zu kennen und zu verstehen.
- Technologie-Entwicklungen zur Ermöglichung von Cloud Computing auszuführen.
- Service Modelle der Cloud zu erläutern und zu bewerten.
- Sicherheitsrisiken von Cloud Lösungen zu verstehen und für Unternehmen zu beurteilen.
- marktübliche Cloud Anbieter zu unterscheiden und deren Services zu vergleichen.
- Geschäftsanwendungen in der Cloud zu bewerten.

Kursinhalt

1. Grundlagen der Cloud Technologien
 - 1.1 Definition und Kategorisierung des Cloud Begriffs
 - 1.2 Historische Entwicklung und technische Evolution von Cloud Computing
 - 1.3 Abgrenzung zu anderen Themen (Grid-Computing, ASP u. a.)
 - 1.4 Chancen und Risiken der Nutzung

2. Ermöglichungstechnologien für Cloud Lösungen
 - 2.1 Internet/Web 2.0
 - 2.2 Data Center
 - 2.3 Virtualisierung
 - 2.4 Containerization
3. Service Modelle der Cloud
 - 3.1 Infrastructure as a Service (IaaS)
 - 3.2 Platform as a Service (PaaS)
 - 3.3 Software as a Service (SaaS)
 - 3.4 Function as a Service (FaaS).
4. Bereitstellungsmodelle der Cloud
 - 4.1 Public Cloud
 - 4.2 Private Cloud
 - 4.3 Community Cloud
 - 4.4 Hybrid Cloud
5. Cloud Sicherheit und Datenschutz
 - 5.1 Typische Sicherheitsrisiken von Anwendungen, Schnittstellen und Daten
 - 5.2 Best Practices (Cloud Verschlüsselung und Sicherheitsarchitektur)
 - 5.3 Datenschutzrechtliche Aspekte
6. Cloud Anbieter
 - 6.1 Übersicht des weltweiten Markts von Cloud Anbietern
 - 6.2 Anbieterbeispiel der „Big-4“ (AWS, IBM, Google und Microsoft)
7. Geschäftsvorfälle und Anwendungsbeispiele
 - 7.1 Business Treiber für Cloud Computing
 - 7.2 Typische Anwendungsbeispiele (Datenanalysen, ERP-Lösungen, IOT, Blockchain u. a.)

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Chang, V./Walters, R.J./Wills, G. (2015): Delivery and Adoption of Cloud Computing Services in Contemporary Organizations. IGI Global, Hershey PA.
- Frank, R./Schuhmacher, G./Tamm, A. (2019): Cloud-Transformation. Wie die Public Cloud Unternehmen verändert. Springer Vieweg, Wiesbaden.
- Longbottom, C. (2017): Evolution of Cloud Computing – How to Plan for Change. BCS The Chartered Institute for IT, Swindon UK.
- Metzger, C./Reitz, T./ Villar, J. (2011): Cloud Computing. Chancen und Risiken aus technischer und unternehmerischer Sicht. Carl Hansa Verlag, München.
- Ramachandran, M. (2016): Software security requirements management as an emerging cloud computing service. In: International Journal of Information Management, 36 (4), S. 580–590.
- Reinheimer, St. (Hrsg.) (2018): Cloud Computing. Die Infrastruktur der Digitalisierung. Springer, Wiesbaden.
- Selzer, A. (2019): Datenschutzrechtliche Zulässigkeit von Cloud-Computing-Services und deren teilautomatisierte Überprüfbarkeit. Eine Betrachtung unter Anwendung der Datenschutz-Grundverordnung. Springer Vieweg, Wiesbaden.
- Vacca, J. R. (2017): Cloud computing security: foundations and challenges. CRC Press, Boca Raton.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Projekt: Cloud Computing

Kurscode: DLMWIWCC02

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	DLMWIWCC01

Beschreibung des Kurses

In der digitalen Unternehmenswelt spielt Cloud Computing eine zentrale Rolle bei der Gestaltung moderner, skalierbarer und flexibler Unternehmenslösungen. Die Einsatzmöglichkeiten reichen von Lösungen zur Verlagerung der unternehmenseigenen Infrastruktur in die Cloud, über das Hosting komplexer Plattform- und Softwarelösungen bis hin zur Bereitstellung einfacher und kostengünstiger Services mit Hilfe von Serverless Architekturen. Ziel dieses Kurses ist es, einen zweckmäßigen Anwendungsfall für die Digitalisierung im Unternehmen zu identifizieren und diesen unter Nutzung der verfügbaren Services eines typischen Cloud Anbieters prototypisch zur Realisierung zu bringen. Dabei sollen die Konzeption und technische Umsetzung, mit Hilfe der Services des ausgewählten Cloud Anbieters, im Fokus stehen. Wichtige geschäftskritische Fragestellungen, wie die Machbarkeit, Skalierbarkeit, Sicherheit und Kosten, sollen im Rahmen der Lösungsentwicklung ebenfalls bewertet und dokumentiert werden. Die gesammelten Erkenntnisse sollen im Rahmen eines Projektberichts zusammenfassend dargestellt werden.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- ein prototypisches Cloud Projekt auf Basis eines ausgewählten Cloud Anbieters zu planen, umzusetzen und zu dokumentieren.
- einen passenden Cloud Service Provider zu identifizieren und dessen verfügbare Services für den Einsatz einer rezeptbasierten Cloud Architektur zu beurteilen.
- typische Probleme in verschiedenen Projektphasen der Entwicklung von Cloud Lösungen durch den Einsatz von geeigneten Methoden zu erkennen und zu bewerten.
- geschäftsrelevante Fragestellung zur Bewertung einer Cloud Lösung zu beantworten und zu bewerten.

Kursinhalt

- Im Rahmen des Projekts Cloud Computing stellen die Studierenden das Wissen sowie die Fertigkeiten und Kompetenzen unter Beweis, um eine prototypische, cloudbasierte Lösung für ein virtuelles Unternehmen zu identifizieren, zu konzeptionieren und zu entwickeln. Die Studierenden sollen auf Basis einer vorgegebenen oder einer eigenständig identifizierten Problemstellung Ideen und Vorschläge zur Lösung entwickeln. Auf Basis der Auswahl eines passenden Anbieters, wie Amazon AWS, soll eine Spezifizierung und Konzeption für die Entwicklung einer prototypischen Cloud Lösung, im Sinne eines Proofs-of-Concept, erfolgen.

Die Dokumentation der erstellten Lösung sowie die Bewertung im Hinblick auf Aspekte wie Sicherheit, Skalierbarkeit und Kosten bilden für die Studierenden den Abschluss des Projekts.

