**Modulhandbuch zum Studiengang**

**Applied Computer Science**

**(Master of Science)**

**Fakultät Informatik**

Inhalt

[Pflichtmodule 3](#_Toc99231624)

[Distributed Systems 3](#_Toc99231625)

[Signals and Systems 4](#_Toc99231626)

[IT Security 6](#_Toc99231627)

[Computer Graphics 7](#_Toc99231628)

[Mobile Systems 9](#_Toc99231629)

[Computational Intelligence 11](#_Toc99231630)

[Agile Software Development 13](#_Toc99231631)

[Service oriented Networks 15](#_Toc99231632)

[Web Applications 17](#_Toc99231633)

[Seminar 18](#_Toc99231634)

[Project 19](#_Toc99231635)

[Master Thesis 20](#_Toc99231636)

[Master Colloquium 22](#_Toc99231637)

[Wahlbereich I 23](#_Toc99231638)

[Semantic Technologies in Distributed Systems 23](#_Toc99231639)

[IT Security - Advanced Chapters 24](#_Toc99231640)

[Text Analysis and Data Search 25](#_Toc99231641)

[Selected Chapters Functional Programming 27](#_Toc99231642)

[Wahlpflichtbereich II 28](#_Toc99231643)

[3D Content Creation 28](#_Toc99231644)

[Interactive Systems 30](#_Toc99231645)

[Image Processing 1 31](#_Toc99231646)

[Image Processing 2 32](#_Toc99231647)

[Media Production 33](#_Toc99231648)

[Virtual and Augmented Environments 34](#_Toc99231649)

[Human Machine Interaction 36](#_Toc99231650)

# Pflichtmodule

|  |  |  |  |
| --- | --- | --- | --- |
| Modulname | | Distributed Systems | |
| Modulverantwortlicher/ Modulverantwortliche | | Prof. Dr. Erwin Neuhardt | |
| Qualifikationsziele | | Die Studierenden kennen die Bestandteile eines verteilten Systems. Sie kennen verschiedene Technologien zur Kommunikation in verteilten System. Die kennen Möglichkeiten zur Skalierung verteilter Systeme und Datenstrukturen zum parallelen Zugriff in verteilten Systemen. Sie können ein verteiltes System in der Programmiersprache Java umsetzen. | |
| Modulinhalte | | Konzepte und Technologien zur Entwicklung verteilter Systeme   * Architektur und Eigenschaften verteilter Systeme: Client-Server Architektur, Transparenz * Programmierkonzepte zur Kommunikation in verteilten Systemen: Socket, Remote Procedure Call, Remote Method Invocation, REST, asynchrone nachrichtenbasierte Kommunikation (Java Message Service) * Parallele Programmierung: Java Threads, Synchronisation und Koordination von Threads, Datenstrukturen für parallele Zugriffe, Threadnutzung im Java Executor Framework | |
| Lehrformen | | Vorlesung (2 SWS) und Übung am PC (2 SWS) | |
| Voraussetzungen für die Teilnahme | | Formelle Voraussetzungen bestehen nicht. Für die erfolgreiche Teilnahme sollten Kenntnisse der Programmiersprache Java vorhanden sein. | |
| Literatur / multimediale Lehr-und Lernprogramme | | Andrew S. Tanenbaum, Maarten van Steen: Distributed Systems, Published by Maarten von Steen, 2017  w/o author: Sockets, Java Remote Method Invocation, Concurrency, Online unter docs.oracle.com  Leonard Richadson, Mike Amundsen: RESTful Web APIs, O’Reilly, 2013  Brian Goetz, Joshua Bloch, Joseph Bowbeer, Doug Lea, David Holmes, Tim Peierls: Java Concurrency in Practice, Addison-Wesley, 2006  David A. Chappell, Richard Monson-Haefel: Java Message Service, O’Reilly 2009 | |
| Lehrbriefautor | |  | |
| Verwendbarkeit | | Master-Studiengang Applied Computer Science | |
| Arbeitsaufwand / Gesamtworkload | | Präsenzzeit: 60 Stunden  Selbststudium: 60 Stunden  Prüfung und Prüfungsvorbereitung: 30 Stunden | |
| ECTS und Gewichtung der Note in der Gesamtnote | | 5 ECTS; Gewichtung: 5/120 | |
| Leistungsnachweis | | Schriftliche Prüfungsleistung (Klausur am PC 120 Minuten) | |
| Semester | | 1. Semester | |
| Häufigkeit des Angebots | | einmal im Studienjahr | |
| Dauer | | 1 Semester | |
| Art der Lehrveranstaltung | | Pflichtmodul | |
| Besonderes | |  | |
| Modulname | | Signals and Systems | |
| Modulverantwortlicher / Modulverantwortliche | | Prof. Dr. Martin Golz | |
| Qualifikationsziele | | Die Studierenden sollen in der Lage sein,   * Typische Problemstellungen der Signalverarbeitung zu analysieren, * Integraltransformationen kontinuierlicher Funktionen zu verstehen, * Diskrete Transformationen von Abtastfolgen zu verstehen, * Die diskrete Fourier-Transformation zu verstehen und anzuwenden, * Digitale Filter zu verstehen und anzuwenden, * Die Spektralanalyse stochastischer Signale zu verstehen und anzuwenden, * Modelle stochastischer Prozesse und deren Optimierung zu verstehen, * Methoden der Zeit-Frequenzanalyse zu verstehen und anzuwenden. | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen, Ver­ste­hen** | **Anwen­den** | **Analy­sieren, Bewer­ten** | **Synthe­tisieren** | | Fourier-Integral | X |  |  |  | | Fourier-Reihe | X |  |  |  | | Faltungsintegral | X | X |  |  | | Abtast-Theorem & Aliasing | X | X | X |  | | Diskrete Fourier-Transformation | X | X | X |  | | Lineare zeitinvariante Systeme | X | X | X |  | | Stochastische Prozesse, Spektralschätzung | X | X | X |  | | Modelle stochastischer Prozesse | X | X |  |  | | Zeit-Frequenz-Analyse | X | X |  |  |   \_ | | | |
| Modulinhalte | | 1. Einführung 2. Fourier-Integral    1. Integraltransformationen, Fourier-Kernfunktion    2. Dirichlet-Bedingungen    3. Elementarsignale    4. Eigenschaften    5. Signalenergie, Signalleistung, Dezibel-Maß    6. Bandbreite 3. Fourier-Reihe 4. Faltungsintegral 5. Abtast-Theorem 6. Diskrete Fourier-Transformation    1. Eigenschaften    2. Diskrete Walsh-Transformation, z-Transformation 7. Lineare, zeitinvariante Systeme    1. Eigenschaften    2. Impulsantwort, Übertragungsfunktion, Bode-Diagramm    3. Pol-Nullstellen-Bild, Stabilität    4. Zustandsraumbeschreibung 8. Stochastische Signale    1. Eigenschaften    2. Wahrscheinlichkeitsdichtefunktion    3. Wiener-Chintschin-Theorem, spektrale Leistungsdichte    4. Cepstrum    5. Spektralschätzung    6. Modelle stochastischer Prozesse    7. Optimierung von Prozessmodellen    8. Zeitreihenvorhersage 9. Zeit-Frequenz-Analyse    1. Kurzzeit-Fourier-Transformation    2. Gabor-Reihe    3. Kontinuierliche Wavelet-Transformation    4. Diskrete Wavelet-Transformation und Multi-Skalen-Analyse | |
| Lehrformen | | Tafelvorlesung mit   * Lückenskripten * Rechnerbasierter Präsentation * Demonstrationsprogrammen   Übungen im Computer-Pool   * Programmierung mit MATLAB und Signalverarbeitungs-Bibliothek * Klärung offener Fragen | |
| Voraussetzungen für die Teilnahme | | Formelle Voraussetzungen bestehen nicht. Grundkenntnisse in Linearer Algebra, Analysis, Statistik sind erforderlich. | |
| Literatur | | Haykin, van Veen (2003) Signals and systems. Wiley  Percival, Walden (2000) Wavelet methods for time series analysis. Cambridge University Press | |
| Lehrbriefautor | |  | |
| Verwendbarkeit | | Master-Studiengang Applied Computer Science | |
| Arbeitsaufwand / Workload | | 150 Stunden, davon 60 Präsenzstunden und 90 Stunden Selbststudium | |
| ECTS und Gewichtung | | 5 Kreditpunkte, Gewichtung der Note in der Gesamtnote: 5 / 120 | |
| Leistungsnachweis | | Schriftliche Prüfung (120 Minuten) | |
| Semester | | 1. Semester | |
| Häufigkeit des Angebots | | Einmal im Studienjahr | |
| Dauer | | Ein Semester | |
| Art der Lehrveranstaltung | | Pflichtmodul | |
| Besonderes | |  | |

