

MODULHANDBUCH

für den Studiengang
Master Informatik
(Prüfungsordnungsversion 2026)

Stand: 22. September 2026

INHALTSVERZEICHNIS

Formale Sprachen und Compilerbau.....	4
Berechenbarkeit und Komplexitätstheorie.....	7
Mathematik und Quantum Computing.....	9
Formale Methoden der Softwaretechnik.....	11
Mathematische Grundlagen der Verschlüsselungstechnik.....	13
Advanced Web Engineering.....	15
Requirements Engineering.....	18
Software-Entwicklungsprozesse.....	21
Design and Modeling of Complex Software Architectures.....	23
Usability Engineering.....	26
Fundamente der Programmierung.....	28
Echtzeitsysteme.....	30
Entwicklung software-intensiver Systeme.....	32
Internet der Dinge.....	34
Autonome mobile Systeme.....	36
Mobile Multi-Agenten-Systeme.....	38
Advanced Regression Methods.....	41
Machine Learning.....	43
Big Data Analytics.....	46
Zeitreihenanalyse.....	48
Anwendungen der KI.....	50
Advanced Regression Methods.....	52
Advanced Web Engineering.....	54
Anwendungen der KI.....	57
Autonome mobile Systeme.....	59
Big Data Analytics.....	61
Design and Modeling of Complex Software Architectures.....	63
Echtzeitsysteme.....	66
Entwicklung software-intensiver Systeme.....	68
Fundamente der Programmierung.....	70
Internet der Dinge.....	72
Machine Learning.....	74
Mobile Multi-Agenten-Systeme.....	77
Requirements Engineering.....	80
Software-Entwicklungsprozesse.....	83
Usability Engineering.....	85
Zeitreihenanalyse.....	87
Projektmanagement.....	89
Personalführung.....	91
Organisatorisch/rechtliche Aspekte von IT-Beschaffungen.....	93
Analoges und Digitales Wahrnehmen.....	95
Ausgewählte Aspekte der Informatik 1.....	97
Ausgewählte Aspekte der Informatik 2.....	99
Ausgewählte Aspekte der Informatik 3.....	101
Ausgewählte Aspekte der Informatik 4.....	103
Ausgewählte Aspekte der Informationssicherheit.....	105

Fortgeschrittene Digitale Forensik.....	107
Grundlagen des Bauens und Entwerfens digitaler Lösungen.....	109
IT-Netze.....	111
Multimodale Interaktion in ambienten Umgebungen.....	113
Anerkannte Wahlpflichtprüfungsleistung.....	116
Anerkannte Wahlpflichtprüfungsleistung.....	117
Anerkannte Wahlpflichtprüfungsleistung.....	118
Guided Research Project.....	119
Thesis mit Kolloquium.....	122

Nummer		Bezeichnung					
46865		Formale Sprachen und Compilerbau					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
					60 h	120 h	
-	Formale Sprachen und Compilerbau		Vorlesung	60			2
-	Formale Sprachen und Compilerbau		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
Nach erfolgreicher Teilnahme an diesem Modul sind die Studierenden in der Lage:							
<u>Wissen und Verstehen:</u>							
- Die Anwendungsgebiete und Systemumgebungen von Compilern zu erläutern. - Die Phasen eines Compilers zu beschreiben und deren Aufgaben und Ergebnisse zu verstehen. - Die Grundlagen der lexikalischen Analyse, insbesondere Grammatiken, reguläre Sprachen und Automaten, zu erklären. - Die Prinzipien der Top-Down-Analyse sowie LL(k)-Grammatiken zu verstehen und anzuwenden. - Die Struktur und Funktion von Syntaxbäumen und abstrakten Syntaxbäumen zu analysieren. - Konzepte der Speicherorganisation, Laufzeitsysteme und Zwischendarstellungen zu erläutern.							
<u>Einsatz, Anwendung und Erzeugung von Wissen:</u>							
- Reguläre Ausdrücke in deterministische endliche Automaten zu übersetzen. - Endliche Automaten geeignet zu vereinfachen und zu minimieren. - Scanner basierend auf deterministischen endlichen Automaten zu implementieren. - Parser unter Verwendung von rekursivem Abstieg für LL(1)-Grammatiken zu entwickeln. - Symboltabellen zu entwerfen und zu implementieren. - Die syntaxgesteuerte Übersetzung mittels attributierter Grammatiken zu realisieren. - Zwischencode (z. B. 3-Adress-Code, stackbasierter Zwischencode) zu erzeugen und maschinenunabhängige sowie maschinenabhängige Optimierungen durchzuführen.							
<u>Kommunikation und Kooperation:</u>							
- In kleinen Teams Compilerkomponenten zu entwerfen und zu implementieren. - Dokumentation und Präsentationen zu Compiler-Techniken für Fachpublikum zu erstellen.							
<u>Wissenschaftliches Selbstverständnis / Professionalität:</u>							
- Effiziente Implementierungen von Compiler-Techniken unter Beachtung theoretischer Grundlagen zu erstellen. - Die praktische Relevanz von Compilern für Softwareentwicklung und Systemprogrammierung kritisch zu reflektieren.							
3	Inhalte						
- Anwendungsgebiete und Systemumgebungen für Compiler - Phasen eines Compilers							

	• Aufgaben und Ergebnisse der Analyse- und Übersetzungsphasen eines Compilers • Grammatiken, Reguläre Sprachen und Automaten im Kontext der Lexikalischen Analyse • Komposition und Minimierung von endlichen Automaten • Systematische Implementierung eines Scanners auf Basis deterministischer, endlicher Automaten • Grundlagen und Prinzip der Top-Down-Analyse inklusive möglicher auftretender Probleme • LL(k)-Grammatiken als Basis für die Top-Down-Analyse • Charakterisierung von LL(1)-Grammatiken • Berechnung von FIRST-, FOLLOW- und Steuermengen für LL(1)-Grammatiken • Implementierung eines Parsers durch rekursiven Abstieg • Syntaxbäume und abstrakte Syntaxbäume • Fehlerbehandlung • Attributierte Grammatiken als Basis für die syntaxgesteuerte Übersetzung • Symboltabellen • Implementierung der syntaxgesteuerte Übersetzung durch eine Variante des rekursiven Abstiegs basierend auf Übersetzungsschemata • Speicherorganisation und Laufzeitsystem • Verschiedene Arten von Zwischendarstellungen., insbesondere 3-Adress-Code und stackbasierter Zwischencode • Übersetzung einer Beispielsprache in Zwischencode • Maschinenunabhängige und maschinenabhängige Optimierungen
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • aktives, selbstgesteuertes Lernen durch Internet-gestützte Aufgaben, Musterlösungen und Begleitmaterialien
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Die Prüfung besteht aus zwei Teilen: 1. schriftliche Klausurarbeit (semesterbegleitend), 60 - 90 Minuten, mit einem Anteil von 30% an der Gesamtnote 2. projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung, 30 Minuten, mit einem Anteil von 70% an der Gesamtnote
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit und bestandene projektbezogene Arbeit (mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung), jeweils mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Robert Rettinger Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • U. Meyer, Grundkurs Compilerbau, Rheinwerk Verlag, zweite Auflage 2024 • C. Wagenknecht, M. Hielscher, Formale Sprachen, abstrakte Automaten und Compiler, Springer 2022 • A.V. Aho, M.S. Lam, R. Sethi und J.D. Ullman, Compilers. Principles, Techniques, and Tools. Addison-Wesley, 2006

- H. Mössenböck, N. Wirth, Compilerbau: Grundlagen und Anwendungen, dpunkt Verlag 2024
- K. D. Cooper und L. Torczon, Engineering a Compiler, Second Edition. Academic Press, 2011

Nummer		Bezeichnung					
46866		Berechenbarkeit und Komplexitätstheorie					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Berechenbarkeit und Komplexitätstheorie		Vorlesung	60			2
-	Berechenbarkeit und Komplexitätstheorie		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>Fach- und Methodenkompetenz:</u> • Grundbegriffe und Konzepte der Berechenbarkeit und Komplexitätstheorie benennen können. • Unterschiedliche Modelle von Turingmaschinen programmieren und analysieren können. • Komplexitätsaussagen von Problemen verstehen, einordnen und bewerten können. • Selbstständig Probleme bezüglich ihrer Berechenbarkeit und Komplexität einschätzen und klassifizieren können. • Bei schweren Problemen die Möglichkeit einer approximativen Lösung prüfen können.						
3	Inhalte • Turingmaschinen: 1-Band-TM; Mehrband-TM; Church-Turing Hypothese; Universelle TM; Nichtdeterministische TM • Berechenbarkeit: Entscheidbare, Semi-entscheidbare und unentscheidbare Probleme; Diagonalisierung; Halteproblem; Reduktion unentscheidbarer Probleme • Komplexitätstheorie: Laufzeiten; Klassen P und NP; P-NP Problem; NP-Vollständigkeit; Polynomielle Reduktion; NP-vollständige Probleme • Approximation: Approximationsgüte; Approximationsalgorithmen; Nicht-Approximierbarkeit						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • Gruppenarbeit • Einzelarbeit • aktives, selbstgesteuertes Lernen durch Internet-gestützte Aufgaben, Musterlösungen und Begleitmaterialien						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen schriftliche Klausurarbeit						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit						
8	Stellenwert der Note für die Endnote						

9	Modulbeauftragte/r Prof. Dr. Robert Preis Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur • Hopcroft, J.E., Motwani, R., Ullman, J.D.; Einführung in Automatentheorie, Formale Sprachen und Berechenbarkeit; Pearson Studium, 3. Auflage, 2011 • Hoffmann, D.W.; Theoretische Informatik; Hanser; 5. Auflage; 2022 • Erk, K., Priese, L.; Theoretische Informatik; Springer; 4. Auflage; 2018

Nummer		Bezeichnung					
47725		Mathematik und Quantum Computing					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
					60 h	120 h	4
-	Mathematik und Quantum Computing		Vorlesung	60			2
-	Mathematik und Quantum Computing		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
Vermittlung der mathematischen Grundlagen des Quantum Computing, soweit sie für das erfolgreiche Studium der Informatik relevant sind. Die Studierenden sollen die unten angegebenen Lehrinhalte kennen und die zentralen Algorithmen beherrschen und in ihrer Bedeutung beurteilen können.							
<u>Wissen und Verstehen:</u>							
- Die Studierenden verstehen die mathematischen Grundlagen der Quantenmechanik, insbesondere Lineare Algebra (Vektoren, Matrizen, Tensorprodukte) und die Bra-Ket-Notation. - Sie können die historische Entwicklung und die Einordnung des Quantum Computings in die Quantenmechanik erklären.							
<u>Einsatz, Anwendung und Erzeugung von Wissen:</u>							
- Die Studierenden analysieren und implementieren grundlegende Quantengatter und deren Zusammensetzung zu Quantenalgorithmen. - Sie wenden Quantenmechanik-Konzepte wie das No-Cloning-Theorem, Quantenteleportation und dichte Codierung praktisch an. - Sie setzen Quanten-Algorithmen (Deutsch, Grover, Shor) zur Lösung spezifischer Problemstellungen ein und bewerten deren Effizienz. - Sie verstehen die Quanten-Fourier-Transformation und implementieren einen Quanten-Addierer auf QFT-Basis. - Sie bewerten Methoden der Quantenkryptografie hinsichtlich ihrer Sicherheit und Anwendbarkeit.							
<u>Kommunikation und Kooperation:</u>							
- Die Studierenden diskutieren zentrale Aspekte des Quantum Computings und dessen mathematische Grundlagen in Arbeitsgruppen. - Sie interpretieren und erklären Forschungsergebnisse im Bereich der Quanteninformatik.							
<u>Wissenschaftliches Selbstverständnis / Professionalität:</u>							
- Sie reflektieren ethische und sicherheitsrelevante Aspekte von Quantenkryptografie und Quantum Computing. - Sie erkennen die Potenziale und Herausforderungen des Quantum Computings für verschiedene Anwendungsbereiche.							
3	Inhalte						
- Mathematische Grundlagen - Quantenmechanischer Überblick - Bits und Qubits - Klassische Gatter und Quantengatter - No-Cloning-Theorem versus Quantenteleportation							