Literatur

Pflichtliteratur

Weiterführende Literatur

- o. V. (o. J.): AWS-Dokumentation. (URL: <https://docs.aws.amazon.com/index.html> [letzter Zugriff: 22.05.2020])
- Wadia, Y. et al. (2019): Implementing AWS: Design, Build, and Manage your Infrastructure. Packt Publishing Ltd., Birmingham UK.
- Zalazar A.S./Ballejos L./Rodriguez S. (2017): Analyzing Requirements Engineering for Cloud Computing. In: Ramachandran M./Mahmood Z. (Hrsg.): Requirements Engineering for Service and Cloud Computing. Springer, Cham.
- Zardari, S./Faniyi, F./Bahsoon R. (2013): Cloud-Based Goal Oriented Requirements Engineering. In: Mistrík, I. et. al. (Hrsg.): Aligning Enterprise, System, and Software Architectures. IGI Global, Hershey PA.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Projekt
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Projektbericht

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed

DLMWIWCC02

Computerkriminalität

Modulcode: DLMIMWCK

Modultyp s. Curriculum	Zugangsvoraussetzungen DLMIMWCK01	Niveau MA	ECTS 10	Zeitaufwand Studierende 300 h
----------------------------------	---	---------------------	-------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Alexander Lawall (Angriffsszenarien und Vorfallreaktion) / Prof. Dr. Alexander Lawall (Projekt: Cyber-Forensik)

Kurse im Modul

- Angriffsszenarien und Vorfallreaktion (DLMIMWCK01)
- Projekt: Cyber-Forensik (DLMIMWCK02)

Art der Prüfung(en)

Modulprüfung

Teilmodulprüfung

Angriffsszenarien und Vorfallreaktion

- Studienformat "Fernstudium": Klausur, 90 Minuten

Projekt: Cyber-Forensik

- Studienformat "Fernstudium": Portfolio

Anteil der Modulnote an der Gesamtnote

s. Curriculum

<p>Lehrinhalt des Moduls</p> <p>Angriffsszenarien und Vorfallreaktion</p> <ul style="list-style-type: none"> ▪ Bedrohungsszenarien ▪ Angriffsvektoren ▪ Präventive Maßnahmen ▪ Reaktive Maßnahmen ▪ Aktuelle Lage der IT-Sicherheit <p>Projekt: Cyber-Forensik</p> <p>Das Projekt befasst sich mit der Frage, welche Vorgehensweise geeignet ist, um auf computerkriminelle Vorfälle im Unternehmen reagieren zu können. Es behandelt forensische Verfahren zur Erfassung gerichtsverwertbarer Beweise sowie Empfehlungen zur Risikominimierung, zur Kommunikation und zur Prävention solcher Vorfälle.</p>	
<p>Qualifikationsziele des Moduls</p> <p>Angriffsszenarien und Vorfallreaktion</p> <p>Nach erfolgreichem Abschluss sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> ▪ Bedrohungsszenarien und deren Auswirkungen zu bewerten. ▪ Angriffsvektoren zu benennen und adäquate Gegenmaßnahmen auszuwählen. ▪ Verfahren der elektronischen Beweisführung auf gewählte Angriffsszenarien anzuwenden. ▪ präventive Maßnahmen zu erarbeiten. ▪ reaktive Maßnahmen zu benennen und deren Wirksamkeit zu bewerten. ▪ Information zur aktuellen Bedrohungssituation zu sammeln und auszuwerten. <p>Projekt: Cyber-Forensik</p> <p>Nach erfolgreichem Abschluss sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> ▪ grundlegende Methoden und Techniken der Computerforensik und deren Limitationen zu benennen. ▪ die von einer computerkriminellen Handlung betroffenen Systeme und Geschäftsprozesse zu identifizieren und eine Risikoabschätzung vorzunehmen. ▪ Maßnahmen zur Sicherstellung elektronischer Beweise zu empfehlen und deren Gerichtsverwertbarkeit zu evaluieren. ▪ Empfehlungen zur Vorfall-Kommunikation, -Reaktion und -Prävention zu entwickeln. 	
<p>Bezüge zu anderen Modulen im Studiengang</p> <p>Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung</p>	<p>Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)</p> <p>Alle Master-Programme im Bereich IT & Technik</p>

Angriffsszenarien und Vorfallreaktion

Kurscode: DLMIMWCK01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Der Kurs vermittelt Studierenden Kenntnisse zur Identifizierung und Maßnahmenplanung im Umgang mit kriminellen Angriffen im digitalen Umfeld. Er beschreibt, wie Schwachstellen in Hardware und Software sowie in deren Anwendung für kriminelle Aktivitäten ausgenutzt werden können. Dazu werden typische Bedrohungsszenarien vorgestellt und die Wege, auf denen angreifende Systeme in ein Computersystem eindringen können. Der Kurs führt zudem in Methoden der elektronischen Beweisführung ein und zeigt, wie im Angriffsfall rechtlich verwertbare Informationen gewonnen werden können. Im Anschluss werden die Entwicklung präventiver Maßnahmen und die Reaktionsmöglichkeiten im konkreten Bedrohungsfall erörtert. Der Kurs behandelt abschließend, wie aus Berichten der Sicherheitsbehörden (wie etwa BSI, Europol, NCA, FBI) Informationen zur aktuellen Sicherheitslage gewonnen werden können.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Bedrohungsszenarien und deren Auswirkungen zu bewerten.
- Angriffsvektoren zu benennen und adäquate Gegenmaßnahmen auszuwählen.
- Verfahren der elektronischen Beweisführung auf gewählte Angriffsszenarien anzuwenden.
- präventive Maßnahmen zu erarbeiten.
- reaktive Maßnahmen zu benennen und deren Wirksamkeit zu bewerten.
- Information zur aktuellen Bedrohungssituation zu sammeln und auszuwerten.

Kursinhalt

1. Einführung
 - 1.1 Computerkriminalität in Abgrenzung zu anderen Angriffen
 - 1.2 Schwachstellen in Computer und Mobilgeräten
 - 1.3 Eine Übersicht über Schadsoftware
 - 1.4 Social Engineering und der menschliche Faktor
2. Strafrechtliche Basis
 - 2.1 Identitätsmissbrauch
 - 2.2 Diebstahl von geistigem Eigentum
 - 2.3 Fälschung beweiserheblicher Daten
 - 2.4 Computerbetrug

3. Spezifische Delikte
 - 3.1 Datendiebstahl
 - 3.2 Digitale Erpressung
 - 3.3 Computersabotage
 - 3.4 Industriespionage
4. Angriffsvektoren
 - 4.1 Angriffe auf Chip- und Firmware-Ebene
 - 4.2 Angriffe auf Betriebssystemebene
 - 4.3 Angriffe auf Netzwerk- und Serverebene
 - 4.4 Angriffe auf Anwendungsebene
 - 4.5 Angriffe auf Organisationsebene
5. IT-Forensik und elektronische Beweisführung
 - 5.1 Identifizierung, Lokalisierung und der Umgang mit Polymorphismen
 - 5.2 Mechanismen zur Angriffserkennung
 - 5.3 Auffinden elektronischer Beweise
 - 5.4 Wiederherstellung von Daten und Beweiserückgewinnung
 - 5.5 Rechtliche Grenzen und prädiktive Methoden
6. Präventive Maßnahmen
 - 6.1 Maßnahmen auf Hardware-Ebene
 - 6.2 Zugangsberechtigung, Autorisierung und Authentifizierung
 - 6.3 Sensibilisierung & Schulung
 - 6.4 Vorfalldaktionsplanung
7. Reaktive Maßnahmen
 - 7.1 Erstbeurteilung und Schadensausmaß
 - 7.2 Unterbindung anhaltender Schäden
 - 7.3 Sammlung, Austausch und Verteilung von Information
 - 7.4 Zusammenarbeit mit Sicherheitsbehörden und Kooperationspartnern
 - 7.5 Handlungsempfehlungen für Unternehmen
8. Die aktuelle Sicherheitslage
 - 8.1 Aktuelle Berichte der Sicherheitsbehörden
 - 8.2 Bewertung der Empfehlungen der Sicherheitsbehörden
 - 8.3 Aktuelle Themen der Europol Awareness Campaign