|  |  |
| --- | --- |
| Modulname | IT Security |
| Modulverantwortlicher / Modulverantwortliche | Prof. Ralf C. Staudemeyer |
| Qualifikationsziele | Ziel dieses Kurses ist frühzeitig zu lernen das Sicherheitsniveau eines Systems oder Dienstes zu bewerten, Schwachstellen konkret zu benennen und den potentiellen Schaden eines erfolgreichen Angriffs im Vorfeld abzuschätzen. Grundlegendes Wissen über die wichtigsten Konzepte für den Betrieb sicherer und (meist) verteilter Systeme, dazu gehören u.a. Teilkomponenten aus den Bereichen Betriebssysteme und Rechnernetzwerke. |
| Modulinhalte | Der Fokus dieses Kurses liegt in der Vertiefung des Verständnisses grundlegender kryptographische Techniken zur Gewährleistung von Integrität und Vertraulichkeit von Informationen. Dazu gehören Teilkomponenten wie bspw. Schlüsselmanagement, Biometrie, Authentifikation in verteilten Systemen und grundlegende Protokolle und Standards für die Netzwerk- und Anwendungssicherheit. |
| Lehrformen | Vorlesung (2 SWS), Übung (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht |
| Literatur / multimediale Lehr- und Lernprogramme | * Eckert, C. (2018). IT-Sicherheit. Berlin, München, Boston. De Gruyter * Stallings, W. (2016). Cryptography and network security, principles and practices (7th edition). Prentice Hall. * Paar, C., & Pelzl, J. (2010). Understanding Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg * Schneier, B. (1996), Applied Cryptography, John Wiley & Sons * Ausgewählte Quellen (werden bekanntgegeben). |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | Präsenzstudium: 60h; Selbststudium incl. Prüfungsvorbereitung: 90h. |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (Anteil an der Gesamtnote 5/120) |
| Leistungsnachweis | Schriftliche Prüfungsleistung |
| Semester | 1. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Computer Graphics |
| Modulverantwortlicher / Modulverantwortliche | Prof. Hartmut Seichter, PhD |
| Qualifikationsziele | Computergrafik ist ein Schmelztiegel von Technologien in der Informatik mit dem Ziel visuelle Inhalte effizient zu generieren und dem Nutzer zu präsentieren. Studierende können den Zusammenhang von visuellen Technologien in der Informatik, den zugrunde liegenden mathematischen Konzepte und der Physiognomie des Menschen, insbesondere des Sehapparates herstellen. Sie können die Eigenschaften verschiedener Darstellungsformen und -techniken analysieren und bewerten. Sie lernen grundsätzliche Technologien der 3D Echtzeitdarstellung kennen und wenden diese an. |
| Modulinhalte | * Grundkenntnisse der menschlichen Wahrnehmung * Grundkonzepte der Bilderzeugung, Speicherung und Transformation * Anwendungen von Computergrafik * Technologien zur Bilddarstellung * 3D Modelle, insbesondere Surface- und Volumemodelle * Transformationspipeline * Homogene Vektorräume und Transformationen * Szenengraphen und Echtzeit Rendering APIs * Bildsyntheseverfahren * Geometrie und Bild Samplingverfahren und Anti-Aliasing Strategien * Lichttransport, Physikalische Beleuchtungsmodelle * Texturierungsverfahren * Überblick Visualisierung * Graphische Nutzeroberflächen und Systeme |
| Lehrformen | Vorlesung und Übungen |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für die erfolgreiche Teilnahme sollten Kenntnisse in linearer Algebra vorhanden sein. |
| Literatur / multimediale Lehr-und Lernprogramme | * H5P Lernmodule * Lernforum * Folien * Auszug aus der Literaturliste:   + Bar-Zeev, Avi. Scenegraphs: Past, Present and Future, 2003 http://www.realityprime.com/scenegraph.php   + Burley, Brent: “Physically-Based Shading at Disney.” In ACM SIGGRAPH, 2012:1–7, 2012   + Goldstein, E. Bruce: Sensation and Perception. 3rd ed. Belmont, Calif.: Wadsworth Pub. Co., 1989   + Hughes, John F: Computer Graphics: Principles and Practice. Third edition. Upper Saddle River, New Jersey: Addison-Wesley, 2014   + Shirley, Peter, and R. Keith Morley: Realistic Ray Tracing. 2. ed. Natick, Mass: A K Peters, 2003 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand / Gesamtworkload | * Vorlesung 2 SWS * Übung 2 SWS * Workload: Präsenz (Vorlesung + Übung) 120 h, Prüfungsvorbereitung 30 h |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (5/120) |
| Leistungsnachweis | schriftliche Prüfungsleistung (90 Minuten) |
| Semester | erstes Semester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | ein Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Mobile Systems |
| Modulverantwortlicher / Modulverantwortliche | Dr.-Ing. David Sommer |
| Qualifikationsziele | Die Studierenden lernen wesentliche Konzepte und Technologien zur Ent­wick­lung smarter, mobiler Anwendungen kennen. Sie können diese anwenden, um mobile Anwendungen zu entwerfen, anzupassen und weiter zu entwickeln. |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Inhalte** | **Kennen & Ver­ste­hen** | **Anwen­den** | **Analy­sieren & Bewer­ten** | **Synthe­tisieren** | | Technologieentwicklung mobiler Anwendungen | X | X |  |  | | Location-based Services | X | X | X | X | | Kommunikation | X | X |  |  | | Sensorik | X | X | X |  | | API-Programmierung mobiler Computer | X | X |  |  | | Track & Trace | X | X | X | X |   \_ | |
| Modulinhalte | Konzepte und Technologien für die Entwicklung avancierter mobiler  Anwendungen. Dabei stehen Kontextbezug und Kommunikation im  Vordergrund der Betrachtung. Folgende Themen werden behandelt:   * Location-based Services: Verwendung von unterschiedlichen Lokalisierungsdiensten mit verschiedenen Eigenschaften, Dienste zur  Visualisierung von Geodaten, Verwaltung von Geodaten, Geofencing, Location-based Social Networking * Kommunikation mobiler Anwendungen: Bluetooth, NFC, http u. a. * Erhebung von Umgebungsdaten über die Sensorschnittstellen * Location-based Social Networking: Matchmaking und Communication * Mining in mobilen sozialen Netzwerken * Track & Trace-Anwendungen: Erhebung von Positions- und Umgebungsdaten per Sensorik, Sammlung und Verwaltung der Daten, automatische Situationsüberwachung und -erkennung |
| Lehrformen | Vorlesung (2 SWS), Übung (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzung bestehen nicht. Zur erfolgreichen Teilnahme sind Kenntnisse zur Programmierung mit Java und Android-Kenntnisse erforderlich. |
| Literatur | Bill Philips, Chris Stewart, Brian Hardy, Kristin Marsiciano, Brian Gardner Android – Programming, Big Nerd Ranch Guide (4th Edition), 2019  Thomas Künneth, Android 11: Das Praxisbuch für Entwickler. Apps entwickeln mit Android Studio 4 und KotlinPress, Rheinwerk Computing, 2020  Android Sensor Programming By Example, Packt Publishing, 2016 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand / Workload | Gesamt 150 Stunden: 60 Stunden Präsenzzeit, 45 Stunden Selbststudium, 45  Stunden Prüfungsvorbereitung |
| ECTS und Gewichtung | 5 CP (Anteil an der Gesamtnote 5/120) |
| Leistungsnachweis | Schriftliche Prüfungsleistung |
| Semester | 2. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester (4 SWS) |
| Art der Lehrveranstaltung | Pflichtveranstaltung aus dem Bereich Verteilte und Mobile Systeme |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Computational Intelligence |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Martin Golz |
| Qualifikationsziele | Die Studierenden sollen in der Lage sein,   * Problemstellungen der computerbasierten Intelligenz zu analy­sieren, * Lerntheoretische Grundlagen zu kennen und zu verstehen, * Prozessketten der adaptiven Datenanalyse zu konzipieren, * Klassifikationsmethoden zu verstehen und anzuwenden, * Methoden der multivariaten Regression zu verstehen und anzuwenden, * Neuroinformatische Methoden zu kennen und zu analysieren, * Tiefe Lernarchitekturen zu kennen und zu analysieren. |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen, Ver­ste­hen** | **Anwen­den** | **Analy­sieren, Bewer­ten** | **Synthe­tisieren** | | Grundlagen der statistischen Inferenz | X |  |  |  | | Prozessketten der adaptiven Datenanalyse | X | X | X |  | | Statistische Lerntheorie | X |  |  |  | | Multivariate Regressions­analyse | X | X | X |  | | Lineare und Kernfunktions-Diskriminanz­analyse | X | X | X |  | | Stützvektor-Methode | X | X | X |  | | Lineare und nichtlineare adaptive Filter | X | X | X |  | | Neuronale Netze | X | X | X |  | | Tiefe Lernarchitekturen | X | X | X |  |   \_ | |