	• Holevo-Schranke versus dichte Codierung • Algorithmus von Deutsch • Algorithmus von Grover • Quanten-Fourier-Transformation • Quanten-Addierer auf QFT-Basis • Algorithmus von Shor • Quantenkryptografie
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • aktives, selbstgesteuertes Lernen durch Internet-gestützte Aufgaben, Musterlösungen und Begleitmaterialien • aktives, selbstgesteuertes Lernen durch Aufgaben, Musterlösungen und Begleitmaterialien • jeweils unmittelbare Rückkopplung und Erfolgskontrolle
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen schriftliche Klausurarbeit
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Robert Rettinger Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • B. Lenze, Mathematik und Quantum Computing, Buch und E-Book, Logos Verlag, Berlin, 2020, zweite Auflage. Ergänzend: • M. Homeister. Quantum Computing verstehen, Springer Vieweg Verlag, Wiesbaden, 2018, fünfte Auflage. • R.J. Lipton, K.W. Regan. Quantum Algorithms via Linear Algebra: A Primer, MIT Press, Cambridge MA, 2014. • M.A. Nielsen, I.L. Chuang. Quantum Computation and Quantum Information, Cambridge University Press, Cambridge, 2010. • C.P. Williams. Explorations in Quantum Computing, Springer-Verlag, London, 2011, zweite Auflage.

Nummer		Bezeichnung						
46859		Formale Methoden der Softwaretechnik						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
-	Formale Methoden der Softwaretechnik			seminaristische Veranstaltung	25	60 h	120 h	4
					35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Formale Methoden sind Sprachen zur Modellierung von Softwaresystemen auf einer gewissen Abstraktions-ebene. Da sie eine formale Semantik besitzen, können die so beschriebenen Modelle auf ihre Korrektheit analysiert werden. Dies ist insbesondere für softwareintensive Systeme wichtig. Die Vorlesung vermittelt Kenntnisse und Fähigkeiten in der Modellierung und Analyse von Softwaresystemen. Die Studenten sollen ferner in die Lage versetzt werden, passende Sprachen und Analysetechniken für die Modellierung auszuwählen. <u>Fach- und Methodenkompetenz:</u> • die Theorie von formalen Methoden einzusetzen • Formale Modelle zu komplexen Systemen zu entwerfen, implementieren und analysieren • Verschiedene formale Methoden und Modelle hinsichtlich Kriterien zu beurteilen <u>Selbstkompetenz:</u> Die/der Studierende/r kann Ideen und Lösungsvorschläge schriftlich und mündlich präsentieren, die eigenständige Präsentation von Lösungen tragen zur Entwicklung von Selbstsicherheit/Sachkompetenz bei; die Entwicklung von Strategien zum Wissens- und Kenntniserwerbs werden durch die Kombination (seminaristische) Vorlesung mit eigenständiger Erarbeitung der Inhalte wissenschaftlicher Literatur unterstützt. <u>Sozialkompetenz:</u> Kooperations- und Teamfähigkeit wird während der Übungs- und Projektphasen trainiert. Die/der Studierende/r kann in Diskussionen zielorientiert argumentieren und mit Kritik sachlich umgehen; er/sie/es kann vorhandene Missverständnisse zwischen Gesprächspartnern erkennen und abbauen. Ergebnisse aus Gruppenarbeit können gemeinsam präsentiert werden.								
3	Inhalte							
• Einbettung von formalen Methoden in den Software-Entwicklungszyklus, Vorgehensmodelle • Methoden zur formalen Programmentwicklung im Großen • Formalismen, die in heutigen Programmentwicklungssystemen Verwendung finden: • Algebraische Spezifikationstechniken • Zustandsorientierte und zeitbehaftete Spezifikationen • Behandlung von Nebenläufigkeit • Verfahren zur Verifikation und Validierung von formalen Entwicklungsschritten, formale Spezifikationssprachen • Werkzeuge zur formalen Programmentwicklung								
4	Lehrformen							
• Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit								

5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Schriftliche Klausurarbeit (Umfang 70%, Dauer 90min) • Prüfungsleistung: Projekt (Umfang 30%)
7	Voraussetzungen für die Vergabe von Kreditpunkten • bestandene Klausurarbeit • bestandenenes Projekt
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Martin Hirsch Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur • Reisig, W. (2013): Understanding Petri Nets – Modeling Techniques, Analysis Methods, Case Studies, Springer • Clarke, E.M., Grumberg, O., Peled, D.A. (1999): Model Checking, MIT Press • Baier, C., Katoen, J.-P. (2008): Principles of Model Checking, MIT Press • Spivey, J.M. (2001): The Z Reference Manual (https://github.com/Spivoxity/zrm/blob/master/zrm-pub.pdf) • Ruhela, V. (2012): Z Formal Specification Language – An Overview, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 01, Issue 06 • http://www.tapaal.net • http://www.uppaal.org

Nummer		Bezeichnung					
46800		Mathematische Grundlagen der Verschlüsselungstechnik					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h			Selbst- studium 120 h		
-	Mathematische Grundlagen der Verschlüsselungstechnik		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme sind die Studierenden in der Lage: - wesentliche aktuelle kryptographische Verfahren zu benennen, - die zugrundeliegenden mathematischen Konzepte zu verstehen, - souverän mit mathematischen Strukturen wie abelschen Gruppen, abstrakten Ringen, endlichen Körpern, Polynomringen, und elliptischen Kurven hantieren zu können, - die abstrakten mathematischen Konzepte in konkreten Implementierungen in einer Programmiersprache umzusetzen, - die Sicherheit verschiedener kryptographischer Verfahren und ihrer Varianten unter gegebenen Parametern zu beurteilen, - kryptographische Verfahren selbst zu implementieren, - kryptographische Verfahren selbst anzugreifen, - potenzielle Schwachstellen einer Implementierung zu erkennen und benennen (z.B. bzgl. Sicherheit, Datenvolumen, Geschwindigkeit), - mit eigenen Varianten zu experimentieren, - die eigenen Varianten kritisch zu analysieren.						
3	Inhalte - Gruppen - Ringe - Ideale - Körper - Kryptographische Anwendungen von modularer Arithmetik (z.B. Diffie-Hellman, RSA,...) - das Diskrete Logarithmusproblem (DLP) - Algorithmisches Finden und Testen von Primzahlen (z.B. Fermat-Test, Miller-Rabin Verfahren) - Algorithmisches Angreifen des DLP (z.B. via Babystep-Giantstep oder Pollard Rho) und generelle algorithmische Techniken, die in der Kryptographie Anwendung finden (z.B. Floyd cycle detection) - Grundlagen zu probabilistischen Algorithmen in der Kryptographie - Mathematische Grundlagen, die zur Analyse probabilistischer Algorithmen herangezogen werden - sog. "sichere Primzahlen" - Vertiefte Analyse, am Beispiel von Diffie-Hellman, wie sich die Parameter (Primzahl, g) auf die Angreifbarkeit des DLP auswirken - Elliptische Kurven über endlichen Körpern - Kryptographische Anwendungen von elliptischen Kurven (z.B. elliptischer Diffie-Hellman)						
4	Lehrformen Abwechselnd Anteile in Form - einer Vorlesung mit Interaktion mit den Studierenden, sowie - eines Praktikums, mit unterstützender Anleitung, in der die Studierenden konkrete Verfahren implementieren oder testen						

	Die Übergänge zwischen diesen Anteilen können fließend sein. Regelmäßig sollen behandelte Themen wiederholt und aufgefrischt werden, oder Erfahrungen aus dem Praktikumsanteil gesammelt und diskutiert werden.
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Es findet ausschließlich eine Abschlussprüfung am Ende der Veranstaltung statt. <i>Prüfungsformate:</i> Schriftliche Klausur, 60 min, oder ersatzweise mündliche Prüfungen Unabhängig vom Prüfungsformat wird mit der Prüfung evaluiert, inwieweit die Studierenden • souverän mit den mathematischen Konzepten aus der Veranstaltung umgehen können, • die abstrakten mathematischen Konzepte mit konkreten Fragen aus der Kryptographie oder Implementierungsfragen verbinden können, • die Angreifbarkeit verschiedener kryptographischer Verfahren und ihrer Varianten unter gegebenen Parametern bewerten können, • bzw. einen konkreten Angriff beschreiben oder durchführen können, • gelernte Techniken und Verfahren wiedergeben oder in konkreten Beispielen anwenden können, • potenzielle Schwachstellen erkennen können
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestehen der Abschlussprüfung
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Oliver Bräunling Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • J. Buchmann, Einführung in die Kryptographie, Springer • H. Delfs, H. Knebl, Introduction to Cryptography, Springer • J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography, Springer • A. Joux, Algorithmic Cryptanalysis, CRC Press • A. Werner, Elliptische Kurven in der Kryptographie, Springer Ergänzend: • B. Lenze, Basiswissen Angewandte Mathematik -- Numerik, Grafik, Kryptik --, Buch und E-Book, Springer Vieweg Verlag, Wiesbaden, 2020, zweite Auflage.