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Fleischer, D. (2016): Wirtschaftsspionage. Springer Fachmedien, Wiesbaden.
- Klipper, S. (2015): Cyber Security. Ein Einblick für Wirtschaftswissenschaftler. Springer, Berlin.
- Kraft, P./Weyert, A. (2017): Network Hacking. Professionelle Angriffs- und Verteidigungstechniken gegen Hacker und Datendiebe. Franzis Verlag, München.
- Labudde, D./Spranger, M. (Hrsg.) (2017): Forensik in der digitalen Welt. Moderne Methoden der forensischen Fallarbeit in der digitalen und digitalisierten realen Welt. Springer, Berlin.
- Lenhard, T. H. (2017): Datensicherheit. Technische und organisatorische Schutzmassnahmen gegen Datenverlust und Computerkriminalität. Springer, Berlin.
- Lewis, J./Baker, S. (2013): The economic impact of cybercrime and cyber espionage. McAfee, Santa Clara, CA.
- Müller, K. R. (2018): IT-Sicherheit mit System. Integratives IT-Sicherheits-, Kontinuitäts- und Risikomanagement–Sichere Anwendungen–Standards und Practices. Springer, München.
- Yar, M./Steinmetz, K. F. (2019): Cybercrime and society. SAGE Publications, Thousand Oaks, CA.
- Hyperlinks zu aktuellen Berichten und Empfehlungen von Sicherheitsbehörden und Institutionen (z.B. BSI, Europol, FBI) werden im Kurs zur Verfügung gestellt.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Projekt: Cyber-Forensik

Kurscode: DLMIMWCK02

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	DLMIMWCK01

Beschreibung des Kurses

Das Projekt dient zur Erstellung eines Aktionsplans zur digitalen Untersuchung und Vorfallbehandlung für ein gegebenes Bedrohungsszenario. Beginnend mit dem konkreten Verdacht auf eine computerkriminelle Handlung (z. B. eines vermuteten Server-Angriffs, dem Verlust von Kundendaten oder der Manipulation von Geschäftsdaten) planen die Studierenden die Durchführung einer digitalen Untersuchung für die elektronische Beweisführung und zur Sicherstellung gerichtsverwertbarer Beweise. Mit den gewonnenen Daten werden Risiken für betroffene Unternehmensprozesse evaluiert und Empfehlungen zur Vorfall-Behandlung und -Prävention gegeben.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- grundlegende Methoden und Techniken der Computerforensik und deren Limitationen zu benennen.
- die von einer computerkriminellen Handlung betroffenen Systeme und Geschäftsprozesse zu identifizieren und eine Risikoabschätzung vorzunehmen.
- Maßnahmen zur Sicherstellung elektronischer Beweise zu empfehlen und deren Gerichtsverwertbarkeit zu evaluieren.
- Empfehlungen zur Vorfall-Kommunikation, -Reaktion und -Prävention zu entwickeln.

Kursinhalt

- Das Projekt dient zur Erstellung eines Aktionsplans für die Durchführung einer digitalen Untersuchung und zur Vorfallbehandlung für ein gegebenes Bedrohungsszenario.
- Beginnend mit dem konkreten Verdacht auf eine computerkriminelle Handlung* erarbeiten die Studierenden einen Vorgehensplan, der folgende Maßnahmen abdeckt:
 - Lokalisierung der betroffenen Systeme (Hardware und Software)
 - Identifizierung der betroffenen Unternehmensprozesse
 - Risikoabschätzung für die Auswirkung auf betroffene Unternehmensprozesse
 - Kommunikation mit internen Abteilungen, Kooperationspartnern, Kunden und der Öffentlichkeit
 - Identifizierung und Erhaltung relevanter Daten
 - Examinierung der Daten
 - Sicherstellung elektronischer Beweise und deren Gerichtsverwertbarkeit
 - Empfehlungen zur Prävention

- Der Aktionsplan soll so verfasst werden, dass er als Prozessvorlage für die kontinuierliche Vorfallobehandlung dient.
*Beispiele für Verdachtsfälle sind ein vermuteter Server-Angriff, der Verlust von Kundendaten, die Manipulation von Geschäftsdaten, die Veröffentlichung interner Firmendaten, der Verdacht auf Produktpiraterie, die Inkonsistenz elektronischer Signaturen in Unternehmensdokumenten, die digitale Erpressung eines Entscheidungsträgers oder der Verdacht auf Industriespionage.

Literatur

Pflichtliteratur

Weiterführende Literatur

- Aebi, D. (2013): Praxishandbuch Sicherer IT-Betrieb. Risiken erkennen, Schwachstellen beseitigen, IT-Infrastrukturen schützen. Springer, Berlin.
- Banaschik, M. (2011): Internationale E-Discovery und Information Governance. Praxislösungen für Juristen, Unternehmer und IT-Manager. Erich Schmidt Verlag, Berlin.
- Geschonneck, A. (2014): Computer-Forensik. Computerstraftaten erkennen, ermitteln, aufklären. dpunkt.verlag, Heidelberg.
- Hamid, J./Gianluigi, M./Lilburn, W. D. (2010): Handbook of electronic security and digital forensics. World Scientific Publishing, Singapur.
- Labudde, D./Spranger, M. (Hrsg.) (2017): Forensik in der digitalen Welt. Moderne Methoden der forensischen Fallarbeit in der digitalen und digitalisierten realen Welt. Springer, Berlin.
- Meier, S. (2017): Digitale Forensik in Unternehmen (Doktorarbeit). Universität Regensburg, Regensburg.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Projekt
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Portfolio

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed

DLMIMWCK02

Data Architectures

Module Code: DLMINFWDA

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	MA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Peter Poensgen (Data Warehousing, Pipelines and Orchestration) / Prof. Dr. Peter Poensgen (Managing Data Projects)

Contributing Courses to Module

- Data Warehousing, Pipelines and Orchestration (DLMDMDWPO01)
- Managing Data Projects (DLMDMMDP01)

Module Exam Type

Module Exam	Split Exam
	<p><u>Data Warehousing, Pipelines and Orchestration</u></p> <ul style="list-style-type: none"> • Study Format "Distance Learning": Written Assessment: Case Study <p><u>Managing Data Projects</u></p> <ul style="list-style-type: none"> • Study Format "Distance Learning": Advanced Workbook (passed / not passed)

Weight of Module

see curriculum

Module Contents**Data Warehousing, Pipelines and Orchestration**

- Principles of Data Warehousing
- Data Pipelines
- Orchestration Tools and Frameworks
- Solution Architecture
- Cloud Migration

Managing Data Projects

- Agile Project Management for data-intensive projects
- Infrastructure as Code
- Continuous Integration/Continuous Delivery
- Testing and Collaboration
- Container Communication and Networking
- Tools for Managing Data Projects

Learning Outcomes**Data Warehousing, Pipelines and Orchestration**

On successful completion, students will be able to

- explain and apply principles of data warehousing and data quality assessment.
- design and implement fully automated data processing pipelines.
- differentiate, assess, and use common data processing orchestration tools and frameworks.
- assess and evaluate different solution architectures for data warehousing and data processing orchestration.
- explain, evaluate, and apply common cloud migration techniques.
- reflect upon and discuss societal implications of automated large-scale data-processing systems.
- understand and implement requirements of interdisciplinary teams towards large-scale data processing pipelines and data warehousing.