| Modulinhalte | 1. Einführung    1. Fünf Typen der statistischen Inferenz    2. Typische Anwendungsbeispiele    3. Prozesskette 2. Statistische Lerntheorie    1. Empirische Risikominimierung    2. Wahrscheinlich annähernd korrektes Lernen (PAC-Lernen)    3. Verallgemeinertes, agnostisches Lernmodell    4. Lernen mit gleichmäßiger Konvergenz    5. Verzerrungs-Komplexitäts-Kompromiss    6. Vapnik-Chervonenkis-Dimension    7. Strukturelle Risikominimierung 3. Multivariate, lineare Regressionsanalyse    1. Einführung    2. Modell    3. Maximum-Likelihood-Prinzip    4. Prinzip der maximalen a-posteriori-Wahrscheinlichkeit 4. Lineare Diskriminanzanalyse (LDA)    1. Einführung    2. Multi-Klassen-LDA    3. Kleinste-Quadrate-LDA    4. Fisher-LDA    5. Quadratische Diskriminanzanalyse 5. Kernfunktions-Diskriminanzanalyse    1. Einführung    2. Theorem von Cover    3. Duale Darstellung    4. Erzeugung von Kernfunktionen    5. Radiale-Basisfunktions-Netzwerke    6. Rekursive-Kleinste-Quadrate-Minimierung    7. Gauß-Prozesse    8. Anwendungsbeispiele 6. Adaptive Filter    1. Lineare, adaptive Filter       1. Kleinste-Quadrate-Algorithmus (KQ)       2. Rekursiver KQ-Algorithmus       3. Erweiterter, rekursiver KQ-Algorithmus    2. Nichtlineare, adaptive Filter       1. Der reproduzierende Kernfunktions-Hilbert-Raum (RKHR)       2. Kernfunktions-KQ-Filter    3. Anwendungsbeispiele 7. Boosting-Methoden    1. Starke und schwache Lernfähigkeit    2. Adaptive Boosting-Methoden 8. Künstliche Neuronale Netze    1. Neuron, Perzeptron, mehrschichtiges Perzeptronen-Netz    2. Vorwärtsgerichtete neuronale Netze, Lernregeln    3. Eigenschaften der Fehlerrückkopplung 9. Tiefe Lern-Strukturen    1. Tiefe Auto-Kodierungs-Netze    2. Netze vom Typ Langes Kurzzeit-Gedächtnis    3. Tiefe Faltungsnetze |
| Lehrformen | Tafelvorlesung mit   * Lückenskripten * Rechnerbasierter Präsentation * Demonstrationsprogrammen   Übungen im Computer-Pool   * Programmierung mit MATLAB * Klärung offener Fragen |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Grundkenntnisse in  Linearer Algebra, Analysis, Statistik, Wahrscheinlichkeitsrechnung |
| Literatur | Vapnik (1982) Estimation of dependences based on empirical data. Springer  Vapnik (1998) Statistical Learning Theory. Wiley  Duda, Hart, Stork (2001) Pattern classification. Wiley  Bishop (2006) Pattern recognition & machine learning. Springer  Mohri, Rostamizadeh (2012) Foundations of machine learning. MIT press  Nielsen (2015) Neural networks and deep learning. Determination press |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science,  Master-Studiengang Wirtschaftsinformatik und Digitale Transformation |
| Arbeitsaufwand / Workload | 150 Stunden, davon 60 Präsenzstunden und 90 Stunden Selbststudium |
| ECTS und Gewichtung | 5 Kreditpunkte, Gewichtung der Note in der Gesamtnote: 5/120 |
| Leistungsnachweis | Schriftliche Prüfung (120 Minuten) |
| Semester | 2. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Agile Software Development |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Englmeier |
| Qualifikationsziele | **Kennen / Verstehen:** Die Studierenden erlernen grundlegende Konzepte und Methoden der Agilen Softwareentwicklung. Auf den Kenntnissen in Projektmanagement aufbauend werden Grundlagen des agilen Manage­ments von Projekten gelernt. Dabei wird insbesondere auf die SCRUM-Methodik eingegangen.  **Anwenden:** Die Studierenden lernen anhand von Praxisbeispielen, wie Me­thoden und Konzepte der Agilen Softwareentwicklung angewendet wer­den.  **Analysieren / Bewerten:** In der Lehrveranstaltung werden intensiv agile Konzepte wie “self-empowered teams” und “continuous improvement” dis­ku­tiert. Die Studierenden reflektieren diese theoretisch erlernten Me­thodi­ken und gelangen so zu einer Bewertung von deren Praxis­taug­lich­keit.  **Synthetisieren:** In Diskussionen wird auf eigenverantwortliche Organi­sa­tion der Projektarbeit eingegangen und auch exploratives Lernen expli­zit angesprochen. Die Studierenden werden dabei angeregt und unterstützt neue Ideen zu entwickeln*.* |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Inhalt | Kennen | Verstehen | Anwen­den | Analy­sieren | Bewerten | Synthe­tisieren | | Grundlagen des Pro­jekt­managements | X | X | X | X |  |  | | Agile Prinzipien | X | X | X | X |  |  | | SCRUM | X | X | X | X | X | X | | Agile Lifecycle | X | X | X | X | X | X | | Performanzmessung | X | X | X | X | X | X | | |
| Modulinhalte | 1. Grundlagen Projektmanagement (Vertiefung ausgewählter Bereiche)    1. Qualitätsmanagement    2. Risikomanagement    3. Personalmanagement    4. Stakeholder-Management    5. Kommunikationsmanagement    6. Integrationsmanagement 2. Agile Prinzipien 3. SCRUM    1. Grundlegende Konzepte (User Stories, Iteration, Sprints, Back­logs, …)    2. Rollen und Teambildung    3. Kommunikation 4. Agile Lifecycle    1. Phasenmodell    2. Release-Planung 5. Performanzmessung |
| Lehrformen | Vorlesung (2 SWS) unterstützt mit multimedialen Inhalten, die von ACM angeboten werden. Workshops, Kooperation im Team (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Zur erfolgreichen Teilnahme sind grundlegende Kenntnisse zu Grundlagen des Projektmanagements erforderlich. |
| Literatur / multimediale Lehr- und Lernprogramme | Project Management Institute: Agile Practice Guide (2017) |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | 150 Stunden, davon 60 Präsenzstunden und 90 Stunden Selbststudium |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP, Gewichtung der Note in der Gesamtnote: 5/120 |
| Leistungsnachweis | Schriftliche Prüfung: 90 Minuten |
| Semester | 2. Semester |
| Häufigkeit des Angebots | Einmal pro Studienjahr (Sommersemester) |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Service oriented Networks |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Heinz-Peter Höller |
| Qualifikationsziele | Die Studierenden   * erwerben vertiefte Kenntnisse über die Anforderungen multimedialer Datenströme an Datenübertragungsnetze, * sind in der Lage Qualitätsdefizite im Netzdienst mit den Netzeigen­schaf­ten und den Verkehrscharakteristiken zu korrelieren, * erwerben vertiefte Kenntnisse zum Begriff und zum Anspruch von Quality-of-Service (Dienstgüte) und Überlastkontrolle, * sind in der Lage, Ansätze des Quality-of-Service und der Überlast­behandlung in den Netztechnologien der OSI-Schichten zu identi­fizieren und zu analysieren und * sind in der Lage bei der Programmierung verteilter Systeme auf die unterschiedlichen Ausprägungen von Dienstgüte zu reagieren. |
| Modulinhalte | Moderne Anwendungen und ihre Anforderungen an Datenübertragungsnetze  Überlastbehandlung  Dienstgüte / Quality-of-Service  Dienstgüteklassen  Signalisierung  Verkehrsmanagement  Puffermanagement  Überlastbehandlung in TCP  Dienstgüte in Schicht-2-Technologien (ATM, LAN, MPLS, LAN)  Dienstgüte im Internet  Integrated Service  Differentiated Service  Neuere Transportschichtprotokolle (RTP, RTCP, DCCP) |
| Lehrformen | Seminaristische Vorlesung  Übungen zur Vertiefung des Stoffes |
| Voraussetzungen für die Teilnahme | Keine formalen Voraussetzungen; Gute Kenntnisse auf dem Gebiet der Datenkommunikation in Rechnernetzen werden erwartet. |
| Literatur / multimediale Lehr-und Lernprogramme | Badach, A., Voice over IP Die Technik, Hanser, München, 2005.  Badach, A. & Hoffmann, E., Technik der IP-Netze Internet Kommunikation in Theorie und Einsatz, 4. Auflage, München, 2019  Braun, T. & Zitterbart, M.. Hochleistungskommunikation, Band 2: Transportdienste und –protokolle.Oldenbourg Verlag 1996.  Kurose, J.F. & Ross, K.W. Computernetzwerke.Pearson Studium, München 2008.  Lu, G., Communication and Computing for Distributed Multimedia Systems.Artech House 1996.  Schmitz, R., Kiefer, R., Maucher, J., Schulze, J. &Suchy, T. Kompendium Medieninformatik. Mediennetze. Springer 2006.  Shin, J., Lee, D.C. &Kuo, C.-C.J., Quality of Service for Internet Multimedia,Prentice Hall 2004  Siegel, E.D., Quality of Service. Solutions for the Enterprise.Wiley 2000  Wang, Z., Internet QoS Architectures and Mechanisms for Quality of Service, San Francisco, 2001 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Gesamt 150 Stunden. Kontaktzeit/Präsenzstudium: 60 Stunden (=15 \* (3 SWS Vorlesungen + 1 SWS Übung)), Selbststudium: 45 Stunden, Prüfung und Prüfungsvorbereitung: 45 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP, Anteil an der Gesamtnote: 5/120 |
| Leistungsnachweis | Schriftliche Prüfung, 120 Minuten |
| Semester | 2. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester (4 SWS) |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Web Applications |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Erwin Neuhardt |