Nummer		Bezeichnung					
46854		Advanced Web Engineering					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Fortgeschrittenes Web-Engineering / Advanced Web-Engineering		Vorlesung	60			4
-	Fortgeschrittenes Web-Engineering / Advanced Web-Engineering		Praktikum	15			2
2 Lernergebnisse (learning outcomes) / Kompetenzen							
<i>Wissen und Verstehen:</i> Nach Abschluss dieses Moduls haben die Studierenden ihre Fähigkeiten vertieft und intensiviert, • verschiedene Web-Architekturen und zentrale Architekturmuster von Web-Anwendungen zu analysieren und zu unterscheiden, • wichtige Web-Standards und -technologien zu benennen und zu kategorisieren.							
<i>Einsatz, Anwendung und Erzeugung von Wissen:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • eine komplexe Web-Engineering-Aufgabe im Rahmen eines mehrwöchigen Projekts umzusetzen, • eine geeignete Web-Architektur zur Lösung eines spezifischen Problems abzuleiten und zu entwerfen, • geeignete Web-Standards und -technologien zur Umsetzung dieser Architektur zu bestimmen und zu kombinieren, • fortgeschrittene Web-Engineering-Tools wie Entwicklungsumgebungen, Bundler, Scaffolding und Transpiler zu verwenden.							
<i>Kommunikation und Kooperation:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • Lösungen kooperativ im Team zu entwickeln und umzusetzen, • ihre Ideen und Lösungen in verschiedenen Formaten wie Gruppenpräsentationen, Code-Reviews, Lightning Talks oder Pitches zu präsentieren, zu erklären und zu diskutieren, insbesondere vor einem Fachpublikum (z.B. Gästen/Partner*innen aus der Industrie oder aus Forschungsprojekten).							
<i>Wissenschaftliches Selbstverständnis/Professionalität:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • industrielle und wissenschaftliche Best Practices aus dem Bereich Web-Engineering situationsadäquat auszuwählen und anzuwenden, • Feedback zu reflektieren und zu bewerten, z. B. aus Code-Reviews mit Experten, und das erhaltene Feedback selbstständig zu nutzen, um ihre Lösungen zu verbessern.							
3 Inhalte							
<i>Modulbeschreibung:</i> In diesem Modul erhalten die Studierenden einen Überblick über die Architekturen komplexer Web-Anwendungen und analysieren deren Unterschiede und Einsatzgebiete. Sie lernen, wie entsprechende Web-Anwendungen durch die Auswahl und den Einsatz geeigneter client- und serverseitiger Technologien realisiert werden können.							
<i>Modulstruktur:</i> Das Modul umfasst die folgenden Themen: 1. Kurze Wiederholung der Grundlagen der Erstellung von Web-Seiten mit HTML, CSS und JavaScript (Bachelor-Stoff) 2. Identifikation, Analyse und Differenzierung von Architekturen moderner Web-Anwendungen:							

	1. Architekturmuster wie MVC und seine Varianten (MVVM, MVP usw.) 2. Anfragebasierte und komponentenbasierte Web-Frameworks 3. Single vs. Multi Page Applications, serverseitiges Rendering, clientseitiges Rendering, hybride Ansätze (z. B. Rehydrierung, Resumability) 4. Reaktive Programmierung/Streaming 3. Vertiefung serverseitiger Technologien zur Entwicklung von Web-Anwendungen (z.B. mit Java, JavaScript) 4. Vertiefung clientseitiger Konzepte und Technologien zur Entwicklung von Web-Anwendungen (z.B. komponentenorientierte Entwicklung, State Management, Routing) 5. Überblick über aktuelle Entwicklungen bei Web-Standards und in der Forschung (z.B. Web Components, WebAssembly)
4	Lehrformen • Flipped/Inverted Classroom: • Online-E-Learning-Materialien mit interaktiven Folien und Videos (asynchrones Selbststudium) • Interaktive Präsenzveranstaltungen für Aufgaben und Übungen anhand von Praxis- und Forschungsbeispielen (z.B. Coding, Gruppenübungen, Lightning Talks), für zusätzliche Vertiefung und zur Beantwortung und Diskussion von Fragen • Projektorientiertes Praktikum: Projektaufgabe, die über das gesamte Semester in Teams bearbeitet wird • Gastvorträge mit Experten und aktuellen Themen aus Forschung und Industrie
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (Umfang: 60%, Dauer: 60-90 Minuten) oder mündliche Prüfung (Umfang 60%, Dauer: 20-25 Minuten); semesterbegleitende Prüfungsleistungen (projektbezogene Arbeit, Umfang: 40%)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan); bestandene semesterbegleitende Prüfungsleistungen
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Simpson, Kyle (2015-2020): You Don't Know JS (Yet), Volume 1-6, O'Reilly/Independently published • Ullenboom, Christian (2024): Spring Boot and Spring Framework 6, Rheinwerk Computing • Jacobson, Daniel; Brail, Greg; Woods, Dan (2011): APIs: A Strategy Guide: Creating Channels with Application Programming Interfaces, O'Reilly • Masse, Mark (2011): REST API Design Rulebook: Designing Consistent Restful Web Service Interfaces, O'Reilly • Porcello, Eve; Banks, Alex (2018): Learning GraphQL: Declarative Data Fetching for Modern Web Apps, O'Reilly • Bass, Len; Clements, Paul; Kazman, Rick (2021): Software Architecture in Practice, SEI Series in Software Engineering, Fourth Edition, Addison-Wesley Professional • Osmani, Addy (2023): Learning JavaScript Design Patterns: A JavaScript and React Developer's Guide, Second Edition, O'Reilly <i>Relevante Standards:</i>

- Ecma International (2025): ECMA-262: ECMAScript® 2025 language specification, 16th Edition, <https://tc39.es/ecma262/>
- WHATWG (2025): HTML Living Standard, <https://html.spec.whatwg.org/>
- WHATWG (2025): DOM Living Standard, <https://dom.spec.whatwg.org>
- WHATWG (2025): Fetch Living Standard, <https://fetch.spec.whatwg.org>
- GraphQL Foundation (2025): GraphQL Specification, <http://spec.graphql.org>

Nummer		Bezeichnung					
46910		Requirements Engineering					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
	-	Requirements Engineering	Vorlesung	60	60 h	120 h	4
-	Requirements Engineering	Übung	20			2	
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage: Wissen und Verstehen: - die Rolle des Requirements Engineering (RE) im Kontext heutiger Herausforderungen in Projekten zu erläutern, z.B. der „VUCA world“ (Volatility, Uncertainty, Complexity, Ambiguity) - verschiedene Kreativitäts- und Erhebungstechniken geeignet zu kombinieren um Anforderungen zu ermitteln, sowie Anforderungen geeignet zu dokumentieren und zu validieren - zu erläutern, wie verschiedene Aktivität im RE mit generativer KI unterstützt werden können und wo Grenzen liegen Einsatz, Anwendung und Erzeugung von Wissen: - Einen RE Prozess in eine Organisation einzupassen und Vorgehensweisen, Techniken und Tools im RE entsprechend den Rahmenbedingungen eines Projekts auszuwählen. - Anforderungen zielorientiert zu erheben, zu analysieren, zu spezifizieren und zu validieren. - Mit spezifischen Themen von Anforderungen wie Variabilität und Adaptivität umzugehen. Kommunikation und Zusammenarbeit: - Sich Techniken und Methoden im RE selbst zu erschließen und an andere zu vermitteln - effektiv mit verschiedenen Stakeholdern, z.B. Kunden, Entwicklern und Endbenutzern, zu kommunizieren, um Anforderungen zu entwickeln, zu verfeinern und zu validieren - in Teams zusammenzuarbeiten, um Konzepte und Lösungen zu entwickeln und dabei unterschiedliche Perspektiven und Interessen abzuwägen Wissenschaftliches Selbstverständnis / Professionalität: - RE Kompetenz in einem domänenübergreifenden Team aufzubauen und zu leiten - die ethische und professionelle Verantwortung zu erkennen, die mit der Umsetzung der Bedürfnisse der Stakeholder in erfolgreiche Systeme verbunden ist - Praktiken im RE kritisch zu bewerten und Verbesserungen zu identifizieren						
3	Inhalte Requirements Engineering (RE) ist die erste Aktivität in der Software-, System- und Serviceentwicklung. Anforderungen sind der Ausgangspunkt für Design, Verifizierung und Validierung sowie für den Test und die Integration von Systemen. Konfigurations- und Änderungsanforderungsmanagement sind mit RE verbunden. Die Definition von Anforderungen und der strukturierte Umgang mit Anforderungen ist nach wie vor ein wichtiger Bereich für die Forschung zu Werkzeugen und Methoden – insbesondere für große und komplexe mechatronische Systeme. In diesem Modul erhalten die Studierenden spezifisches Wissen über den aktuellen Stand der Technik und die wichtigsten zukünftigen Herausforderungen im Bereich RE.						

	• Einführung in das Requirements Engineering • Definition, Relevanz und Herausforderungen • Einbettung abhängig von Systemtyp und Projekteigenschaften • Frameworks (z. B. Jacksons WRSPM-Modell) • Anforderungsermittlung • Stakeholder-Identifizierung • Kreativität und Innovation • Interviews, Fokusgruppen und Ethnographie • Brainstorming und kollaborative Workshops • Anforderungsdokumentation • Standards für Anforderungsspezifikationen (SRS) • Informelle Methoden: Prototypen, Storyboards • Modellierungsanforderungen: i-star, UML, User Stories • Tools: JIRA, Confluence, ReqIF • Validierung und Verifizierung • Qualitätsmerkmale: Vollständigkeit, Konsistenz, Richtigkeit • Prototyping und Benutzerfeedback • Strategien zum Testen von Anforderungen • Anforderungsmanagement • Priorisierungstechniken: z.B. MoSCoW, Kano, Weighted Scoring • Traceability • Auswirkungsanalyse für Änderungen • Versionierung und Änderungsmanagement • Verbesserung des RE-Prozesses • Fortgeschrittene Themen • Software-Produktlinien, adaptive Systeme und Crowd-basierte Systeme • Domänenspezifische Sprachen • Generative KI und natürliche Sprachverarbeitung im RE
4	Lehrformen Vorlesung mit Tafelanschrieb und Projektion
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit, Dauer 90 min (70%) • semesterbegleitende Prüfungsleistung, als projektbezogene Arbeit mit Präsentation (30%)
7	Voraussetzungen für die Vergabe von Kreditpunkten • bestandene Klausurarbeit • erfolgreiches Miniprojekt (projektbezogene Arbeit)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Erik Kamsties Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund

10 Literatur

- Klaus Pohl. Requirements Engineering: Fundamentals, Principles and Techniques. Springer, 2025
- Klaus Pohl und Chris Rupp: Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level, 2015
- Brian Berenbach, Daniel Paulish, Juergen Kazmeier, Arnold Rudorfer. Software and Systems Requirements Engineering In Practice, McGraw-Hill, March 2009
- Klaus Pohl, Günter Böckle und Frank J. van der Linden. Software Product Line Engineering: Foundations, Principles and Techniques, Springer, Januar 2011
- Søren Lausen. Software Requirements - Styles and Techniques, Addison-Wesley, 2002.
- Ellen Gottesdiener. Requirements by Collaboration - Workshops for Defining Needs. Addison-Wesley, 2002

Nummer		Bezeichnung					
46000		Software-Entwicklungsprozesse					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Software-Entwicklungsprozesse		Vorlesung	60			2
-	Software-Entwicklungsprozesse		Praktikum	60			2
2							
Lernergebnisse (learning outcomes) / Kompetenzen							
Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierende in der Lage:							
Wissen und Verstehen:							
• ein hierarchisches Begriffsmodell zu Klassifizierung von Softwareentwicklungsprozesse zu beschreiben. • bekannte Prozessmodelle zu beschreiben. • Ansätze zur Bestimmung der Prozessqualität zu erläutern. • gegebene Prozessbeschreibungen zu klassifizieren. • die wesentlichen Unterschiede bekannte Prozessmodell zu erläutern. • den Einfluss unterschiedlicher Stakeholder in Entwicklungsprozessen zu beschreiben.							
Einsatz, Anwendung und Erzeugung von Wissen:							
• auf der Grundlage von Rahmenbedingungen ein geeignetes Vorgehensmodell auszuwählen. • Alternativen auf Grundlagen einer Risikobetrachtung zu bewerten. • auf Grundlage von Rahmenbedingungen ein Prozessmodell anzupassen bzw. zu konzipieren.							
Kommunikation und Kooperation							
• alternative Ansätze der Prozessgestaltung zu diskutieren. • den Informationsaustausch in Software-Projekten zu gestalten. • die Teamarbeit durch geeignete Prozesse zu unterstützen.							
Wissenschaftliches Selbstverständnis / Professionalität							
• ein Projektumfeld zu analysieren. • bestehende Prozesse zu beurteilen. • Software-Entwicklungsprozesse zu optimieren und zu gestalten.							
3							
Inhalte							
• Vorgehensmodelle • Prozessmodelle • Merkmale bekannter Prozessmodelle • Prozess-Frameworks • Faktor Mensch • Ökonomie von Software-Projekten • Kommunikation in der Softwareentwicklung • Artefakte und Dokumentation • Prozess- und Produktqualität • Integration von Projekt-, Risiko-, Qualitäts-, Problem-, Änderungs- und Anforderungs-Management • Tailoring • Prozessentwurf							