Managing Data Projects

On successful completion, students will be able to

- use agile project management techniques to efficiently manage data-intensive projects.
- design and implement Infrastructure as Code for data-intensive systems.
- organize data-projects with Continuous Integration/Continuous Delivery pipelines.
- apply techniques for efficient collaboration and testing in data-intensive projects.
- explain the concepts and apply the techniques for container communication and networking.
- differentiate and use common tools and frameworks for managing data projects.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Data Science & Artificial Intelligence

Links to other Study Programs of IU International University of Applied Sciences (IU)

All Master Programs in the IT & Technology field

Data Warehousing, Pipelines and Orchestration

Course Code: DLMDMDWPO01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	none

Course Description

The responsibilities of a data manager expand beyond mere data storage and one-time processing tasks. To holistically manage data processing systems, it is necessary to orchestrate automated processes from a system-wide perspective, with respect to the whole data processing lifecycle and considering requirements of interdisciplinary teams as end-users of these systems. In this course, students learn the principles and practical application for Data Warehousing and Data Processing Orchestration. Definitions within this context are explained, such as Data Layers, Data Zones and Data Marts, and the distinction between Data Warehouses and Data Lakes is made. Within the context of data processing pipelines, principles such as the ETL and ELT approach are explained. Students learn the principles as well as the practical application of common cloud-based data processing orchestration tools and frameworks. They are enabled to efficiently perform tasks within this context by making use of solution architecture principles. With respect to societal implications of automated large-scale data processing, students are enabled to contribute to the public discussion about these in an academic and well-informed way. Finally, students learn about common cloud migration techniques and how to apply these principles in practice.

Course Outcomes

On successful completion, students will be able to

- explain and apply principles of data warehousing and data quality assessment.
- design and implement fully automated data processing pipelines.
- differentiate, assess, and use common data processing orchestration tools and frameworks.
- assess and evaluate different solution architectures for data warehousing and data processing orchestration.
- explain, evaluate, and apply common cloud migration techniques.
- reflect upon and discuss societal implications of automated large-scale data-processing systems.
- understand and implement requirements of interdisciplinary teams towards large-scale data processing pipelines and data warehousing.

Contents

1. Principles of Data Warehousing
 - 1.1 Data Layers, Zones and Marts
 - 1.2 Data Warehouses and Data Lakes
 - 1.3 Data Schemas
 - 1.4 Data Quality Assessment
 - 1.5 Applied Examples from Google BigQuery
2. Data Pipelines
 - 2.1 ETL and ELT
 - 2.2 OLAP and OLTP
 - 2.3 Triggers and Schedules
3. Orchestration Tools and Frameworks
 - 3.1 Airflow
 - 3.2 Google Cloud Composer
 - 3.3 Azure Data Factory
 - 3.4 Databricks
4. Solution Architecture
 - 4.1 Tasks and Responsibilities of the Solution Architect
 - 4.2 Solution Architecture Design
 - 4.3 Reference Architectures
5. Cloud Migration
 - 5.1 Lift and Shift
 - 5.2 Cloud-native
 - 5.3 Retain and Retire

Literature**Compulsory Reading****Further Reading**

- Burns, B. (2018): Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services. O'Reilly, Sebastopol, CA.
- Cote, C. (2018): Hands-On Data Warehousing with Azure Data Factory: ETL techniques to load and transform data from various sources, both on-premises and on cloud. Packt Publishing, Birmingham, UK.
- Kleppmann, M. (2017): Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. O'Reilly, Sebastopol, CA.
- Richards, M./Ford, N. (2020): Fundamentals of Software Architecture. O'Reilly, Sebastopol, CA.
- Shrivastava, S. (2020): Solutions Architect's Handbook: Kick-start your solutions architect career by learning. Packt Publishing, Birmingham, UK.
- Uttamchandani, S. (2020): The Self-Service Data Roadmap. Democratize Data and Reduce Time to Insight. O'Reilly, Sebastopol, CA.

Study Format Distance Learning

Study Format Distance Learning	Course Type Case Study
--	----------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Written Assessment: Case Study

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

Managing Data Projects

Course Code: DLMDMMDP01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	none

Course Description

Managing data projects in a collaborative, efficient, and modular way, with thorough testing phases and short feedback back loops is a challenge in its own. In this course, students acquire practical knowledge for agile project management in data-intensive projects. The Scrum theory and values are explained as well as its terminology and practical applications for data-intensive projects. Building on this framework, students learn about concepts and their implementations, such as Infrastructure as Code (IaC), Continuous Integration/Continuous Delivery (CI/CD) pipelines, as well as tools and platform solutions for agile data project management. Students learn how to build data systems in a reproducible way by applying principles of IaC with common infrastructure template languages. To build lean data systems with short feedback loops and short product increment cycles, students are enabled to build automated CI/CD pipelines with incremental testing stages, collaborative version control and branching strategies. Most modern data systems are implemented as containerized microservice architectures. In this course, students learn how to ensure a safe and reliable communication between containers and suitable networking setups. Finally, students are enabled to use tools and frameworks for managing data projects in an efficient way and in accordance with the principles which they have learned about in this course.

Course Outcomes

On successful completion, students will be able to

- use agile project management techniques to efficiently manage data-intensive projects.
- design and implement Infrastructure as Code for data-intensive systems.
- organize data-projects with Continuous Integration/Continuous Delivery pipelines.
- apply techniques for efficient collaboration and testing in data-intensive projects.
- explain the concepts and apply the techniques for container communication and networking.
- differentiate and use common tools and frameworks for managing data projects.