| Qualifikationsziele | Die Studierenden kennen die Struktur und den Aufbau einer Webanwendung. Sie kennen die Aufgaben, die bei der Entwicklung einer Webanwendung gelöst werden müssen. Sie können eine Webanwendung mit zwei unterschiedlichen Frameworks auf Basis der Programmiersprache Java entwickeln. |
| Modulinhalte | Elemente einer Webanwendung:   * HTML, CSS, HTTP, Model-View-Controller   Webanwendungen mit Spring MVC und Vaadin   * benötigte Softwareartefakte * Routing * Formulare erzeugen * Formulardaten einlesen * Konvertierung und Validierung * Internationalisierung * Ausgabe von Listen |
| Lehrformen | Vorlesung (3 SWS) und Übung am PC (1 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Zur erfolgreichen Teilnahme sind Kenntnisse der Programmiersprache Java erforderlich. |
| Literatur / multimediale Lehr-und Lernprogramme | Rod *J*ohnson, Rod et al.: Spring Framework Reference Documentation, Online unter docs.spring.io  w/o author: Spring Guides, Online unter spring.io/guides  w/o author: Apache FreeMarker, Online unter freemarker.org  w/o author: Bootstrap, Online unter getbootstrap.com  w/o author: Vaadin Documentation, Online unter vaadin.com |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Präsenzzeit: 60 Stunden  Selbststudium: 60 Stunden  Prüfung und Prüfungsvorbereitung: 30 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 ECTS  Gewichtung: 5/120 |
| Leistungsnachweis | Schriftliche Prüfungsleistung (Klausur am PC, 120 Minuten) |
| Semester | 3. Semester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | 1 Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Seminar |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr.-Ing. Thomas Heimrich |
| Qualifikationsziele | Die Studierenden erwerben Fähigkeiten, sich mit neuen wissenschaftlichen Themen aus ihrem Fachgebiet auseinanderzusetzen. Sie wenden Recherche­techniken an, analysieren wissenschaftliche Quellen, erarbeiten Exzerpte und fertigen eine wissenschaftliche Ausarbeitung zu einem abgesteckten Thema an. Hierbei sind wissenschaftliche Standards einzuhalten. Abschließend wird die Ausarbeitung in einem Kurzvortrag vorgestellt und zur Diskussion gestellt. |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen** | **Verstehen** | **Anwenden** | **Analysieren** | **Bewerten** | **Synthetisieren** | | Selbststudium | x | x | x | x | x |  | | Recherchen | x | x | x | x | x |  | | Wissenschaft­li­che Aus­ar­bei­tung | x | x | x | x | x |  | | Wissenschaftl. Vortrag mit Diskussion | x | x | x | x | x |  | | |
| Modulinhalte | Techniken wissenschaftlichen Arbeitens, insbesondere das Recherchieren und Exzerpieren. Strukturiertes Verfassen wissenschaftlicher Texte zu aktuellen The­men aus dem Fachgebiet, die noch kein Lehrbuchinhalt sind. Wissenschaftliches Präsentieren und Diskutieren. |
| Lehrformen | Seminar (2 SWS) |
| Voraussetzungen für die Teilnahme | Keine |
| Literatur / multimediale Lehr- und Lernprogramme | Ausgewählte Quellen zur Aufarbeitung aktueller Themen |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand/  Gesamt-Workload | 150 Stunden: 30 Stunden Präsenzzeit, 100 Stunden Selbststudium, 20 Stunden Prüfungsvorbereitung |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP, Anteil an der Gesamtnote: 5/120 |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | 3. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Project |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr.-Ing. Thomas Heimrich |
| Qualifikationsziele | Die Studierenden erwerben die Fähigkeit, ein spezielles und aktuelles Thema der Forschung in einem Fachgebiet für die praktische Umsetzung auszuwählen und abzugrenzen. Sie analysieren die Schritte der softwaretechnischen Umset­zung im Rahmen der im Studium erlernten Konzepte und Technologien. Hierfür sind geeignete Entwurfs- und Implementationstechniken anzuwenden. Es wer­den ei­nige Prinzipien des Projektmanagements im praktischen Einsatz kennen­ge­lernt. Fähigkeiten zur Team-Arbeit, zur Kommunikation und zur Reflektion im Pro­zess der Softwareentwicklung werden erworben. |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen** | **Ver­stehen** | **Anwen­den** | **Analy­sieren** | **Bewer­ten** | **Syntheti­sieren** | | Analyse einer wissenschaftlichen Problemstellung  der praktischen Informatik | x | x | x | x | x |  | | Softwareentwicklungsprozess | x | x | x | x | x |  | | Softwaretechnische Problemlösung | x | x | x | x | x | x | | Multinationale Teamarbeit | x | x | x | x | x |  | | Selbstorganisation | x | x | x | x | x |  | | |
| Modulinhalte | Durchführung eines abgesteckten Software-Entwicklungsprojektes mit Team-An­forderung in allen Phasen. Es wird eine zur softwaretechnischen Umsetzung ge­eig­nete, wissenschaftliche Problemstellung aus dem Gebiet Smart Systems aus­ge­wählt, analysiert und die Projektziele spezifiziert. Die praktische Umsetzung erfolgt weitgehend eigenständig im kleinen Team.. |
| Lehrformen | Vorlesung (1 SWS), Projekt |
| Voraussetzungen für die Teilnahme | Keine |
| Literatur / multimediale Lehr- und Lernprogramme | Ausgewählte Quellen zu einem ausgewählten wissenschaftlichen Thema im Rahmen der Literatur-Recherche. |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | 150 Stunden: 30 Stunden Präsenzzeit, 100 Stunden Selbststudium, 20 Stunden Prüfungsvorbereitung |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (Anteil an der Gesamtnote 5/120) |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | 3. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester (4 SWS) |
| Art der Lehrveranstaltung | Pflichtmodul |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Master Thesis |
| Modulverantwortlicher / Modulverantwortliche | Betreuer der Master-Arbeit |
| Qualifikationsziele | Die Studierenden erwerben folgende Fähigkeiten:   * Analyse wissenschaftliches Problemstellungen * Strukturierung und Eingrenzung eines Themas * Entwicklung einer Gliederung * Entwicklung und Bewertung wissenschaftlicher Lösungswege * Kennen und Einhalten formaler und wissenschaftliche Kriterien einer Abschlussarbeit * Leisten eines klar erkennbaren, eigenen wissenschaftlichen Anteils * Schriftliches wissenschaftliches Argumentieren |
| Modulinhalte | In der Master-Arbeit sollen Studierende zeigen, dass die während des Stu­diums erworbenen Kenntnisse und Fähigkeiten erfolgreich für gestellte wis­sen­schaft­li­che Ziele eingesetzt werden können. Diese sollen eigenverantwortlich, selbstän­dig und problemorientiert erreicht werden. Einige der Ziele sollen in ei­nem An­wendungsprojekt in die Praxis umgesetzt werden. Es sind eigen­stän­di­ge wis­sen­schaftliche Anteile zu erarbeiten und zu bewerten. Das Thema wird ausgegeben und betreut von einem Prüfer gemäß § 18 Absatz 1 der Prüfungsordnung. Für die Themenstellung können die Studierenden dem Prüfer Vorschläge unterbrei­ten. Gemeinsam mit dem Betreuer werden Thema, Aufgabenstellungen und der Umfang der schriftlichen Ausarbeitung so begrenzt, dass eine Bearbeitung in 24 Wochen möglich ist.  Die Betreuer leiten die Studierenden zum wissenschaftlichen Arbeiten an.  Die Master-Arbeit wird nach strengen formalen und wissenschaftlichen Kriterien schriftlich ausgearbeitet. Dabei ist besonders auf die Maßstäbe gemäß der Sat­zung der Hochschule Schmalkalden zur Sicherung guter wissenschaftlicher Pra­xis zu achten. Falls wissenschaftliche Daten gewonnen oder verarbeitet werden, sind die Leitlinien der Hochschule Schmalkalden zum Umgang mit Forschungs­daten zu berücksichtigen. |
| Lehrformen | Projekt |
| Voraussetzungen für die Teilnahme | Anerkennung des Master-Themas durch den Betreuer. Die Ausgabe der Masterarbeit kann erfolgen, wenn bis auf einen Umfang von höchstens 10 Kreditpunkten alle Prüfungsleistungen bestanden sind. |
| Literatur / multimediale Lehr-und Lernprogramme | Abhängig vom Thema  Grundlegende Werke zum wissenschaftlichen Arbeiten sind, z.B.:   * Helmut Balzert, C. Schäfer, M. Schröder, U. Kern: Wissenschaftliches Arbeiten, 2. Aufl., 2011 * Werner Sesink, Einführung in das wissenschaftliche Arbeiten inklusive E-Learning, Web-Recherche, digitale Präsentation u.a., 9. Aufl., 2012 * Theo Hug, Gerald Poscheschnik: Empirisch Forschen: Über die Planung und Umsetzung von Projekten im Studium, 2010 |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand / Gesamtworkload | Kontaktzeit: 10 Stunden, Selbststudium: 800 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 27 Leistungspunkte Gewichtung der Note in der Gesamtnote: 27/120 |
| Leistungsnachweis | Schriftliche Prüfungsleistung |
| Semester | 4. Semester |
| Häufigkeit des Angebots | Jedes Semester |
| Dauer | Die Bearbeitungszeit für die Master-Arbeit beträgt 24 Wochen. |
| Art der Lehrveranstaltung | Pflichtmodul |