4	Lehrformen • Seminaristische Vorlesung mit Tafelanschrieb und Projektion • Situationsbedingter Wechsel zwischen Vorlesung, praktischen Übungen und der Diskussion von Fallstudien
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Schriftliche Klausurarbeit (60 - 90 Minuten) oder mündliche Prüfung.
7	Voraussetzungen für die Vergabe von Kreditpunkten • Bestandene Klausurarbeit.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Dirk Wiesmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Ludewig K, Lichter H.; "Software Engineering: Grundlagen, Menschen, Prozesse, Techniken", dpunkt.verlag, Heidelberg, 2023 • Balzert H, Ebert Ch.; "Lehrbuch der Software-Technik, Software-Management", Spektrum Akademischer Verlag, Heidelberg, 2008 • Sommerville I.; "Software Engineering", Pearson, Boston, 2016 • Chrissis M.B., Konrad M., Shrum S; "CMMI for Development", Addison-Wesley, Upper Saddle River, NJ, 2011 • Dawson C, Dawson R.; "Software development process models: a technique for evaluation and decision-making", Knowl Process Manag. 21(1), 43-53, 2014

Nummer		Bezeichnung					
46862		Design and Modeling of Complex Software Architectures					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
					60 h	120 h	
-	Entwurf und Modellierung komplexer Software-Architekturen / Design and Modeling of Complex Software Architectures		Vorlesung	60			4
-	Entwurf und Modellierung komplexer Software-Architekturen / Design and Modeling of Complex Software Architectures		Praktikum	15			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
<i>Wissen und Verstehen:</i> Nach Abschluss dieses Moduls werden die Studierenden in der Lage sein,							
• Grundprinzipien des Softwareentwurfs zu unterscheiden, • relevante Werkzeuge und Methoden für den domänenorientierten Entwurf zu unterscheiden und zu kategorisieren, • aktuelle Forschungsansätze zur Modellierung von Softwarearchitekturen zu benennen und einzuordnen.							
<i>Einsatz, Anwendung und Erzeugung von Wissen:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• eine komplexe Problemstellung/Domäne zu analysieren und in Teilproblem/-domänen zu zerlegen, • eine komplexe Software-Entwurfsaufgabe im Rahmen eines mehrwöchigen Projekts zu realisieren, • adäquate Prinzipien des Softwaredesigns auszuwählen und auf konkrete Anwendungsszenarien anzuwenden, • zentrale Muster auf Ebene der Makro- und Mikroarchitektur zu unterscheiden, zu analysieren und anzuwenden, • geeignete Methoden für den domänengetriebenen Entwurf auszuwählen, zu kombinieren und anzuwenden.							
<i>Kommunikation und Kooperation:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• Lösungen kooperativ im Team zu entwickeln und umzusetzen, • geeignete Methoden für die interdisziplinäre Entwicklung von Lösungen auszuwählen und anzuwenden, insbesondere zusammen mit Domänenexpert*innen ohne technischen Hintergrund, • ihre Ideen und Lösungen in verschiedenen Formaten wie Gruppenpräsentationen, Code Reviews, Lightning Talks oder Pitches insbesondere vor einem Fachpublikum (z.B. Gästen/Partner*innen aus der Industrie oder aus Forschungsprojekten) zu präsentieren, zu erläutern und zu diskutieren.							
<i>Wissenschaftliches Selbstverständnis/Professionalität:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• Best Practices aus Industrie und Wissenschaft für den Software-Entwurf auszuwählen und anzuwenden, • Feedback, insbesondere von nicht-technischen Domänenexpert*innen, zu reflektieren und zu bewerten und das erhaltene Feedback selbstständig zur Verbesserung ihrer Lösungskonzepte umzusetzen.							

3	Inhalte <i>Modulbeschreibung:</i> In diesem Modul vertiefen Studierende ihre Kompetenzen im Entwurf von Software-Architekturen komplexer Systeme. Die Studierenden lernen, wie sie durch Auswahl und Anwendung geeigneter Prinzipien, Muster und Methoden eine skalierbare, robuste und wartbare Software-Architektur domänengetrieben entwerfen können. Die Analyse und Diskussion solcher Software-Architekturen erfolgt anhand von Praxisbeispielen und konkreten Lösungen aus Forschungsprojekten. <i>Modulstruktur:</i> Das Modul umfasst die folgenden Themen: 1. Kurze Wiederholung des Bachelor-Stoffs zum Software-Entwurf (z.B. Entwurfsmuster nach Gamma et al., Separation of Concerns, Schichtenarchitektur) 2. Vertiefte Aspekte des Software-Entwurfs: 1. Prinzipien (z.B. Lose Kopplung - hohe Kohäsion, SOLID) 2. Architekturmuster (z.B. Ports and Adapter, CQRS) 3. Methoden (z.B. Domain-Driven Design, WAM-Ansatz) 3. Charakteristika und Muster moderner Architekturstile (z.B. modulare Architekturen, ereignisbasierte Architekturen, Microservice-Architekturen) 4. Modellgetriebener Entwurf, Entwicklung und Rekonstruktion von Software-Architekturen
4	Lehrformen • Flipped/Inverted Classroom: • Online-E-Learning-Materialien mit interaktiven Folien und Videos (asynchrones Selbststudium) • Interaktive Präsenzveranstaltungen für Aufgaben und Übungen anhand von Praxis- und Forschungsbeispielen (z.B. Coding, Gruppenübungen, Lightning Talks), für zusätzliche Vertiefung und zur Beantwortung und Diskussion von Fragen • Projektorientiertes Praktikum: Projektaufgabe, die über das gesamte Semester in Teams bearbeitet wird • Gastvorträge mit Experten und aktuellen Themen aus Forschung und Industrie
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (Umfang: 60%, Dauer: 60-90 Minuten) oder mündliche Prüfung (Umfang 60%, Dauer: 20-25 Minuten); semesterbegleitende Prüfungsleistungen (projektbezogene Arbeit, Umfang: 40%)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan); bestandene semesterbegleitende Prüfungsleistungen
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Vernon, Vernon (2016): Domain-Driven Design Distilled, Addison-Wesley • Evans, Eric (2003): Domain-Driven Design: Tackling Complexity in the Heart of Software, Addison-Wesley • Richardson, Chris (2018): Microservice Patterns: With examples in Java, Manning • Martin, Robert C. (2017): Clean Architecture: A Craftsman's Guide to Software Structure and Design, Pearson

- Lilienthal, Carola (2019): Sustainable Software Architecture: Analyze and Reduce Technical Debt; dpunkt.verlag
- Bass, Len; Clements, Paul; Kazman, Rick (2021): Software Architecture in Practice, SEI Series in Software Engineering, Fourth Edition, Addison-Wesley Professional
- Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1994): Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley
- Combemale, Benoit; France, Robert; Jézéquel, Jean-Marc; Rumpe, Bernhard; Steel, James; Vojtisek, Didier (2016): Engineering Modeling Languages. CRC Press
- Rademacher, Florian (2022). A language ecosystem for modeling microservice architecture, Phd Thesis, <https://dx.doi.org/doi:10.17170/kobra-202209306919>

Nummer		Bezeichnung						
46908		Usability Engineering						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Usability Engineering			Vorlesung	60			2
-	Usability Engineering			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Wissen und Verstehen Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • systematisch Gestaltungslösungen zu entwickeln und auf deren Gebrauchstauglichkeit zu prüfen • Forschungsdesigns zu erstellen, durchzuführen und auszuwerten								
Einsatz, Anwendung und Erzeugung von Wissen Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • methodische Abwägungen vorzunehmen und Auswahlen zu begründen								
Kommunikation und Kooperation Durch die erfolgreiche Teilnahme am Modul werden Studierende • gemeinsam als Kleingruppe ein eigenes Forschungsvorhaben planen und umsetzen • das Vorgehen in einem gemeinsamen Projekt einüben								
Wissenschaftliches Selbstverständnis / Professionalität Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • über die Methodik ihres Fachgebiets und darüber hinaus zu reflektieren und konkrete Arbeiten auf deren Methodik zu untersuchen								
3	Inhalte							
1. Grundbegriffe Usability 2. Bezüge und Anknüpfungspunkte zu Mensch-Computer-Interaktion 3. Usability Engineering – frühe Ansätze 4. Studiendesign und Evaluationsmethoden 5. Projektarbeit 6. Spezialkapitel: Usability bezogen auf einen aktuellen Forschungsgegenstand								
4	Lehrformen							
seminaristische Vorlesung mit Präsentation, Gruppenarbeit und Arbeitsaufträgen								
5	Teilnahmevoraussetzungen							
Keine								

6	Prüfungsformen Hausarbeit mit Referat (80%) Semesterbegleitleistung durch Arbeitsaufträge (20%)
7	Voraussetzungen für die Vergabe von Kreditpunkten insgesamt bestandene Leistung aus • Hausarbeit und Referat (80%) • Semesterbegleitleistung (20%)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Andreas Harrer Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Dix, A., Finlay, J., Abowd, G., Beale, R. (2004): Human-Computer Interaction. Pearson Education Limited. DIN EN ISO 9241 Wilde, T., Hess, T. (2007): Forschungsmethoden der Wirtschaftsinformatik. Wirtsch. Inform. 49, 280–287. Aktuelle Forschungspapiere zum Thema Usability und User Experience Standardisierte Fragebogeninstrumente zum Thema Usability und User Experience

Nummer		Bezeichnung					
46914		Fundamente der Programmierung					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Konzepte in Programmiersprachen / Fundamente der Programmierung		Vorlesung	60			4
-	Konzepte in Programmiersprachen / Fundamente der Programmierung		Praktikum	15			2
2 Lernergebnisse (learning outcomes) / Kompetenzen							
Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierende in der Lage:							
<i>Wissen und Verstehen:</i>							
- wesentliche Eigenschaften funktionaler Programmiersprachen zu benennen. - das Konzept der Funktionsabschlüsse (Closure) zu erläutern. - den syntaktischen Aufbau von Lambda-Ausdrücke darzustellen. - den Normalisierungsprozess von Lambda-Ausdrücken zu erläutern. - grundlegende Programmkonstrukte als Lambda-Ausdrücke zu formulieren. - unterschiedliche Ansätze der Typisierung von Lambda-Ausdrücken zu erläutern. - die Eigenschaften von Dependent Typs zu erläutern.							
<i>Einsatz, Anwendung und Erzeugung von Wissen:</i>							
- die Grundregeln des natürlichen Schließens anzuwenden - Schlussregeln der konstruktiven Logik den Grundkonstrukten typisierter Programme gegenüberzustellen - elementare Beweise als Programme in einem Beweisunterstützungssystem zu schreiben							
<i>Kommunikation und Kooperation:</i>							
- zwischen unterschiedlichen formalen Beschreibungen zu vermitteln - präzise zu formulieren - logisch zu argumentieren							
<i>Wissenschaftliches Selbstverständnis / Professionalität:</i>							
- Anforderungen formal zu spezifizieren - die Zuverlässigkeit von Teilsystemen über Korrektheitsbeweise sicherzustellen - Abstraktionskonzepte zu beurteilen							
3 Inhalte							
<i>Modulbeschreibung:</i>							
Auf Basis des Curry-Howard-Isomorphismus werden fundamentale Konzepte der Programmierung diskutiert.							
<i>Modulstruktur:</i>							
- Grundlagen der praktischen Funktionalen Programmierung - Lambda-Kalkül							