Contents

1. Agile Project Management for Ddata-lntensive Pprojects
 - 1.1 Scrum Theory and Values
 - 1.2 Scrum Teams in Data-Intensive Projects
 - 1.3 Scrum Events in Data-Intensive Projects
 - 1.4 Scrum Artifacts in Data-Intensive Projects

2. Infrastructure as Code
 - 2.1 Principles of Infrastructure as Code
 - 2.2 Terraform
 - 2.3 ARM templates
3. Continuous Integration/Continuous Delivery
 - 3.1 Concepts of a CI/CD Pipeline
 - 3.2 Building a CI/CD Pipeline for Data-Intensive Systems
4. Testing and Collaboration
 - 4.1 Environments and Stages
 - 4.2 Branching Strategies
 - 4.3 Testing Strategies
5. Container Communication and Networking
 - 5.1 Containers and APIs
 - 5.2 Container Orchestration and Networking
6. Tools for Managing Data Projects
 - 6.1 GitHub
 - 6.2 DevOps
 - 6.3 Jenkins

Literature**Compulsory Reading****Further Reading**

- Brikman, Y. (2019): Terraform: Up & Running: Writing Infrastructure as Code. 2nd Edition, O'Reilly, Sebastopol, CA.
- Burns, B./Beda, J., Hightower, K. (2019): Kubernetes: Up and Running: Up and Running. 2nd Edition, O'Reilly, Sebastopol, CA.
- Farcic, V. (2016): The DevOps 2.0 toolkit: Automating the continuous deployment pipeline with containerized microservices. CreateSpace, Independent Publishing Platform, Scotts Valley, CA.
- Gift, N. (2019): Python for DevOps: Learn Ruthlessly Effective Automation. 1st Edition, O'Reilly, Sebastopol, CA.
- Ibryam, B. (2019): Kubernetes Patterns: Reusable Elements for Designing Cloud Native Applications. 1st Edition, O'Reilly, Sebastopol, CA.
- Kane, S. P. (2018): Docker: Up and Running: Shipping Reliable Containers in Production. 1st Edition, O'Reilly, Sebastopol, CA.
- Leszko, R. (2019): Continuous Delivery with Docker and Jenkins: Create secure applications by building complete CI/CD pipelines. 2nd Edition, O'Reilly, Sebastopol, CA.
- Loeliger, J. (2012): Version Control with Git: Powerful tools and techniques for collaborative software development. 2nd Edition, O'Reilly, Sebastopol, CA.
- Percival, H. (2017): Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium, and JavaScript. O'Reilly, Sebastopol, CA.
- Schwaber, K./Sutherland, J. (2021): Scrum Guide. (URL: <https://scrumguides.org/scrum-guide.html>, updated on 02.03.2021 [last access: 11.03.2021]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Advanced Workbook (passed / not passed)

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

DLMDMMDP01

Database Querying

Module Code: DLMINFWDQ

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	MA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Carsten Skerra (Data Query Languages) / Prof. Dr. Carsten Skerra (NoSQL Databases)

Contributing Courses to Module

- Data Query Languages (DLMDMDQL01)
- NoSQL Databases (DLMDMNDB01)

Module Exam Type

Module Exam

Split Exam

Data Query Languages

- Study Format "Distance Learning": Advanced Workbook (passed / not passed)

NoSQL Databases

- Study Format "Distance Learning": Advanced Workbook

Weight of Module

see curriculum

<p>Module Contents</p> <p>Data Query Languages</p> <ul style="list-style-type: none"> ▪ Definition of Data Query Languages and Typical Examples ▪ Different Types of Data and the Role of Databases ▪ Data Query Languages and Standards ▪ Fundamentals of SQL ▪ Use of Data Query Languages for NoSQL Database and other Purposes ▪ Data Query Languages in the Context of Application Programming <p>NoSQL Databases</p> <ul style="list-style-type: none"> ▪ SQL Databases ▪ NoSQL Concepts ▪ Key-Value-oriented Databases ▪ Document-oriented Databases ▪ Column-oriented Databases ▪ Graph-oriented Databases 	
<p>Learning Outcomes</p> <p>Data Query Languages</p> <p>On successful completion, students will be able to</p> <ul style="list-style-type: none"> ▪ understand the basics of data query languages. ▪ understand different data structuring options and types of data sources. ▪ explain the difference between various data query languages, their application and their distinction from other programming languages. ▪ review and determine data query languages for appropriate use. ▪ apply and create SQL queries on self-created and given data in relational databases. ▪ understand the use of data query languages for application programming. <p>NoSQL Databases</p> <p>On successful completion, students will be able to</p> <ul style="list-style-type: none"> ▪ differentiate between Relational Databases and NoSQL Databases in terms of usage and underlying principles. ▪ explain universal concepts of NoSQL Databases, such as no strongly enforced schema. ▪ clarify the different concepts of Key-Value-oriented Databases, Document-oriented, Column-oriented and Graph-oriented Databases, use common databases of these kinds in data-intensive projects and assess their suitabilities for specific use cases. 	
<p>Links to other Modules within the Study Program</p> <p>This module is similar to other modules in the field of Data Science & Artificial Intelligence</p>	<p>Links to other Study Programs of IU International University of Applied Sciences (IU)</p> <p>All Master Programs in the IT & Technology field</p>

Data Query Languages

Course Code: DLMDMDQL01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	none

Course Description

The course is a general introduction to data query languages and the use by application interface-oriented and programming-oriented approaches, with a focus on SQL for relational databases.

Course Outcomes

On successful completion, students will be able to

- understand the basics of data query languages.
- understand different data structuring options and types of data sources.
- explain the difference between various data query languages, their application and their distinction from other programming languages.
- review and determine data query languages for appropriate use.
- apply and create SQL queries on self-created and given data in relational databases.
- understand the use of data query languages for application programming.

Contents

1. Introduction to Data Query Languages
 - 1.1 Definition of Data Query Languages
 - 1.2 Differentiation to other Languages
 - 1.3 Typical Examples of Data Query Languages
2. Data Management
 - 2.1 Data Life Cycle
 - 2.2 Types of Datasets (Structured, Semi-Structured and Unstructured Data)
 - 2.3 Role of Databases (SQL & NoSQL Databases)
3. Fundamentals of SQL
 - 3.1 Brief Overview
 - 3.2 Data Definition Language (DDL)
 - 3.3 Data Query Language (DQL)
 - 3.4 Data Manipulation Language (DML)

4. Advanced SQL
 - 4.1 Transaction Control Language (TCL)
 - 4.2 Data Control Language (DCL)
 - 4.3 Differences between various SQL Versions (MSSQL, PL/SQL, etc.)
5. Data Query Languages for NoSQL Database and other Purposes
 - 5.1 Document Databases (N1QL/couchbase and MongoDB)
 - 5.2 Graph Databases (Cypher/Neo4j)
 - 5.3 GraphQL for APIs
6. Using Data Query Languages within Application Programming
 - 6.1 Special Aspects (Architecture, Connection Management, Coding and Testing)
 - 6.2 Examples (SQL in Python and SQL in Java)

Literature**Compulsory Reading****Further Reading**

- Badia, A. (2020): SQL for Data Science: Data Cleaning, Wrangling and Analytics with Relational Databases. Springer, Cham, Switzerland.
- Hogan, A. (2020): The Web of Data. Springer, Cham, Switzerland.
- Meier, A./Kaufmann, M. (2019): SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management. Springer, Wiesbaden, Germany.
- Molinaro, A./Graaf, R. de (2020): SQL cookbook: Query solutions and techniques for all SQL users. O'Reilly, Beijing.
- Wiese, L. (2015): Advanced Data Management: For SQL, NoSQL, Cloud and Distributed Databases. De Gruyter, Berlin.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Advanced Workbook (passed / not passed)

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

NoSQL Databases

Course Code: DLMDMNDB01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
MA	English		5	none

Course Description

The usefulness of relational SQL databases has been proven by their universal distribution and diverse applications. In some aspects, however, relational SQL databases do not meet the requirements of modern applications in terms of, for instance, flexibility and cardinality. This gave birth to a family of database concepts which became known as NoSQL databases. In this course, students learn how traditional SQL databases are different from these NoSQL databases which usually, and as one of the most noticeable characteristics, do not enforce a data schema on write. Students acquire a thorough understanding of the concepts of NoSQL databases and learn how to evaluate the suitability of various NoSQL databases for specific data-intensive projects. Students are enabled to explain the main concepts of Key-Value-oriented, Document-oriented, Column-oriented and Graph-oriented Databases and will be provided with applied examples for each of these database types. Finally, students learn how to practically use these databases in specific problem-oriented use cases.