|  |  |
| --- | --- |
| Modulname | Master Colloquium |
| Modulverantwortlicher / Modulverantwortliche | Betreuer der Master-Arbeit |
| Qualifikationsziele | Die Studierenden erwerben folgende Fähigkeiten:   * Konzipieren einer wissenschaftlichen Präsentation * Präsentieren eines wissenschaftlichen Projekts * Wissenschaftliches Argumentieren * Diskutieren von Fragen, die Prüfer und Zuhörer stellen |
| Modulinhalte | * Ausgewählte Inhalte der Master-Arbeit sind im Rahmen des Kolloquiums zu präsentieren und zur Diskussion zu stellen. * Ein wissenschaftlicher Vortrag ist selbständig zu erarbeiten und vor einem hochschulöffentlichen Fachpublikum zu halten * Hierbei steht das freie Reden und Argumentieren im Fokus. * Die aufgestellten wissenschaftlichen Hypothesen sind in diesem Rahmen hochschulöffentlich zu vertei­di­gen. * Es sind auch Inhalte anderer Autoren darzulegen und von den eigenen ab­zugrenzen   Der Nutzen der Arbeit soll erörtert werden |
| Lehrformen | Projekt |
| Voraussetzungen für die Teilnahme | Bewertung der Master-Arbeit mit mindestens „ausreichend“. |
| Literatur / multimediale Lehr-und Lernprogramme | Abhängig vom Thema  Grundlegende Werke zum wissenschaftlichen Arbeiten sind, z.B.:   * Helmut Balzert, C. Schäfer, M. Schröder, U. Kern: Wissenschaftliches Arbeiten, 2. Aufl., 2011 * Werner Sesink, Einführung in das wissenschaftliche Arbeiten inklusive E-Learning, Web-Recherche, digitale Präsentation u.a., 9. Aufl., 2012 * Theo Hug, Gerald Poscheschnik: Empirisch Forschen: Über die Planung und Umsetzung von Projekten im Studium, 2010 |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Kontaktzeit: 2 Stunden; Selbststudium: 88 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 3 Leistungspunkte Gewichtung der Note in der Gesamtnote: 3/120 |
| Leistungsnachweis | Mündliche Prüfungsleistung |
| Semester | 4. Semester |
| Häufigkeit des Angebots | Jedes Semester |
| Dauer | 60 Minuten |
| Art der Lehrveranstaltung | Pflichtmodul |