	• Typisierung im Lambda-Kalkül • Natürliches Schließen • Propositions-as-Types • Dependent Types • Programmierung von Beweisen • Formalisierung ausgewählter Grundlagen der Programmierung
4	Lehrformen • Seminaristische Vorlesung, mit Tafelanschrieb und Live-Programmierung • Situationsbedingter Wechsel zwischen Vorlesung, praktischen Übungen und Diskussion von Lösungen
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (60 - 90 Minuten) oder mündliche Prüfung (gemäß akt. Prüfungsplan)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Dirk Wiesmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Bove A., Dybjer P.; "Dependent types at work. In Language Engineering and Rigorous Software Development", LNCS 5520, 57-99, Springer, Berlin, 2008 • Hindley, J.R., Seldin J.P.; "Lambda-Calculus and Combinators, an Introduction", Cambridge University Press, Cambridge, UK, 2008 • Hutton G.; "Programming in Haskell. Cambridge", University Press, Cambridge, UK, 2016 • Wadler P.; "Proposition as types", Commun. ACM, 58(12):75-84, Nov, 2015 • Wadler P., Kokke W., Siek J.G.; "Programming language foundations in Agda", 2021

Nummer		Bezeichnung						
46816		Echtzeitsysteme						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
						60 h	120 h	4
-	Echtzeitsysteme		Vorlesung	60				2
-	Echtzeitsysteme		Praktikum	15				2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:								
Wissen und Verstehen:								
- die technischen Parameter, die für die Auswahl von Planungsverfahren relevant sind, zu ermitteln und anhand derer ein geeignetes Verfahren situationsgerecht auszuwählen, zu implementieren und zu validieren - die besonderen zeitlichen Aspekte von synchronisierten Prozessen und von verteilten Echtzeitsystemen in Entwurf und Implementierung zu adressieren.								
Einsatz, Anwendung und Erzeugung von Wissen:								
- Prozessmengen hinsichtlich Ausführbarkeit zu analysieren und deren Zeitverhalten zu bewerten - Systeme zu konzipieren, die in ihrem zeitlichen Verhalten vorhersagbar sind								
Kommunikation und Kooperation:								
- sich wissenschaftliche Erkenntnisse zu erschließen und - technische Entwürfe von Echtzeitsystemen zu präsentieren und - diese in interdisziplinären Teams umzusetzen.								
Wissenschaftliches Selbstverständnis / Professionalität:								
- die Entwurfsphilosophie hinter Echtzeitsystemen zu artikulieren - ethische und gesellschaftliche Auswirkungen von sicherheitskritischen Echtzeitsystemen zu reflektieren								
3	Inhalte							
- Der Zeitbegriff: mathematische Reduktion der Zeit, Aufbau und Eigenschaften einer Uhr, Zeitstandards - Einführung in die Planung: Prozessparameter, WCET, Brauchbarkeit, Prozesspräzedenz, Prozessanomalien - Prioritätsbasierte Planungsverfahren für aperiodische Prozesse: Earliest Due Date, Earliest Deadline First, Least Laxity First - Prioritätsbasierte Planungsverfahren für periodische Prozesse: Rate und Deadline Monotonic Scheduling, Einplanungstests (LL-Test, kritisches Intervall, RT-Test), Earliest Deadline First - Zeitbasierte Planungsverfahren: äußerer und innerer Zyklus, Anforderungen nach Baker&Shaw, Implementierung des Cyclic Executives - Planungsverfahren für synchronisierte Prozesse: Prioritätsumkehr, Verfahren (Nicht-präemptive kritische Abschnitte, Prioritätsvererbung, Prioritätsobergrenzen), Berechnung von Blockierungszeiten - Echtzeitbetriebssysteme: u.a. Architektur, Scheduler, Umgang mit Interrupts, Fallstudie - Verteilte Systeme: Synchronisation von Uhren, Echtzeiteigenschaften verschiedener Medienzugriffsverfahren, aktuelle Echtzeit-Protokolle								

4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • mündliche Prüfung, 30min (70%) • semesterbegleitende Prüfungsleistung, als projektbezogene Arbeit mit Präsentation (30%)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Erik Kamsties Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Dieter Zöbel. Echtzeitsysteme Grundlagen der Planung, Springer, 2020. • Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications, Springer, 2022 • Jane Liu. Real-Time Systems, Prentice Hall, 2000. • Peter Marwedel. Eingebettete Systeme - Grundlagen Eingebetteter Systeme in Cyber-Physikalischen Systemen, Springer, 2021. • Burns, A., Wellings, A.; Real-Time Systems and Programming Languages; Pearson, 2009. • Heinz Wörn und Dieter Brinkschulte. Echtzeitsysteme, Springer, 2005.

Nummer		Bezeichnung					
46850		Entwicklung software-intensiver Systeme					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Entwicklung software-intensiver Systeme		Vorlesung	60			4
-	Entwicklung software-intensiver Systeme		Übung	20			2
2							
Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:							
Wissen und Verstehen							
- zentrale Konzepte, Methoden und aktuelle Entwicklungen in der Systementwicklung software-intensiver Systeme zu analysieren und in einen fachlichen Kontext einzuordnen - Architektur- und Designprinzipien software-intensiver Systeme kritisch zu reflektieren und deren Auswirkungen auf Skalierbarkeit, Wartbarkeit und Sicherheit zu bewerten							
Einsatz, Anwendung und Erzeugung von Wissen							
- moderne Methoden des Konfigurationsmanagements, der Entwicklung, Integration und Bereitstellung software-intensiver Systeme praktisch anzuwenden und deren Effizienz in realen Projekten zu evaluieren - Test-, Verifikations- und Validierungsverfahren gezielt auszuwählen, anzupassen und auf komplexe Systementwicklungsprojekte zu übertragen - innovative Entwicklungsansätze und Technologien im Bereich DevOps, CI/CD und Dependable Systems zu erforschen, kritisch zu hinterfragen und für spezifische Anwendungsfälle zu adaptieren - wissenschaftlich fundierte Entscheidungen bei der Entwicklung sicherheitskritischer Systeme zu treffen und deren gesellschaftliche sowie ethische Implikationen abzuwägen							
Kommunikation und Kooperation							
- Ergebnisse technischer Analysen und Systementwicklungsprojekte zielgruppengerecht zu dokumentieren, zu präsentieren und kritisch zu diskutieren - in interdisziplinären Teams effektiv zusammenzuarbeiten, Entwicklungsstrategien abzustimmen und Konfliktpotenziale in der Systementwicklung zu reflektieren							
Wissenschaftliches Selbstverständnis / Professionalität							
- ihr eigenes professionelles Handeln in der Systementwicklung unter Berücksichtigung wissenschaftlicher Methoden und ethischer Standards zu reflektieren und weiterzuentwickeln - sich selbstständig neues Wissen zu aktuellen Trends und Technologien der Systementwicklung anzueignen und dieses in praktischen Projekten anzuwenden							

3	Inhalte • Charakterisierung und Abgrenzung Software-intensiver Systeme • Anwendungen, Referenzarchitekturen und Lebenszyklen in verschiedenen Industrien • Systemorientierung (Systembegriff, Sytemkontexte, Systemebenen/Bausteine) • Entwicklungsprozesse für komplexe Software-intensive Systeme (traditionell und agil) • Architektur und Design: Prinzipien und Best Practices für Softwarearchitekturen. • Implementierung und Integration: DevOps-Ansätze, CI/CD-Pipelines, Virtualisierung und Containertechnologien. • Test, Verifikation und Validierung komplexer und eingebetteter Systeme aus Hard- und Software • Deployment: Bereitstellungsstrategien mit modernen Tools • Funktionale Sicherheit: Risiko, -reduktion, SIL und Sicherheit, Normen, Kultur in verschiedenen Industrien • Dependable Systems: Konzepte zu Zuverlässigkeit, Verfügbarkeit, Wartbarkeit und Sicherheit Software-intensiver Systeme. • Architekturmuster und Prozesse für Verfügbarkeit und Funktionale Sicherheit
4	Lehrformen • Das Modul wird in Seminarform abgehalten und kombiniert Vorlesungen, praktische Übungen und Projektarbeit. • Lösung von praxisorientierten Übungsaufgaben, einzeln oder in Teams • Bearbeitung von Programmieraufgaben am Computer in Einzel- oder Gruppenarbeit • vorlesungsbegleitende Projektarbeit mit Abschlusspräsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Mündliche Prüfung (Umfang 100%, Dauer 20-30 min)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Sommerville: Software Engineering ,10. Aufl. , Pearson Deutschland, 2018. • Liggesmeyer, Rombach: Software Engineering eingebetteter Systeme, Spektrum Akademischer Verlag, 2005. • Bass, Clements, Kazman: Software Architecture in Practice, 3rd ed., Addison Wesley, 2013. • Delligatti: SysML Distilled, Addison Wesley, 2014. • Zörner: Software-Architekturen dokumentieren und kommunizieren, 2. Aufl., Hanser, 2015. • Starke, Hruschka: Arc42 in Aktion, Hanser, 2016. • Grünfelder: Software-Test für Embedded Systems, 2. Aufl., dpunkt Verlag, 2017. • Grenning: Test Driven Development of Embedded C, O'Reilly, 2011. • MISRA, MISRA C:2012, Guidelines for the use of the C language in critical systems, HORIBA MIRA Limited, 2015. • Hobbs: Embedded Software Development for Safety-Critical Systems, CRC Press (2016). • verschiedene Normen (werden in der Veranstaltung bekannt gegeben)

Nummer		Bezeichnung						
46860		Internet der Dinge						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS	
					Kontakt- zeit 60 h	Selbst- studium 120 h		
	-	Internet der Dinge		Vorlesung	60			2
	-	Internet der Dinge		Übung	20			1
-	Internet der Dinge	Praktikum	15			1		
2	Lernergebnisse (learning outcomes) / Kompetenzen Fach- und Methodenkompetenz: Nach Abschluss der Lehrveranstaltung sind die Studierenden in der Lage, • Das Konzept des Internets der Dinge (Internet of Things, IoT) einzuordnen und von Machine 2 Machine Communication (m2m) und Industrie 4.0 abzugrenzen • Anwendungsfelder von IoT zu kennen und derer Anforderungen an Technologie und Architektur anzugeben • Technologien, Architekturen und Protokolle des IoT zu verstehen und vorhandene IoT-Systeme zu analysieren • Drahtlose Funktechnologien wie UWB, LoRaWAN, Z-Wave, ZigBee, Bluetooth Smart hinsichtlich Reichweite, Datenrate, Interoperabilität und Stromverbrauch einzuordnen • Routingprotokolle für die Ad-Hoc-Vernetzung wie OLSR, AODV, DSR zur verstehen und in eigene Systeme zu implementieren • Architekturen, Technologien und Protokolle für vorgegebene IoT-Anwendungen auszuwählen und in eigenen Systemen zu implementieren • Neue Architekturen und Routingprotokolle für spezielle IoT-Anwendungen zu entwerfen und zu implementieren							
3	Inhalte • Einleitung • Motivation, Definition, Abgrenzung zu m2m, Industrie 4.0 • Anwendungsgebiete und deren Anforderungen • Übersicht Schichtenmodelle: ISO/OSI, TCP/IP, IPv6 und 6LoWPAN, Bluetooth Smart • Übersicht Funkübertragung: ISM-Bänder, lizenzierte Bänder, UWB • Einordnung Technologien: IEEE 802.15.4, Bluetooth Smart, RFID, LoRaWAN • Architekturen und Protokolle des IoTs • Protokolle der Anwendungsschicht: CoAP, MQTT, GATT • Protokoll-Gateways der Anwendungsschicht: REST-HTTP/CoAP, REST-HTTP/GATT • Topologien: Stern u. Baum-Topologien mit zentralem Gateway, Mesh-Networking, Multi-Gateway • Routing-Protokolle: OLSR, AODV, DSR • IPv6, 6LoWPAN • Grundlagen der digitalen Kommunikation • Abtastung von Signalen, Nyquist Abtasttheorem • Kodierung, Modulation, Kanalkapazität Shannon Fano • Mehrfachzugriffsverfahren: ALOHA, CSMA/CA, FDMA, TDMA, CDMA, OFDM							