Course Outcomes

On successful completion, students will be able to

- differentiate between Relational Databases and NoSQL Databases in terms of usage and underlying principles.
- explain universal concepts of NoSQL Databases, such as no strongly enforced schema.
- clarify the different concepts of Key-Value-oriented Databases, Document-oriented, Column-oriented and Graph-oriented Databases, use common databases of these kinds in data-intensive projects and assess their suitabilities for specific use cases.

Contents

1. SQL Databases
 - 1.1 Principles of Relational Databases
 - 1.2 Overview over common Relational Databases
 - 1.3 Introduction to SQL
 - 1.4 Cardinality and its Limits
 - 1.5 The Relational and Document Model

2. NoSQL Concepts
 - 2.1 Schemaless Data and the ACID Principle
 - 2.2 Consistency and Availability
 - 2.3 Row-based and Column-based Storage
 - 2.4 Updates and Appends
 - 2.5 Multi-model Databases
3. Key-Value-oriented Databases
 - 3.1 The Concept of Key-Value-oriented Databases
 - 3.2 Redis
 - 3.3 DynamoDB
 - 3.4 Ignite
4. Document-oriented Databases
 - 4.1 The Concept of Document-oriented Databases
 - 4.2 MongoDB
 - 4.3 CouchDB
 - 4.4 OrientDB
5. Column-oriented Databases
 - 5.1 The Concept of Column-oriented Databases
 - 5.2 Cassandra
 - 5.3 HBase
 - 5.4 CosmosDB
6. Graph-oriented Databases
 - 6.1 The Concept of Graph-oriented Databases
 - 6.2 Neo4j

Literature**Compulsory Reading****Further Reading**

- Bradshaw, S./Brazil, E./Chodorow, K. (2019): MongoDB: The Definite Guide. 3rd Edition, O'Reilly, Sebastopol, CA.
- Carpenter, J./Hewitt, E. (2020): Cassandra: The Definite Guide. 3rd Edition, O'Reilly, Sebastopol, CA.
- George, L. (2011): HBase: The Definitive Guide. 1st Edition, O'Reilly, Sebastopol, CA.
- Harrison, G. (2016): Next Generation Databases: NoSQL, NewSQL, and Big Data. 1st Edition, Apress, New York, NY.
- Hillar, G. C./Yondem, D. (2018): Guide to NoSQL with Azure Cosmos DB: Work with the massively scalable Azure database service with JSON, C#, LINQ, and .NET Core 2. 1st Edition, Packt Publishing, Birmingham, UK.
- Hodler, A. (2019): Graph Algorithms: Practical Examples in Apache Spark and Neo4j. 1st Edition, O'Reilly, Sebastopol, CA.
- Holt, B. (2011): Scaling CouchDB. 1st Edition, O'Reilly, Sebastopol, CA.
- Kelly, A./McCreary, D. (2013): Making Sense of NoSQL. 1st Edition, O'Reilly, Sebastopol, CA.
- Nelson, J. (2016): Mastering Redis. 1st Edition, Packt Publishing, Birmingham, UK.
- Ploetz, A./Kandhare, D./Kadambi, S./Wu, X. (2018): Seven NoSQL Databases in a Week. 1st Edition, Packt Publishing, Birmingham, UK.
- Sadalage, P. (2009): NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. 1st Edition, Addison-Wesley Professional, Boston, MS.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture
--	--------------------------------------

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Advanced Workbook

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Course Book <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input type="checkbox"/> Exam Template	<input type="checkbox"/> Review Book <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Guideline <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Slides

DLMDMNDB01

Sichere Software-Entwicklung

Modulcode: DLMCSEEDSO_D

Modultyp s. Curriculum	Zugangsvoraussetzungen <ul style="list-style-type: none"> ▪ keine ▪ DLMCSEEDSO01_D oder DLMCSEEDSO01_E 	Niveau MA	ECTS 10	Zeitaufwand Studierende 300 h
----------------------------------	---	---------------------	-------------------	---

Semester s. Curriculum	Dauer Minimaldauer: 1 Semester	Regulär angeboten im WiSe/SoSe	Unterrichtssprache Deutsch
----------------------------------	---	--	--------------------------------------

Modulverantwortliche(r)

Prof. Dr. Sandra Rebholz (Sichere Software-Entwicklung) / Prof. Dr. Sandra Rebholz (Projekt: Sichere Software-Implementierung)

Kurse im Modul

- Sichere Software-Entwicklung (DLMCSEEDSO01_D)
- Projekt: Sichere Software-Implementierung (DLMCSEEDSO02_D)

Art der Prüfung(en)

Modulprüfung

Teilmodulprüfung

Sichere Software-Entwicklung

- Studienformat "Fernstudium": Klausur,
90 Minuten

Projekt: Sichere Software-Implementierung

- Studienformat "Fernstudium": Schriftliche
Ausarbeitung: Projektbericht

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls**Sichere Software-Entwicklung**

- Sicheres Software Design und -Implementierung
- Sicherheitsprüfung und -Auditierung
- Patch- und Schwachstellenmanagement
- Software-Lebenszyklus

Projekt: Sichere Software-Implementierung

- Sicheres Software Design und -Implementierung
- Sicherheitsprüfung und -Auditierung
- Patch- und Schwachstellenmanagement
- Software-Lebenszyklus

Qualifikationsziele des Moduls**Sichere Software-Entwicklung**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- sichere Anwendungen zu entwerfen.
- zu verstehen, was zu Software-Kompromittierung führt.
- gewöhnliche Kodierungsfehler zu vermeiden.
- den sicheren Software-Lebenszyklus zu steuern.
- ein strenges Sicherheitsprüfungssystem anzuwenden.
- die Offenlegung von Schwachstellen zu steuern.

Projekt: Sichere Software-Implementierung

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Sicherheitsmaßnahmen für ein einfaches Software-Projekt zu entwerfen.
- häufige Kodierungs- und Designfehler zu vermeiden.
- zu definieren, welche Schritte erforderlich sind, um sicheren Code zu implementieren.
- einen Prozess zu schaffen, der die kontinuierliche Sicherheit der Anwendung während ihrer gesamten Lebensdauer gewährleistet.
- die Offenlegung von Schwachstellen effektiv zu nutzen.