# Wahlbereich I

|  |  |
| --- | --- |
| Modulname | Semantic Technologies in Distributed Systems |
| Modulverantwortlicher / Modulverantwortliche | Lehrbeauftragter |
| Qualifikationsziele | Die Studierenden sind in der Lage, Konzepte und Technologien aus dem Bereich „Intelligente Middleware“ zu verstehen und anzuwenden. Bestehende Lösungen können Sie anhand der vermittelten Inhalte analysieren und bewerten. |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen** | **Ver­stehen** | **Anwen­den** | **Analy­sieren** | **Bewer­ten** | **Synthe­tisieren** | | Ontologien | x | x | x | x | x |  | | SOA-Plattformen, Enterprise Service Bus | x | x | x | x | x |  | | Situationsüberwachung, Complex Event Processing | x | x | x | x | x |  | | |
| Modulinhalte | Konzepte und Technologien aus dem Bereich Intelligente Middleware   * Ontologien: Konzepte zur semantischen Datenverarbeitung, ETL-Strecken, Ontologiebeschreibungssprachen, Knowledge Bases und Inferenzmechanismen * Middleware-Plattformen und Architekturen: Enterprise Service Bus, Serviceorientierte Architekturen * Situationserkennung: ereignisorientierte Architekturen, ereignisorientierte Programmierung, Complex Event Processing |
| Lehrformen | Vorlesung (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Zur erfolgreichen Teilnahme sind Kenntnisse zur Programmierung mit Java erforderlich. |
| Literatur / multimediale Lehr- und Lernprogramme | Franz Baader, Deborah L. McGuinness, Daniele Nardi, Peter F. Patel-Schneider, The Description Logic Handbook (2nd Edition), Cambridge University Press, 2010  David Chappell, Enterprise Service Bus: Theory in Practice, O’Reilly 2004  David Luckham, The Power of Events: An Introduction to Complex Event Processing in Distributed Systems, Addison-Wesley 2002 |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | 150 Stunden: 30 Stunden Präsenzzeit, 100 Stunden Selbststudium, 20 Stunden Prüfungsvorbereitung |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (Anteil an der Gesamtnote 5/120) |
| Leistungsnachweis | Schriftliche Prüfungsleistung |
| Semester | 1. und 3. Semester (Wintersemester) |
| Häufigkeit des Angebots | Jedes zweite Jahr |
| Dauer | Ein Semester (2 SWS) |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich I) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | IT Security - Advanced Chapters |
| Modulverantwortlicher / Modulverantwortliche | Prof. Ralf C. Staudemeyer |
| Qualifikationsziele | In diesem Kurs verbessern die Studierenden gezielt ihre Fähigkeit sich in aktuellen Entwicklungen im Bereich IT-Sicherheit und Datenschutztechnologien erfolgreich einzuarbeiten. |
| Modulinhalte | Sie erlernen dabei Recherchetechniken um sich an vorgegebene Themen heranzuarbeiten. Neben einer ausführlichen Literaturrecherche soll ein wissenschaftlicher Eigenbeitrag geleistet werden. Die Ergebnisse der Arbeit werden in Form eines Konferenzbeitrags präsentiert. Dies beinhaltet einen Vortrag und eine wissenschaftliche Veröffentlichung. |
| Lehrformen | Vorlesung (2 SWS), Übungen (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für eine erfolgreiche Teilnahme sind Kenntnisse zu Grundlagen der IT-Sicherheit erforderlich. |
| Literatur / multimediale Lehr- und Lernprogramme | Ausgewählte Quellen zur Aufbereitung aktueller Themen |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | Insgesamt: 150h; Kontaktzeit/Präsenzstudium: 60h; Selbststudium incl. Prüfungsvorbereitung: 90h |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP, Anteil an der Gesamtnote: 5/120 |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | 3. Semester (Wintersemester) |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester (4 SWS) |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich I) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Text Analysis and Data Search |
| Modulverantwortlicher /  Modulverantwortliche | Prof. Dr. Englmeier |
| Qualifikationsziele | **Kennen / Verstehen:** Die Studierenden erlernen grundlegende Konzepte und Methoden der semantischen Datenanalyse und des Information Retrieval. Da­bei wird insbesondere auf die Gestaltung von Suchmaschinen eingegangen.  **Anwenden:** Die erlernten Methoden der Datenanalyse werden mit gängigen Werkzeugen (z.B. Apache Lucene) umgesetzt, die für die Konstruktion von Such­maschinen wertvoll sind.  **Analysieren / Bewerten:** In der Teamarbeit des Projektes setzen die Studie­ren­den ihr Theoriewissen in der Gestaltung eines konkreten Text Mining-Tools um. Sie erschließen sich dadurch die Gestaltungsvielfalt in der Konstruktion von Analysefunktionen für Texte und für die Suche. Sie reflektieren ihre Ansät­ze und deren Praxistauglichkeit im Rahmen ihrer Teamarbeit.  **Synthetisieren:** Das Ergebnis des Kurses manifestiert sich in einem kurs­über­greifenden Projekt, das die Entwicklung eines Text-Mining-Tools oder eines Tools mit speziellen Suchfunktionen beinhaltet. Die Anwendungserstellung wird dabei in kleinere Arbeitspakete zerlegt. Jedes Team (zwei bis drei Studierende) übernimmt dabei ein Arbeitspaket, organisiert seine individuellen Aufgaben und bringt sich in das Management des Gesamtprojektes ein.  Die eigenverantwortliche Organisation der Projektarbeit schließt auch explo­ra­ti­ves Lernen ein. Die Studierenden werden dabei angeregt und unterstützt, sich neue Methodiken oder Werkzeuge in Eigeninitiative anzueignen, sofern dies ihre individuelle Projektarbeit erfordert. |
| Modulinhalte | 1. Grundlagen des Information Retrieval (IR)   * Grundlegende IR Konzepte * Regular Expressions * XML   2. Retrieval models & evaluation  3. Apache Lucene/Python  4. Indexing   * Tokenisation * Stopwords * Stemming * Synonyms   5. Query matching   * Query vectors * Matching models |
| Lehrformen | Vorlesung, Workshops, Kooperation im Team |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für eine erfolgreiche Teilnahme sind Kenntnisse zu Grundlagen der Programmierung erforderlich. |
| Literatur / multimediale Lehr- und Lernprogramme | Baeza-Yates, R.; Ribeiro-Neto, B.: “Modern Information Retrieval”, ACM Press, New York, 1999.  McCandless, M. et al: “Lucene in Action”, Second Edition, Manning, Stamford, 2010  Anwendungsbeispiele zu Suchmaschinen in der Praxis |
| Verwendbarkeit | Master-Studiengang Wirtschaftsinformatik und Digitale Transformation,  Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Kontaktzeit/Präsenzstudium: 45 Stunden; Selbststudium 45 Stunden; Praxisarbeitszeit: 30 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 von 120 ECTS |
| Leistungsnachweis | Alternative Prüfungsleistung (Projektarbeit) |
| Semester | 3. Semester |
| Häufigkeit des Angebots | Einmal pro Studienjahr (Wintersemester) |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich I) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Selected Chapters Functional Programming |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Erwin Neuhardt |
| Qualifikationsziele | Die Studierenden kennen die Konzepte der funktionalen Programmierung und ihre Umsetzung in Scala. Sie können diese Konzepte in einem konkreten Anwendungskontext umsetzen. |
| Modulinhalte | Elemente der funktionalen Programmierung   * Funktionen, nicht-rekursiv und rekursiv * Fehlerbehandlung in funktionalen Programmen * Listen * Case Klassen * Funktoren und Monaden * Ein- und Ausgabe in funktionalen Programmen * Eigenschaftsbasierte Tests |
| Lehrformen | Vorlesung (2 SWS) und Übung am PC (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für eine erfolgreiche Teilnahme sind Kenntnisse zu Grundlagen der Programmiersprache Java erforderlich. |
| Literatur / multimediale Lehr-und Lernprogramme | Alvin Alexander: Functional Programming, Simplified, [CreateSpace Independent Publishing Platform], 2017  Paul Chiusano, Runar Bjarnason: Functional Programming in Scala, Manning, 2014  Jason Swartz: Learning Scala, O'Reilly, 2015 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Präsenzzeit: 60 Stunden  Selbststudium: 60 Stunden  Prüfung und Prüfungsvorbereitung: 30 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 ECTS  Gewichtung: 5/120 |
| Leistungsnachweis | Klausur am PC (120 Minuten) |
| Semester | 2. Semester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | 1 Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich I) |
| Besonderes |  |