	• Grundlagen Funkübertragung: Antennen, Freiraumdämpfung, Fresnelsche Zone, • Beispielhafte Anwendungsgebiete • Smart Home • Szenarien und deren Anforderungen • Technologien: Z-Wave, ZigBee, EnOcean • Beispielhafte Umsetzung anhand eines aktuellen AAL-Forschungsprojektes • Logistik • Szenario Tracking & Tracing • Technologien: RFID, LoRaWAN, UWB • Beispielhafte Umsetzung anhand eines aktuellen Forschungsprojektes
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • vorlesungsbegleitendes Praktikum • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit oder mündliche Prüfung (gemäß akt. Prüfungsplan) • semesterbegleitende Prüfungsleistungen
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Christof Röhrig Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Jan Höller: From machine-to-machine to the internet of things - introduction to a new age of intelligence, Elsevier, 2014 • Peter Waher: Learning Internet of Things - explore and learn about Internet of Things with the help of engaging and enlightening tutorials designed for Raspberry Pi, Packt Publishing, Birmingham, 2015 • Ralf Gessler, Thomas Krause: Wireless-Netzwerke für den Nahbereich, Eingebettete Funksysteme, Vergleich von standardisierten und proprietären Verfahren, Vieweg+Teubner, 2009 • Martin Meyer: Kommunikationstechnik, Konzepte der modernen Nachrichtenübertragung, Vieweg+Teubner, 4. Auflage, 2011. • Andrew S. Tanenbaum, David J. Wetherall: Computernetzwerke, 5. Auflage, Pearson Studium, 2012

Nummer		Bezeichnung					
46863		Autonome mobile Systeme					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Autonome mobile Systeme		Vorlesung	60			2
-	Autonome mobile Systeme		Übung	20			1
-	Autonome mobile Systeme		Praktikum	15			1
2							
Lernergebnisse (learning outcomes) / Kompetenzen							
Fach- und Methodenkompetenz:							
Nach Abschluss der Lehrveranstaltung sind die Studierenden in der Lage,							
• Methoden und Algorithmen der autonomen mobilen Systeme zu verstehen und anzuwenden • Zustandsregler und Zustandsbeobachter zu entwerfen und zu implementieren • Algorithmen zur Zustandschätzung dynamischer Systeme anzuwenden • Algorithmen zur Lokalisierung, Pfadplanung und Kollisionsvermeidung autonomer mobiler Systeme anzuwenden und zu implementieren							
Fachübergreifende Methodenkompetenz:							
• Analyse von dynamischen Systemen • Mathematische Modellbildung dynamischer Systeme • Simulation von dynamischen Systemen mit Matlab/Simulink							
3							
Inhalte							
• Grundlagen dynamischer Systeme • Beschreibung zeitdiskreter Systeme, z-Transformation • Zustandsraumdarstellung • Regelung mittels Zustandsrückführung • Zustandsbeobachtung • Zustandsschätzung dynamischer Systeme • Methode der kleinsten Fehlerquadrate • Kalman Filter, Erweitertes Kalman Filter, Unscented Kalman Filter • Monte Carlo Methoden • Autonome mobile Systeme • Grundlagen der Lokalisierung mobiler Systeme • Lokalisierung mittels Kalman Filter • Lokalisierung mittels Partikel Filter • Umgebungsmodelle und Kartierung • Navigation und Pfadplanung • Kollisionsvermeidung • Ausgewählte Problemstellungen aus aktuellen Forschungsprojekten							
4							
Lehrformen							
• Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • vorlesungsbegleitendes Praktikum • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation							

5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit oder mündliche Prüfung (gemäß akt. Prüfungsplan) • semesterbegleitende Prüfungsleistungen
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Christof Röhrig Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Sebastian Thrun, Wolfram Burgard, Dieter Fox: Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), MIT Press, 2005 • Siegwart, Roland; Nourbakhsh, Illah R.: Introduction to Autonomous Mobile Robots, MIT Press, 2nd Edition, 2011 • Karsten Berns, Ewald von Puttkamer: Autonomous Land Vehicles: Steps towards Service Robots, Vieweg + Teubner Verlag, 2009 • Hertzberg, Joachim; Lingemann, Kai; Nüchter, Andreas: Mobile Roboter - Eine Einführung aus Sicht der Informatik, Springer Vieweg Verlag, 2012 • Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, Sebastian Thrun: Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents), MIT Press, 2005 • Unbehauen, Heinz: Regelungstechnik II, Vieweg Verlag, 9. Auflage, 2007 • Lunze, Jan: Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung, Springer Verlag, 6. Auflage, 2010

Nummer		Bezeichnung					
46852		Mobile Multi-Agenten-Systeme					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
	-	Verteilte und mobile Systeme / Mobile Multi-Agenten-Systeme	Vorlesung	60			4
	-	Verteilte und mobile Systeme / Mobile Multi-Agenten-Systeme	Praktikum	15			2
2							
Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:							
Wissen und Verstehen							
- zentrale Konzepte, Architekturen und Interaktionsmechanismen von mobilen Multiagentensystemen (MAS) zu erklären und sie von klassischen verteilten Systemen abzugrenzen. - Methoden der Koordination, Kooperation und Konsensfindung in mobilen MAS zu analysieren und deren Einsatz in realen Anwendungen zu bewerten.							
Einsatz, Anwendung und Erzeugung von Wissen							
- Planungs-, Lern- und Selbstorganisationsmechanismen in mobilen MAS zu entwerfen, zu implementieren und auf Problemstellungen wie Schwarmintelligenz oder autonome Systeme anzuwenden. - Mobile MAS mit geeigneten Simulationstools zu modellieren, zu testen und ihre Leistungsfähigkeit zu evaluieren.							
Kommunikation und Kooperation							
- wissenschaftliche Erkenntnisse und technische Konzepte zu mobilen MAS klar zu präsentieren und in interdisziplinären Teams anzuwenden. - ethische und gesellschaftliche Auswirkungen intelligenter mobiler MAS kritisch zu reflektieren und verantwortungsbewusst zu gestalten.							
Wissenschaftliches Selbstverständnis / Professionalität							
- sich eigenständig in aktuelle Forschungsthemen zu mobilen MAS einzuarbeiten, neue Fragestellungen zu entwickeln und geeignete Lösungsansätze abzuleiten. - wissenschaftlich fundierte Entscheidungen bei der Entwicklung und Anwendung von mobilen MAS zu treffen und dabei technische sowie ethische Rahmenbedingungen zu berücksichtigen.							
3							
Inhalte							
Grundlagen von Multiagentensystemen (MAS) - Definition und Konzepte von Agenten und Multiagentensystemen - Eigenschaften und Klassifikation von Agenten - Abgrenzung zu klassischen verteilten Systemen Agentenarchitekturen und -modelle - Reaktive, deliberative und hybride Architekturen - Entscheidungsfindung und Autonomie von Agenten - Architekturen für intelligente Agenten Kommunikation und Interaktion in MAS - Agent Communication Languages (ACL) - Sprechakttheorie und Ontologien							

	• Mechanismen der Koordination und Kooperation • Konsensfindung und Verhandlung in MAS • Nash-Gleichgewicht und kooperative Strategien • Verhandlungsprotokolle (Aufgaben- und Wertorientierung) • Anwendungen in Auktionen und elektronischen Märkten • Planung und Lernen in Multiagentensystemen • Zentrale vs. verteilte Planung • Reinforcement Learning für Agentensysteme • Selbstorganisation und evolutionäre Algorithmen • Modellierung von Umgebungen für MAS • Einfluss von deterministischen und dynamischen Umgebungen • Simulation von Agentensystemen mit Mesa (Python) • Herausforderungen in offenen und komplexen Systemen • Kollektive Bewegung und Schwarmintelligenz • Prinzipien der Schwarmintelligenz (bspw. Boid-Modell) • Selbstorganisierte Formationen in Multiagentensystemen • Anwendungen: Drohnenschwärme, autonome Fahrzeuge, Such- und Rettungsmissionen • Anwendungsfälle und Fallstudien • Multiagentensysteme in Robotik und Logistik • Agentensysteme in der Spieleindustrie und Simulation sozialer Systeme • Intelligente Agenten für Smart Cities und IoT • Praktische Übungen und semesterbegleitendes Projekt • Implementierung von Agenten mit Mesa (Python) • Entwicklung eines kooperativen Multiagentensystems • Evaluierung verschiedener Interaktions- und Planungsmechanismen
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • studienbegleitende Prüfungsleistung mit abschließender Präsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • mündliche Prüfung [Umfang: 75%](20-30 min) • studienbegleitende Prüfungsleistung [Umfang: 25%]
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung einschließlich SBL mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund

10 Literatur

- Wooldridge, M.: *An Introduction to MultiAgent Systems*, 2nd ed., John Wiley & Sons, 2009
- Russell, S., & Norvig, P.: *Artificial Intelligence: A Modern Approach*, 4th ed, Pearson, 2021
- Shoham, Y., & Leyton-Brown, K.: *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*, Cambridge University Press, 2009
- Mesa Framework Documentation: <https://mesa.readthedocs.io>

Nummer		Bezeichnung						
46801		Advanced Regression Methods						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Angewandte Statistik / Advanced Regression Methods			Übung	20			4
-	Angewandte Statistik / Advanced Regression Methods			Vorlesung	60			2
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme an der Modulveranstaltung sind die Studierenden in der Lage, mit Regressionsverfahren Informationen aus Daten gewinnen zu können. Sie können eigenständig erkennen, welche Art von statistischem Modell aus dem Bereich der Regressionsanalyse für eine Fragestellung geeignet ist, und die Modellannahmen benennen. Sie können mit Hilfe geeigneter Software ein Modell an Datensätze anpassen und erklären warum welche Methode für die Parameterschätzung und Modellselektion eingesetzt wird. Studierende können einen statistischen Versuchsplan für eine Screening- oder Optimierungsfragestellung aufstellen, Daten selbst erheben und analysieren und eine Dokumentation in Berichtsform erstellen.							
3	Inhalte Die Modulveranstaltung vermittelt Annahmen und Definitionen verschiedener statistischer Modelle aus dem Bereich der Regressionsmodelle. Neben klassischen linearen Modellen werden Varianzanalyse, Generalisierte Lineare Modelle und Gaußprozessmodelle behandelt. Es wird der gesamte Prozess einer statistischen Datenanalyse beginnend mit Modellannahmen, über Hypothesentests, Parameterschätzung mittels Maximum Likelihood und Kleinste-Quadrate-Methode, Residualanalyse, Variablenselektion, bis zur Modellinterpretation und Prognose und Prognoseintervallen abgedeckt. Für die Erhebung eines eigenen Datensatzes werden Grundlagen der statistischen Versuchsplanung (Versuchsplan, Versuchsbereich, Kodierung, Randomisierung, Wiederholungen, Blockbildung) und verschiedene Versuchspläne (Screening- und Optimierungspläne, Raumfüllende Pläne) vermittelt. (odelle,)							
4	Lehrformen Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion, Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit, Vorlesungsbegleitende Projektarbeiten an praxisnahen Anwendungen mit Dokumentationen und Präsentationen							
5	Teilnahmevoraussetzungen Keine							

6	Prüfungsformen Projektarbeit mit mündlicher Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten erfolgreiche Projektarbeit
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sonja Kuhnt Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G. (2016), Statistik - der Weg zur Datenanalyse, 8. Aufl., Springer, Berlin. • Fahrmeir, L., Kneib, Th., Lang, S., Marx, B. (2013), Regression: Models, Methods and Applications, Springer, Berlin. • Dobson, A.J., Barnett, A.G. (2018), An Introduction to Generalized Linear Models, 4th edition, Taylor & Francis Ltd, Boca Raton. • Sievertz, K., van Bebber, D., Hochkirchen, Th. (2017) Statistische Versuchsplanung - Design of Experiments (DoE), 4te Auflage, Springer Vieweg, Berlin.