Bezüge zu anderen Modulen im Studiengang

Baut auf Modulen aus dem Bereich Informatik & Software-Entwicklung auf

Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)

Alle Master-Programme aus dem Bereich IT & Technik

Sichere Software-Entwicklung

Kurscode: DLMCSEEDSO01_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	keine

Beschreibung des Kurses

Der Angriff auf Schwachstellen in unsicherer Software ist ein führender Angriffsweg für Kriminelle und böswillige staatliche Akteure. Das Auffinden unbekannter so genannter Zero-Day-Schwachstellen ist ein zentrales Werkzeug für professionelle Kriminelle. Daher ist es von größter Bedeutung, sichere Software zu entwickeln und zu implementieren. Zuerst müssen wir allgemeine Softwareschwächen verstehen und diese dann so früh wie möglich in der Entwicklung und im Software-Lebenszyklus durch eine "Security-by-Design"-Philosophie vermeiden. Außerdem soll ein Prozess für Sicherheitstests und die Offenlegung von Schwachstellen durchgeführt und gesteuert werden. Die Entwicklung und Implementierung von zeitgerechten Softwareupdates „Patches“ ist essentiell.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- sichere Anwendungen zu entwerfen.
- zu verstehen, was zu Software-Kompromittierung führt.
- gewöhnliche Kodierungsfehler zu vermeiden.
- den sicheren Software-Lebenszyklus zu steuern.
- ein strenges Sicherheitsprüfungssystem anzuwenden.
- die Offenlegung von Schwachstellen zu steuern.

Kursinhalt

1. Security-by-Design
 - 1.1 IT-Unterstützung und Tests durch die "Shift Left"-Methodologie
 - 1.2 Skriptbasierte Steuerung - Infrastruktur as a Code
 - 1.3 Vorteile einer frühzeitigen Berücksichtigung der Sicherheit
2. Privacy-by-Design
 - 2.1 Verschlüsselung
 - 2.2 Schutz der Privatsphäre durch Differentielm Privacy
 - 2.3 Zero-Knowledge-Beweise und Protokolle

3. Prüfung und Auditierung
 - 3.1 Prüfung der Unit
 - 3.2 Sicherheitsprüfung
 - 3.3 Prüfung von Sicherheitscodes
4. Sicherheit der Software-Lieferkette
 - 4.1 Sicherheit von Paketen
 - 4.2 Container-Sicherheit
 - 4.3 Überlegungen zur Programmiersprache
5. Gängige Programmierfehler
 - 5.1 Klassen von Fehlern
 - 5.2 Quellen von Fehlern
 - 5.3 Schweregrad der Fehler
6. Projektleitung
 - 6.1 Der Software-Lebenszyklus
 - 6.2 Umgang mit der Offenlegung von Schwachstellen
 - 6.3 Steuern von Patches/Aktualisierungen
 - 6.4 Steuern von Pentesting- und Bug-Bounty-Programmen
7. DevSecOps
 - 7.1 DevOps
 - 7.2 Cloud-Sicherheit
 - 7.3 Kontinuierliche Integration, Prüfung und Bereitstellung
 - 7.4 Kurzlebige Prozesse
 - 7.5 Automatisierung

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Adkins, H. et al (2020): Building Secure and Reliable Systems. O'Reilly Media, Newton, MA.
- Common Weakness Enumeration, <https://cwe.mitre.org/>
- Dwork, C. / Roth, A. (2014): The Algorithmic Foundations of Differential Privacy. In Foundations and Trends in Theoretical Computer Science Vol. 9, Nos. 3–4 (2014) 211–407.
- The Open Web Application Security Project, <https://owasp.org/>

Studienformat Fernstudium

Studienform Fernstudium	Kursart Online-Vorlesung
-----------------------------------	------------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Ja Evaluation: Nein
Prüfungsleistung	Klausur, 90 Minuten

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
90 h	0 h	30 h	30 h	0 h	150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input type="checkbox"/> Folien

Projekt: Sichere Software-Implementierung

Kurscode: DLMCSEEDSO02_D

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		5	DLMCSEEDSO01_D oder DLMCSEEDSO01_E

Beschreibung des Kurses

Frei nach dem Spruch „Software is eating the world“ kann es sich keine Organisation leisten, unsicheren Code einzusetzen, ohne letztendlich schlimme Folgen zu erleiden. In diesem Projekt sollen Studierende eine sichere Anwendungsimplementierung in Angriff nehmen und einen Bericht schreiben, in dem die getroffenen Entscheidungen zur Gewährleistung der Sicherheit des laufenden Systems begründet werden.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Sicherheitsmaßnahmen für ein einfaches Software-Projekt zu entwerfen.
- häufige Kodierungs- und Designfehler zu vermeiden.
- zu definieren, welche Schritte erforderlich sind, um sicheren Code zu implementieren.
- einen Prozess zu schaffen, der die kontinuierliche Sicherheit der Anwendung während ihrer gesamten Lebensdauer gewährleistet.
- die Offenlegung von Schwachstellen effektiv zu nutzen.

Kursinhalt

- Für ein gegebenes Problem und/oder einen gegebenen Kontext entwirft und entwickelt der Studierende ein einfaches Softwareprojekt und reicht dann einen Bericht, Code und Daten ein, die die Entscheidungen des Sicherheitsdesigns sowie Pläne für den zukünftigen Software-Lebenszyklus beschreiben. Spezifische Projekte werden vom Tutor zur Verfügung gestellt, aber Vorschläge der Studierenden können berücksichtigt werden.

Literatur

Pflichtliteratur

Weiterführende Literatur

- Adkins, H. et al (2020): Building Secure and Reliable Systems. 1st edition, O'Reilly Media, Newton, MA.
- Common Weakness Enumeration, <https://cwe.mitre.org/>
- Dwork, C. / Roth, A. (2014): The Algorithmic Foundations of Differential Privacy. In Foundations and Trends in Theoretical Computer Science Vol. 9, Nos. 3–4 (2014) 211–407.
- The Open Web Application Security Project, <https://owasp.org/>

Studienformat Fernstudium

Studienform Fernstudium	Kursart Projekt
-----------------------------------	---------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Projektbericht

Zeitaufwand Studierende					
Selbststudium 120 h	Präsenzstudium 0 h	Tutorium 30 h	Selbstüberprüfung 0 h	Praxisanteil 0 h	Gesamt 150 h

Lehrmethoden	
<input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur	<input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed

DLMCSEEDSO02_D

Masterarbeit

Modulcode: DLMMAB

Modultyp	Zugangsvoraussetzungen	Niveau	ECTS	Zeitaufwand Studierende
s. Curriculum	Gemäß Studien- und Prüfungsordnung	MA	15	450 h

Semester	Dauer	Regulär angeboten im	Unterrichtssprache
s. Curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	Deutsch

Modulverantwortliche(r)

Studiengangsleiter (SGL) (Masterarbeit) / Studiengangsleiter (SGL) (Kolloquium)

Kurse im Modul

- Masterarbeit (DLMMAB01)
- Kolloquium (DLMMAB02)

Art der Prüfung(en)

Modulprüfung

Teilmodulprüfung

Masterarbeit

- Studienformat "Fernstudium": Schriftliche Ausarbeitung: Masterarbeit (90)

Kolloquium

- Studienformat "Fernstudium": Kolloquium (10)