# Wahlpflichtbereich II

|  |  |
| --- | --- |
| Modulname | 3D Content Creation |
| Modulverantwortlicher / Modulverantwortliche | Prof. Hartmut Seichter, PhD |
| Qualifikationsziele | Studierenden erlernen ein breites Spektrum an computergrafischen Grundlagen zur Erstellung von 3D Inhalten für Spiele, Animationen und Visualisierungen. Es wird vertieftes, praxisorientiertes Wissen im Bereich der 3D Modellierung Animation und Compositing vermittelt. Darüber hinaus lernen die Studierenden Grundlagen der Ästhetik, Komposition des Raumes um mit computergrafischen Mitteln visuell anspruchsvolle Inhalte zu konzipieren, planen und umzusetzen. Es werden technologisch neueste Methoden eingesetzt um die verschiedene Ausspielwege von 3D Inhalten an Projektarbeiten zu erlernen. |
| Modulinhalte | * Menschliche Wahrnehmung * Visuelle Kommunikation und Design Methodik * Szenengestaltung, Gestalt und Semiotik * Didaktik des Raums * 3D Modelle und Darstellung * Modellierungstechniken * Animationsprinzipien und techniken * Match-Moving und Motion-Capturing * Szenen und Projektmanagement * Bildsyntheseverfahren (Rasterization, Raytracing, Pathtracing und Radiosity) * Beleuchtungsmodelle, Texturierung, PBR workflow * Licht- und Kamerasetzung * Special FX und 3D Compositing |
| Lehrformen | Seminaristische Vorlesung mit Übung |
| Voraussetzungen für die Teilnahme | Formale Voraussetzung bestehen nicht. Für eine erfolgreiche Teilnahme sollte das Modul „Grundlagen der Computergrafik“ im Vorfeld belegt werden. |
| Literatur / multimediale Lehr-und Lernprogramme | * Seminare * Übungen * Blitzentwürfe * Video-Tutorien |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Digitales Marketing,  Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | * Seminar 2 SWS * Übungen 2 SWS * Workload: Präsenz (Vorlesung + Übungen) 120 h * Workload: Projektbearbeitung 30 h |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (5/120) |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | zweites Semester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Wahlpflichtbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Interactive Systems |
| Modulverantwortlicher /  Modulverantwortliche | Prof. Hartmut Seichter, PhD |
| Qualifikationsziele | Studierende lernen vertieftes Wissen um grafische Nutzeroberflächen zu analysieren und informierte Designentscheidungen zu treffen. Dabei werden auch Grundlagen der Wahrnehmung vermittelt. Studierende entwickeln anhand von Prototypen anwendbares Wissen Interaktionen zu messen und anhand von wissenschaftlichen Methoden zu bewerten. |
| Modulinhalte | * Wahrnehmung * Visuelle Gestaltung von Graphischen Nutzerschnittstellen * Applikationsdesign mit Fokus auf GUI Konzepte * Nutzerstudien * Evaluierungsmethoden mit interaktiven visuellen Systemen |
| Lehrformen | Seminaristische Vorlesungen und Übungen |
| Voraussetzungen für die Teilnahme | Formale Voraussetzung bestehen nicht. Für eine erfolgreiche Teilnahme sollte das Modul „Grundlagen der Statistik“ im Vorfeld belegt werden. |
| Literatur / multimediale Lehr-und Lernprogramme | * H5P Lernmodule, Lernforum und Übungen am PC * Auszug aus der Literaturliste:   + Card, Stuart K., Thomas P. Moran, and Allen Newell. The Psychology of Human-Computer Interaction. Repr. Mahwah, NJ: Erlbaum, 2008.   + Cooper, Alan. About Face: The Essentials of Interaction Design, 4th Edition. 4th edition. Indianapolis, IN: John Wiley and Sons, 2014.   + Dix, Alan, Janet Finlay, Gregory D Abowd, and Russell Beale. Human-Computer Interaction. Pearson Education, 2003   + Krug, Steve. Don’t Make Me Think, Revisited: A Common Sense Approach to Web Usability. Third edition. Berkeley, Calif.: New Riders, 2014.   + Nielsen, Jakob. Usability Engineering. Boston: Academic Press, 1993. |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | * Vorlesung / Lernforum 2 SWS * Übung 2 SWS * Workload: Präsenz (Vorlesung + Übungen) 120 h Projektbearbeitung 30 h |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (5/120) |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | Wintersemester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Wahlbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Image Processing 1 |
| Modulverantwortlicher /  Modulverantwortliche | Prof. Dr. Klaus Chantelau |
| Qualifikationsziele | * Die Studierenden können typische Problemstellungen bei der Entwick­lung von digitalen audio-visuellen Formaten zu analysieren * Die Studierenden verstehen die Funktionsweise der Grundlagen der Kom­pression von audio-visuellen Signalen. * Die Studierenden verstehen den Aufbau der wichtigsten digitalen audio-visu­ellen Standards zu verstehen (G7xx, mp3, GIF/PNG, JPEG, H26x, DV, MPEG1 u. 2 u. 4 ) * Die Studierenden können die wichtigsten mathematischen und algo­rith­mi­schen Methoden für die Implementierung von Kompressionsverfah­ren an­wen­den*.* |
| Modulinhalte | Farbräume, Filterprozesse, Fourier-, DCT-, und Wavelet-Transformation, Bild­seg­men­tierung, Entropiecodierung, Transformationscodierung, Prädiktions­co­die­rung, verlustbehaftete Codierung mit Quantisierungstechniken, Eigen­schaf­ten der menschlichen visuellen Informationsverarbeitung, Bewegungsprädiktion bei Kompressionsverfahren für Bildsequenzen. |
| Lehrformen | Vorlesung (3 SWS), Rechnerübungen (1 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für eine erfolgreiche Teilnahme sind Kenntnisse in Linearer Algebra und in der Programmierung erforderlich. |
| Literatur / multimediale Lehr-und Lernprogramme | Ohm, Jens Rainer „Digitale Bildcodierung“ Springer, ISBN 3-540-58579-6 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Kontaktzeit / Präsenzstudium: 60 Stunden; Selbststudium 45 Stunden, Prüfungsvorbereitung 45 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 ECTS Gewichtung: 5/120 ECTS |
| Leistungsnachweis | Schriftliche Prüfungsleistung, Klausur von 90 Minuten |
| Semester | 1. Semester |
| Häufigkeit des Angebots | Jährlich*.* |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Image Processing 2 |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Klaus Chantelau |
| Qualifikationsziele | * Die Studierenden können typische Problemstellungen bei der Verarbeitung von digitalen visuellen Signalen analysieren * Die Studierenden verstehen die wichtigsten mathematischen und algorithmischen Methoden für die Merkmalsextraktion und die Klassifikation und die 3D- Analyse von visuellen Signalen. * Die Studierenden können die wichtigsten mathematischen und algorithmischen Methoden für die Implementierung von Audio- und Bildanalyseverfahren anwenden. |
| Modulinhalte | Bildaufnahme und Beleuchtung, Bildwandlung (Vorder-Hintergrund-Trennung, Transformationen,...), Bildverbesserung (Filterungen, Segmentierung, Label­ling,...), Merkmalsextraktion, (Geometrie/Kontur-Deskriptoren, Textur-Deskripto­ren,...), 3D-Szenenanalyse, Klassifikation und Vermessung |
| Lehrformen | Vorlesung und Rechnerübungen |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. Für eine erfolgreiche Teilnahme sind Kenntnisse aus dem Modul „Image Processing 1“ erforderlich. |
| Literatur / multimediale Lehr-und Lernprogramme | „Handbuch zur Industriellen Bildverarbeitung“, FhG IRB Verlag,  ISBN 978-3-8167-7386-3  “Introduction to MPEG 7” - Manjunath, Salembier, Sikora Wiley, ISBN 0-471-48678-7 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Kontaktzeit/ Präsenzstunden: 30 Stunden (=15 \* (1 SWS Vorlesung + 1 SWS Übung)),Selbststudium: 45 Stunden, Prüfung/Prüfungsvorbereitung: 15 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 3 ECTS Gewichtung: 3/120 ECTS |
| Leistungsnachweis | Schriftliche Prüfungsleistung, Klausur von 90 Minuten |
| Semester | 3. Semester |
| Häufigkeit des Angebots | Jährlich*.* |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Media Production |
| Modulverantwortlicher / Modulverantwortliche | Prof. Dr. Klaus Chantelau |
| Qualifikationsziele | * Die Studierenden verstehen die fortgeschrittenen technischen Grundlagen moderner Kamera- und Studiotechnik * Die Kursteilnehmer kennen die Leistungsmerkmale moderner Video­pro­duk­tions- und Übertragungsformate wie RAW und H.264 * Die Studierenden kennen die fortgeschrittenen Grundlagen zum Verständnis von Farbräumen, der Farbkorrektur und dem Keying * Die Kursteilnehmer kennen die Problemstellung und die algorithmisch-technischen Lösungsansätze zur Kamera-Kalibrierung * Die Studierenden kennen die Problemstellung und die algorithmisch-technischen Lösungsansätze für das Tracking und das Videokompositing * Die Kursteilnehmer kennen die Problemstellung und die algorithmisch-technischen Lösungsansätze für 360 Grad- und Stereovideoproduktionen * Die Studierenden besitzen die praktischen Kompetenzen zur Produktion von audio-visuellen Medien und beherrschen insbesondere den Einsatz moder­ner Videokamera- und Beleuchtungstechnik, Audioaufnahmetechnik, Ani­ma­tions-, Audio- und Videopostproduktionsystemen (wie z.B. den Adobe CC-Produkten) sowie den Einsatz von Green-Screen- und Mixed-Reality-Techniken |
| Modulinhalte | * Studio- und Green-Screen-Technik für die Videoproduktion * Kamera-Kalibrierung, Tracking und Videokompositing * Colorgrading und Postproduktion * Praktische Umsetzung von Green-Screen-basierten Videoproduktionen*.* |
| Lehrformen | Vorlesungen (2 SWS) und Übungen (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzungen bestehen nicht. |
| Literatur/ multimediale Lehr-und Lernprogramme | Hasche, E., Ingwer, P., Games of Color - Moderne Bewegtbildproduktion, Springer Verlag, 2016 |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | Kontaktzeit/Präsenzstudium: 60 Stunden; Projektbearbeitung 45 Stunden, Prüfungsvorbereitung 45 Stunden |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 ECTS  Gewichtung: 5/120 ECTS |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | 2. Semester |
| Häufigkeit des Angebots | Einmal im Studienjahr |
| Dauer | Ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Virtual and Augmented Environments |
| Modulverantwortlicher / Modulverantwortliche | Prof. Hartmut Seichter, PhD |
| Qualifikationsziele | Studierende beschäftigen sich tiefgreifend mit Technologien der Mixed, Augmen­ted und Virtual Reality. Es werden der Stand der Technik im Seminar detailliert erörtert und es werden wissenschaftliche Diskussionen um die jeweiligen Tech­no­logien geführt. Anhand von heutigen VR/AR und MR Technologien setzen sich die Teilnehmer mit dem Stand der Technik und den Möglichkeiten der Umsetzung von immersiven Medienprodukten auseinander. |
| Modulinhalte | * Tracking Technologien * Display Technologien in VR und AR * Interaktionsgeräte * Interaktionstechniken * Echtzeitrendering-Methoden * Stereorendering * Compositing * Evaluierungsmethoden in VR/AR * Human Factors in VR/AR |
| Lehrformen | Seminaristische Vorlesung mit Übungen |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzung bestehen nicht. Für eine erfolgreiche Teilnahme sollte das Modul „Computer Graphics“ im Vorfeld belegt werden. |
| Literatur/ multimediale Lehr-und Lernprogramme | * Folien, Vorlesungen, Vorträge * Auszug aus der Literaturliste:   + Drummond, T., and R. Cipolla. “Real-Time Visual Tracking of Complex Structures.” IEEE Transactions on Pattern Analysis and Machine Intelligence 24, no. 7 (July 2002): 932-46. https://doi.org/10.1109/TPAMI.2002.1017620.   + LaViola, Joseph J., Ernst Kruijff, Ryan P. McMahan, Doug A. Bowman, and Ivan Poupyrev. 3D User Interfaces: Theory and Practice. Second edition. Addison-Wesley Usability and HCI Series. Boston: Addison-Wesley, 2017.   + Stanney, Kay, Cali Fidopiastis, and Linda Foster. “Virtual Reality Is Sexist: But It Does Not Have to Be.” Frontiers in Robotics and AI 7 (January 31, 2020). <https://doi.org/10.3389/frobt.2020.00004>.   + Wloka, Mathias M. “Interacting with Virtual Reality.” In In Virtual Environments and Product Development Processes, edited by J. Rix, S. Haas, and J. Teixeira. Chapman and Hall, 1995. |
| Lehrbriefautor |  |
| Verwendbarkeit | Master-Studiengang Applied Computer Science |
| Arbeitsaufwand /  Gesamtworkload | * Seminar 2 SWS * Übungen 2 SWS * Workload: Präsenz (Vorlesung + Übung) 120 h, Projektbearbeitung 30 h |
| ECTS und Gewichtung der Note in der Gesamtnote | 5 CP (5/120) |
| Leistungsnachweis | Alternative Prüfungsleistung |
| Semester | drittes Semester |
| Häufigkeit des Angebots | einmal im Studienjahr |
| Dauer | ein Semester |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich II) |
| Besonderes |  |