Nummer		Bezeichnung					
46839		Machine Learning					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Maschinelles Lernen / Machine Learning		Vorlesung	60			2
-	Maschinelles Lernen / Machine Learning		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
Die Lehrveranstaltung beschäftigt sich mit der Entwicklung und Analyse von maschinellen Lernverfahren in Anwendungen der Informatik, der Medizinischen Informatik bzw. für allgemeine Informationssysteme. Nach erfolgreicher Teilnahme an der Lehrveranstaltung haben die Studierenden die folgenden Kompetenzen erworben: Kenntnisse (Wissen): - kennen die wichtigsten Fachbegriffe des maschinellen Lernens und können damit lernende Systeme erklären. - Sie können den Einsatz maschineller Lernverfahren für eigene Anwendungsaufgaben implementieren und evaluieren. Dazu kennen die Studierenden typische Anwendungen dieser Methoden. - kennen Projektmanagementmethoden für Anwendungen des Maschinellen Lernens wie CRISP-DM - kennen Erklärungskomponenten für maschinelles Lernen und können die Ergebnisse interpretieren - kennen typische Probleme des maschinellen Lernens wie Overfitting und Information Leakage und können diese vermeiden - kennen die theoretischen Grenzen maschineller Lernsysteme und können diese formal beschreiben und zur Abschätzung der Grenzen eigener Anwendungen nutzen. Fertigkeiten - können maschinelle Lernsysteme für konkrete Anwendungskontexte der Informatik entwerfen, implementieren und analysieren. - können ethische Grundlagen und Grenzen maschineller Lernsysteme hinterfragen und diskutieren - können geeignete Methoden des maschinellen Lernens für Anwendungen auswählen - können einfache Deep Learning Lösungen mit JupyterHub implementieren - können Miniprojekte mit Industriebezug selbstständig im Team planen, durchführen, analysieren und dokumentieren. Kompetenzen (Selbst- und Sozialkompetenz) - können Ideen und Lösungsvorschläge mündlich und schriftlich formulieren - können Aufgaben in Übungen und Praktika selbständig lösen und die Ergebnisse präsentieren - können sich theoretische Inhalte zum Thema Maschinelles Lernen aus wissenschaftlicher Literatur selbstständig erarbeiten und präsentieren - können in den Übungs- und Projektphasen kooperativ Lösungen erarbeiten - können in den Projektphasen kooperativ Aufgaben zur Lösung planen, verteilen und gemeinsam durchführen - können in Diskussionen zielorientiert argumentieren und mit Kritik sachlich umgehen - können Ergebnisse von Gruppenarbeiten gemeinsam präsentieren - können Projektergebnisse bewerten und Verbesserungsvorschläge formulieren - können Missverständnisse zwischen Gesprächspartnern erkennen und abbauen							

3	Inhalte - Grundbegriffe des Maschinellen Lernens - Nutzung von KNime für das Maschinelle Lernen - Design von Evaluationsstudien für Maschinelle Lernverfahren und Durchführung solcher Studien - Lineare Modelle - Verschiedene Modelle überwachter und unüberwachter Neuronaler Netzwerke - Lernverfahren für strukturierte Daten: Support Vektor Maschine, Entscheidungsbäume, Random Forest, Gradient Boosting Machines (GBM) - Nächster Nachbarverfahren, Lazy Learning und Embeddings - Bayesische Netzwerke - Unüberwachte Lernverfahren (k-means) - Kombinationsmodelle (Ensembles, Boosting Machines) - Deep Learning Modelle (Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), Transformer Architekturen z.B. BERT, Visual Transformer) - Deep Learning Konzepte - Transferlernen, Datenvermehrung, Generative Adversarial Networks (GAN) - Deep Learning - Parallelisierung mit GPUs, Implementierung auf mobilen Plattformen mit geringen Ressourcen - Large Language Models und Nutzung von Embeddings, Retrieval Augmented Generation (RAG) - Theoretische Konzepte: Bias-Varianz Dilemma, No Free Lunch Theorem - Erklärbarkeit von Modellen - Anwendungen mit Daten verschiedener Modalitäten (Text, Image, Sound), Word2Vec, FastText, Transformer - Methoden zur Verbesserung der Generalisierungsleistung (Regularisierung, Feature Selektion, Dimensionsreduktion, Komplexitätsanpassung) - Problemlösung am Beispiel studiengangsbezogener Miniprojekte aus industriellen Anwendungen (studentische Miniprojekte in 2-3er Teams)
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation • Umgedrehter Unterricht (inverted classroom)
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit (70% der Prüfungsleistung) • semesterbegleitende Prüfungsleistungen (30% der Prüfungsleistung)
7	Voraussetzungen für die Vergabe von Kreditpunkten • bestandene Klausurarbeit • erfolgreiche semesterbegleitende Prüfungsleistung (Miniprojekt mit Präsentation)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr.-Ing. Christoph M. Friedrich Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur - I. Witten, E. Frank, M. Hall und C. J. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4. Auflage, Morgan Kaufmann (2017) - elektronische Version im Intranet verfügbar - C. M. Bishop, Pattern Recognition and Machine Learning, Springer (2006)

- E. Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning), Revised and Updated Edition, MIT Press (2021)
- I. Goodfellow, Y. Bengio und A. Courville: Deep Learning, MIT Press (2016)

Nummer		Bezeichnung						
46001		Big Data Analytics						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
						60 h	120 h	
-	Big Data Analytics			Vorlesung	60			2
-	Big Data Analytics			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreicher Teilnahme an den Modulveranstaltungen und dem Abschließen der Projektarbeit sind die Studierenden in der Lage ...								
• Anforderungen an Big Data Anwendungen zu beurteilen und geeignete Verfahren zur Datenanalyse auszuwählen • Big Data Anwendungen von klassischen Anwendungen abzugrenzen und zu erläutern • anhand von Anwendungsszenarien zu entscheiden, welche Verfahren der Datenanalyse großer Datenmengen (sowohl statischer als auch Echtzeitdaten) in Big Data Anwendungen eingesetzt werden • exemplarisch Big Data Anwendungen zu konzipieren, zu entwerfen, zu implementieren und kritisch zu reflektieren. • die Grundlagen der Datenanalyse von großen Datenmengen und Echtzeitdaten zu erklären								
3	Inhalte							
• Grundlagen zum Thema Big Data • Herausforderungen und Architektur von Big Data Anwendungen • Grundlagen von Echtzeitanwendungen im Big Data Kontext und deren Besonderheiten • Grenzen von klassischen Lösungskonzepten im Big Data Kontext (CAP-Theorem, vertikales / horizontales Skalieren), NoSQL Datenbanken und Datenvisualisierung • Ethische, soziale und wirtschaftliche Bewertung von Big Data-Anwendungen • Grundlagen und Verfahren zur Datenanalyse von großen Datenmengen • Verfahren zum Verteilen von Daten (MapReduce, RDD) • Management Rechenclustern und verteilte Datenhaltung • Methoden und Technologien der Datenanalyse großer Datenmengen • Grundlagen und Verfahren zur Datenanalyse in Echtzeitsystemen • Konzeption und Entwurf von Datenstrom Anwendungen (Datenstrom-Managementsysteme, Datenstrom-Designmuster) • Methoden und Technologien der Datenstromanalyse (Streaming Analytics) • Anomalie-Erkennung und Überwachung von Datenströmen • Unterschiede und Gemeinsamkeiten bei der Datenanalyse vor großen Datenmengen bzw. Daten in Echtzeitsystemen, insbesondere der betrachteten Verfahren								
4	Lehrformen							
• Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitendes Praktikum • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • vorlesungsbergleitendes Projekt								
5	Teilnahmevoraussetzungen							
Keine								
6	Prüfungsformen							
Die Modulprüfung besteht aus zwei Teilen:								

	1. zu 60% aus einer mündlichen Prüfung (20-30 Minuten, gemäß akt. Prüfungsplan), in der die Studierenden Anwendungsszenarien analysieren, verschiedene theoretische Grundlagen erläutern und situativ anwenden können und 2. zu 40% aus einer semesterbegleitenden Prüfungsleistung (Projektarbeit mit abschließendem Referat), mit der die Studierenden zeigen sollen, dass sie die in der Vorlesung gelernten Technologien anwenden, reflektieren und bewerten können.
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung und bestandene projektbezogene Arbeit (mit Dokumentation und Präsentation), jeweils mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Klaus Kaiser Prof. Dr. Inga Saatz Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Kleppmann, Designing data-intensive applications, O'Reilly Media (2017) • Bifet, Machine learning for data stream, MIT-Press (2017) • Ellis, Real-time analytics, Wiley & Sons (2014) • Parsian, Data Algorithmns, O'Reilly Media (2015) • Arundel und Domingus, Cloud Native DevOps mit Kubernetes, dpunkt.verlag (2019) Weitere aktuelle Literatur wird in der Vorlesung bekannt gegeben.