Anteil der Modulnote an der Gesamtnote

s. Curriculum

<p>Lehrinhalt des Moduls</p> <p>Masterarbeit</p> <ul style="list-style-type: none"> ▪ Masterarbeit <p>Kolloquium</p> <ul style="list-style-type: none"> ▪ Kolloquium zur Masterarbeit 	
<p>Qualifikationsziele des Moduls</p> <p>Masterarbeit</p> <p>Nach erfolgreichem Abschluss sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> ▪ eine Problemstellung aus ihrem Studienschwerpunkt unter Anwendung der fachlichen und methodischen Kompetenzen, die sie im Studium erworben haben, zu bearbeiten. ▪ eigenständig – unter fachlich-methodischer Anleitung eines akademischen Betreuers – ausgewählte Aufgabenstellungen mit wissenschaftlichen Methoden zu analysieren, kritisch zu bewerten sowie entsprechende Lösungsvorschläge zu erarbeiten. ▪ eine dem Thema der Masterarbeit angemessene Erfassung und Analyse vorhandener (Forschungs-)Literatur vorzunehmen. ▪ eine ausführliche schriftliche Ausarbeitung unter Einhaltung wissenschaftlicher Methoden zu erstellen. <p>Kolloquium</p> <p>Nach erfolgreichem Abschluss sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> ▪ eine Problemstellung aus ihrem Studienschwerpunkt unter Beachtung akademischer Präsentations- und Kommunikationstechniken vorzustellen. ▪ das in der Masterarbeit gewählte wissenschaftliche und methodisch Vorgehen reflektiert darzustellen. ▪ themenbezogene Fragen von Fachexperten (Gutachter der Masterarbeit) aktiv zu beantworten. 	
<p>Bezüge zu anderen Modulen im Studiengang</p> <p>Alle Module im Masterprogramm</p>	<p>Bezüge zu anderen Studiengängen der IU Internationale Hochschule (IU)</p> <p>Alle Masterprogramme im Fernstudium</p>

Masterarbeit

Kurscode: DLMMAB01

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		13.5	Gemäß Studien- und Prüfungsordnung

Beschreibung des Kurses

Ziel und Zweck der Masterarbeit ist es, die im Verlauf des Studiums erworbenen fachlichen und methodischen Kompetenzen in Form einer akademischen Abschlussarbeit mit thematischem Bezug zum Studienschwerpunkt erfolgreich anzuwenden. Inhalt der Masterarbeit kann eine praktisch-empirische oder aber theoretisch-wissenschaftliche Problemstellung sein. Studierende sollen unter Beweis stellen, dass sie eigenständig unter fachlich-methodischer Anleitung eines akademischen Betreuers eine ausgewählte Problemstellung mit wissenschaftlichen Methoden analysieren, kritisch bewerten und Lösungsvorschläge erarbeiten können. Das von dem Studierenden zu wählende Thema aus dem jeweiligen Studienschwerpunkt soll nicht nur die erworbenen wissenschaftlichen Kompetenzen unter Beweis stellen, sondern auch das akademische Wissen des Studierenden vertiefen und abrunden, um seine Berufsfähigkeiten und -fertigkeiten optimal auf die Bedürfnisse des zukünftigen Tätigkeitsfeldes auszurichten.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- eine Problemstellung aus ihrem Studienschwerpunkt unter Anwendung der fachlichen und methodischen Kompetenzen, die sie im Studium erworben haben, zu bearbeiten.
- eigenständig – unter fachlich-methodischer Anleitung eines akademischen Betreuers – ausgewählte Aufgabenstellungen mit wissenschaftlichen Methoden zu analysieren, kritisch zu bewerten sowie entsprechende Lösungsvorschläge zu erarbeiten.
- eine dem Thema der Masterarbeit angemessene Erfassung und Analyse vorhandener (Forschungs-)Literatur vorzunehmen.
- eine ausführliche schriftliche Ausarbeitung unter Einhaltung wissenschaftlicher Methoden zu erstellen.

Kursinhalt

- Im Rahmen der Masterarbeit muss die Problemstellung sowie das wissenschaftliche Untersuchungsziel klar herausgestellt werden. Die Arbeit muss über eine angemessene Literaturanalyse den aktuellen Wissensstand des zu untersuchenden Themas widerspiegeln. Der Studierende muss seine Fähigkeit unter Beweis stellen, das erarbeitete Wissen in Form einer eigenständigen und problemlösungsorientierten Anwendung theoretisch und/oder empirisch zu verwerten.

Literatur

Pflichtliteratur

Weiterführende Literatur

- Hunziker, A.W. (2010): Spass am wissenschaftlichen Arbeiten. So schreiben Sie eine gute Semester-, Bachelor- oder Masterarbeit. 4. Auflage, SKV, Zürich.
- Wehrlin, U. (2010): Wissenschaftliches Arbeiten und Schreiben. Leitfaden zur Erstellung von Bachelorarbeit, Masterarbeit und Dissertation – von der Recherche bis zur Buchveröffentlichung. AVM, München.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Thesis-Kurs
-----------------------------------	-------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Schriftliche Ausarbeitung: Masterarbeit

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
405 h	0 h	0 h	0 h	0 h	405 h

Lehrmethoden
Die Studierenden schreiben ihre Masterarbeit eigenständig unter der methodischen und wissenschaftlicher Anleitung eines akademischen Betreuers.

Kolloquium

Kurscode: DLMMAB02

Niveau	Unterrichtssprache	SWS	ECTS	Zugangsvoraussetzungen
MA	Deutsch		1.5	Gemäß Studien- und Prüfungsordnung

Beschreibung des Kurses

Das Kolloquium wird nach Einreichung der Masterarbeit durchgeführt. Es erfolgt auf Einladung der Gutachter. Im Rahmen des Kolloquiums müssen die Studierenden unter Beweis stellen, dass sie den Inhalt und die Ergebnisse der schriftlichen Arbeit in vollem Umfang eigenständig erbracht haben. Inhalt des Kolloquiums ist eine Präsentation der wichtigsten Arbeitsinhalte und Untersuchungsergebnisse durch den Studierenden, und die Beantwortung von Fragen der Gutachter.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- eine Problemstellung aus ihrem Studienschwerpunkt unter Beachtung akademischer Präsentations- und Kommunikationstechniken vorzustellen.
- das in der Masterarbeit gewählte wissenschaftliche und methodisch Vorgehen reflektiert darzustellen.
- themenbezogene Fragen von Fachexperten (Gutachter der Masterarbeit) aktiv zu beantworten.

Kursinhalt

- Das Kolloquium umfasst eine Präsentation der wichtigsten Ergebnisse der Masterarbeit, gefolgt von der Beantwortung von Fachfragen der Gutachter durch den Studierenden.

Literatur

Pflichtliteratur

Weiterführende Literatur

- Renz, K.-C. (2016): Das 1 x 1 der Präsentation. Für Schule, Studium und Beruf. 2. Auflage, Springer Gabler, Wiesbaden.

Studienformat Fernstudium

Studienform Fernstudium	Kursart Thesis-Kurs
-----------------------------------	-------------------------------

Informationen zur Prüfung	
Prüfungszulassungsvoraussetzungen	BOLK: Nein Evaluation: Nein
Prüfungsleistung	Kolloquium

Zeitaufwand Studierende					
Selbststudium	Präsenzstudium	Tutorium	Selbstüberprüfung	Praxisanteil	Gesamt
45 h	0 h	0 h	0 h	0 h	45 h

Lehrmethoden
Moderne Präsentationstechnologien stehen zur Verfügung.