|  |  |
| --- | --- |
| Modulname | Human Machine Interaction |
| Modulverantwortlicher /  Modulverantwortliche | Prof. Dr. Englmeier |
| Qualifikationsziele | **Kennen / Verstehen:** Die Studierenden erlernen kognitionswissenschaft­liche Grundlagen der Mensch-Maschine-Kommunikation, die den theoreti­schen Grundstock für das erfolgreiche Design von Benutzerinteraktionen liefern. Sie setzen sich insbesondere mit der Nutzer-und Aufgabenanalyse auseinander.  **Anwenden:** Das theoretische Wissen leitet die Studierenden im Entwurf von User Stories, die wiederum als Blaupausen für die Nutzerinteraktion dienen. Im Vordergrund steht dabei die Umsetzung der Interaktion in unter­schiedlichen Umgebungen, z.B. in den Beschreibungssprachen für Benut­zer­schnittstellen in mobilen Anwendungen (XAML etc.). Dabei wird auch die Rolle der natürlichen Sprache in der Interaktion beleuchtet.  **Analysieren / Bewerten:** Im Team erarbeiten die Studierenden konkrete Benutzerschnittstellen für unterschiedliche Aufgabenstellungen. Entspre­chend der Aufgaben- und Benutzeranalyse werden Ziele festgelegt, deren Erreichen in der konkreten Umsetzung validiert werden.  **Synthetisieren:** Das Ergebnis des Kurses manifestiert sich in einem kurs­übergreifenden Projekt, das die Entwicklung einer Anwendung mit einem hohen Grad an Nutzerinteraktion beinhaltet. Die Anwendungserstellung wird dabei in kleinere Arbeitspakete disaggregiert. Jedes Team (zwei bis drei Studierende) übernimmt dabei ein Arbeitspaket, organisiert seine indi­viduellen Aufgaben und bringt sich in das Management des Gesamt­pro­jek­tes ein.  Die eigenverantwortliche Organisation der Projektarbeit schließt auch ex­plo­ratives Lernen ein. Die Studierenden werden dabei angeregt (und un­ter­stützt) sich neue Methodiken oder Werkzeuge in Eigeninitiative anzu­eig­nen, sofern dies ihre individuelle Projektarbeit erfordert. |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Inhalt** | **Kennen** | **Verste­hen** | **Anwen­den** | **Analy­sieren** | **Bewer­ten** | **Synthe­tisieren** | | Grundlagen Kognition | X | X | X | X |  |  | | Nutzeranalyse | X | X | X | X | X | X | | Design und Implementierung | X | X | X | X | X | X | | Bewertung |  |  | X | X | X | X | | |
| Modulinhalte | 1. Grundlagen Kognition    * Grundlegende IR Konzepte    * Regular Expressions    * XML 2. Nutzeranalyse    * Definition von Nutzern und Aufgaben    * Mentale Modelle    * Entwicklung von User Stories 3. Design und Implementierung    * GUI controls    * XAML    * Umsetzung in unterschiedlichen Betriebssystemumgebungen 4. Bewertung    * Usability-Prinzipien    * Methoden |
| Lehrformen | Vorlesung unterstützt mit multimedialen Inhalten (2 SWS), die von ACM angeboten werden. Workshops, Kooperation im Team (2 SWS) |
| Voraussetzungen für die Teilnahme | Formelle Voraussetzung bestehen nicht. Für eine erfolgreiche Teilnahme sind Grundlegen und Praxiswissen in der Programmierung erforderlich. |
| Literatur / multimediale Lehr- und Lernprogramme | Carroll, J.M.: ”Human-Computer Interaction in the New Millennium”, ACM Press, New York, 2001.  Cohn, M.: “User Stories Applied”, Addison-Wesley, 2004.  Online-Kurse von ACM zu User Stories und User-Centred Design |
| Verwendbarkeit | Master Applied Computer Science |
| Arbeitsaufwand /  Gesamt-Workload | Gesamt 150 Stunden. Kontaktzeit/Präsenzstudium: 60 Stunden; Selbststudium 40 Stunden; Praxisarbeitszeit: 50 Stunden |
| ECTS, Gewichtung der Note in der Gesamtnote | 5 CP, Gewichtung der Note in der Gesamtnote: 5/120 |
| Leistungsnachweis | Alternative Prüfungsleistung (Projektarbeit) |
| Semester | 2. Semester (Sommersemester) |
| Häufigkeit des Angebots | Einmal pro Studienjahr (Sommersemester) |
| Dauer | Ein Semester (4 SWS) |
| Art der Lehrveranstaltung | Wahlpflichtmodul (Modul aus dem Wahlbereich II) |
| Besonderes |  |