Nummer		Bezeichnung						
46002		Zeitreihenanalyse						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Zeitreihenanalyse			Vorlesung	60			2
-	Zeitreihenanalyse			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme an den Modulveranstaltungen und dem Abschließen der Projektarbeit sind die Studierenden in der Lage: • Methoden zur Modellierung, Analyse und Vorhersage von Zeitreihen umfassend zu verstehen und anzuwenden • Klassische und moderne Ansätze der Zeitreihenanalyse auf unterschiedliche Anwendungsbereiche auszuwählen und zu implementieren. • Zeitreihendaten mit fortgeschrittenen Programmierertools zu analysieren, zu modellieren und zu visualisieren • Unsicherheiten und Vorhersagefehler in Modellen zu bewerten und adäquate Maßnahmen zur Verbesserung vorzuschlagen • Projekte zur Zeitreihenanalyse selbstständig zu planen, durchzuführen, kritisch zu reflektieren, zu dokumentieren und zu präsentieren							
3	Inhalte Einführung in Zeitreihen • Stationarität, Trendanalysen • Autokorrelation und Saisonalität • Glättungs- und Filterverfahren • Umgang mit fehlenden und extremen Werten Klassische Methoden der Zeitreihenanalyse • Lineare Modelle: AR, MA, ARMA, ARIMA • Erweiterte Modelle: SARIMA, GARCH • Modellanpassung: Modellidentifikation, Schätzung der Modellparameter, Validierungsmethoden und Leistungsmetriken • Modellvorhersage und Modelldiagnose • State-Space-Modelle (Zustandsraummodelle) und Kalman-Filter Weitere Methoden der Zeitreihenanalyse • Maschinelles Lernen für Zeitreihen (z. B. LSTMs, Prophet, XGBoost) • Regressionsmodelle für Zeitreihen Softwarepakete in R bzw. Python Praktische Anwendungen • Zeitreihenanalyse in Wirtschaft, Klima- und Wettervorhersagen, Finanz- und Gesundheitsdaten • Praxisprojekt: Analyse realer Zeitreihendaten							
4	Lehrformen • seminaristischer Unterricht mit Flipchart, Smartboard oder Projektion • Projektarbeit begleitendes Praktikum • Bearbeitung von Projektarbeit in Teamarbeit • Projektarbeit mit abschließender Präsentation							

5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Seminaristische Ausarbeitung und Vorstellung eines Themas • Projektarbeit mit abschließender Präsentation
7	Voraussetzungen für die Vergabe von Kreditpunkten • erfolgreiche Projektarbeit • erfolgreiche mündliche Präsentation • erfolgreich durchgeführte Ausarbeitung und Präsentation eines Themas
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Nadja Bauer Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Schlittgen, R., Sattarhoff, C (2020): Angewandte Zeitreihenanalyse mit R, 4., erweiterte und aktualisierte Auflage. Walter de Gruyter GmbH, Berlin/Boston. • Vogel, J. (2015): Prognose von Zeitreihen. Eine Einführung für Wirtschaftswissenschaftler. Springer Gabler, Wiesbaden • Box, G.E.P., Jenkins, G.M. & Reinsel (1994). Time Series Analysis: Forecasting and Control. Revised Edition. Holden Day, San Francisco. • Brockwell, J.P. & Davis, R.A. (1996). Introduction to Time Series and Forecasting. Springer, New York

Nummer		Bezeichnung						
46918		Anwendungen der KI						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
						60 h	90 h	
-	Anwendungen der KI			Vorlesung	60			4
-	Anwendungen der KI			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
<u>Fach- und Methodenkompetenz:</u>								
- Absolvent*innen des Moduls beherrschen grundlegende und weiterführende Konzepte der Künstlichen Intelligenz und sind in der Lage, aktuelle Entwicklungen und Methoden der Künstlichen Intelligenz auf konkrete praktische Fragestellungen der Wirtschaftsinformatik anzuwenden - Die Teilnehmer*innen sind in der Lage, den Nutzen und die Grenzen der betrachteten Inhalte und Methoden in Bezug auf konkrete praktische Anwendungen der Wirtschaftsinformatik sicher einzuschätzen. - Die Teilnehmer*innen sind sicher im Umgang mit aktuellen Programmbibliotheken und können diese projektorientiert auf konkrete Fragestellungen anwenden.								
<u>Selbstkompetenz:</u>								
Die Teilnehmer*innen sind in der Lage, sich eigenständig mit aktuellen Entwicklungen im Bereich der Künstlichen Intelligenz und seinen Spezialisierungen und aktuellen Anwendungen im Bereich der Wirtschaftsinformatik auseinanderzusetzen und die Kernaussagen nachzuvollziehen.								
<u>Sozialkompetenz:</u>								
- Die Teilnehmer*innen sind in der Lage, Diskussionen zu wissenschaftlichen Fragestellungen (insbesondere hinsichtlich der Anwendbarkeit der vermittelten Inhalte für ihr Studiengebiet) zu führen. - Die Teilnehmer*innen erfassen die Relevanz der vermittelten Inhalte für ihr Studiengebiet und sind fähig, diese Relevanz adäquat zu kommunizieren. - Die Teilnehmer*innen können in projektorientierter Gruppenarbeit gemeinsam Herausforderungen der Projektaufgaben diskutieren, mögliche alternative Vorgehensweisen identifizieren und begründete Vorgehensweisen festlegen, umsetzen und evaluieren.								
3	Inhalte							
Die Veranstaltung beinhaltet folgende Themenbereiche:								
Grundlagenteil:								
- Einführung in die Programmiersprache Python und ausgewählte Bibliotheken zur Vorbereitung und Manipulation von Daten - Grundbegriffe der Künstlichen Intelligenz (Agententheorie, Umgebungen, Anwendungsbereiche, Suchverfahren, Markov Decision Processes, Constraint Satisfaction Problems, etc.) - Einführung in maschinelles Lernen (neuronale Netze, Lernparadigmen, Anwendungen etc.)								
Hauptteil:								
- Klassische Optimierungsverfahren der KI und zugehörige Lösungsansätze am Beispiel des Traveling Salesman Problems - Einführung in das Gebiet des Natural Language Processings mit Fokus auf autonomen textbasierten Dialogsystemen (chatbots) und Sentiment Analysis								

	• Reinforcement Learning: aktuelle Verfahren und Grenzen sowie exemplarische Anwendung in den Bereichen Optimierungsverfahren und autonome textbasierte Dialogsysteme • Betrachtung ausgewählter Themen aus dem Bereich KI Ethik mit Fokus auf Anwendungen und Relevanz in der Wirtschaftsinformatik Projektorientierter Teil: • Projektorientierte praktische Anwendung der erlernten Inhalte auf konkrete, im Bereich der Wirtschaftsinformatik relevante Themenbereiche. Mögliche in der Lehrveranstaltung auszuwählende Beispiele: Entwicklung autonom agierender Chatbots im Kundensupport, Lösung konkreter Optimierungsprobleme beispielsweise in der Lagerhaltung, etc. Hierbei sollen in der praktischen Anwendung die in der Lehrveranstaltung betrachteten Verfahren angewendet und bewertet werden.
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation
5	Teilnahmevoraussetzungen Siehe jeweils gültige Prüfungsordnung (BPO/MPO) des Studiengangs.
6	Prüfungsformen Die Prüfung besteht aus zwei Teilen: • Mündliche Prüfung (30 Minuten) mit einem Anteil von 50% an der Gesamtnote • Projektarbeit mit mündlicher Prüfung mit einem Anteil von 50% an der Gesamtnote
7	Voraussetzungen für die Vergabe von Kreditpunkten • Bestandene mündliche Prüfung • Bestandene Projektarbeit Jeweils mit der Note 4.0 oder besser.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sebastian Bab Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Auswahl an Literaturhinweisen: • Stuart Russell und Peter Norvig, Artificial Intelligence: A Modern Approach, Global Edition, Pearson 2021

Nummer		Bezeichnung						
46801		Advanced Regression Methods						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
						60 h	120 h	
-	Angewandte Statistik / Advanced Regression Methods			Übung	20			4
-	Angewandte Statistik / Advanced Regression Methods			Vorlesung	60			2
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme an der Modulveranstaltung sind die Studierenden in der Lage, mit Regressionsverfahren Informationen aus Daten gewinnen zu können. Sie können eigenständig erkennen, welche Art von statistischem Modell aus dem Bereich der Regressionsanalyse für eine Fragestellung geeignet ist, und die Modellannahmen benennen. Sie können mit Hilfe geeigneter Software ein Modell an Datensätze anpassen und erklären warum welche Methode für die Parameterschätzung und Modellselektion eingesetzt wird. Studierende können einen statistischen Versuchsplan für eine Screening- oder Optimierungsfragestellung aufstellen, Daten selbst erheben und analysieren und eine Dokumentation in Berichtsform erstellen.							
3	Inhalte Die Modulveranstaltung vermittelt Annahmen und Definitionen verschiedener statistischer Modelle aus dem Bereich der Regressionsmodelle. Neben klassischen linearen Modellen werden Varianzanalyse, Generalisierte Lineare Modelle und Gaußprozessmodelle behandelt. Es wird der gesamte Prozess einer statistischen Datenanalyse beginnend mit Modellannahmen, über Hypothesentests, Parameterschätzung mittels Maximum Likelihood und Kleinste-Quadrate-Methode, Residualanalyse, Variablenselektion, bis zur Modellinterpretation und Prognose und Prognoseintervallen abgedeckt. Für die Erhebung eines eigenen Datensatzes werden Grundlagen der statistischen Versuchsplanung (Versuchsplan, Versuchsbereich, Kodierung, Randomisierung, Wiederholungen, Blockbildung) und verschiedene Versuchspläne (Screening- und Optimierungspläne, Raumfüllende Pläne) vermittelt. (odelle,)							
4	Lehrformen Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion, Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit, Vorlesungsbegleitende Projektarbeiten an praxisnahen Anwendungen mit Dokumentationen und Präsentationen							
5	Teilnahmevoraussetzungen Keine							

6	Prüfungsformen Projektarbeit mit mündlicher Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten erfolgreiche Projektarbeit
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sonja Kuhnt Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G. (2016), Statistik - der Weg zur Datenanalyse, 8. Aufl., Springer, Berlin. • Fahrmeir, L., Kneib, Th., Lang, S., Marx, B. (2013), Regression: Models, Methods and Applications, Springer, Berlin. • Dobson, A.J., Barnett, A.G. (2018), An Introduction to Generalized Linear Models, 4th edition, Taylor & Francis Ltd, Boca Raton. • Sievertz, K., van Bebber, D., Hochkirchen, Th. (2017) Statistische Versuchsplanung - Design of Experiments (DoE), 4te Auflage, Springer Vieweg, Berlin.

Nummer		Bezeichnung						
46854		Advanced Web Engineering						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Fortgeschrittenes Web-Engineering / Advanced Web-Engineering			Vorlesung	60			4
-	Fortgeschrittenes Web-Engineering / Advanced Web-Engineering			Praktikum	15			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
<i>Wissen und Verstehen:</i> Nach Abschluss dieses Moduls haben die Studierenden ihre Fähigkeiten vertieft und intensiviert, • verschiedene Web-Architekturen und zentrale Architekturmuster von Web-Anwendungen zu analysieren und zu unterscheiden, • wichtige Web-Standards und -technologien zu benennen und zu kategorisieren. <i>Einsatz, Anwendung und Erzeugung von Wissen:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • eine komplexe Web-Engineering-Aufgabe im Rahmen eines mehrwöchigen Projekts umzusetzen, • eine geeignete Web-Architektur zur Lösung eines spezifischen Problems abzuleiten und zu entwerfen, • geeignete Web-Standards und -technologien zur Umsetzung dieser Architektur zu bestimmen und zu kombinieren, • fortgeschrittene Web-Engineering-Tools wie Entwicklungsumgebungen, Bundler, Scaffolding und Transpiler zu verwenden. <i>Kommunikation und Kooperation:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • Lösungen kooperativ im Team zu entwickeln und umzusetzen, • ihre Ideen und Lösungen in verschiedenen Formaten wie Gruppenpräsentationen, Code-Reviews, Lightning Talks oder Pitches zu präsentieren, zu erklären und zu diskutieren, insbesondere vor einem Fachpublikum (z.B. Gästen/Partner*innen aus der Industrie oder aus Forschungsprojekten). <i>Wissenschaftliches Selbstverständnis/Professionalität:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage, • industrielle und wissenschaftliche Best Practices aus dem Bereich Web-Engineering situationsadäquat auszuwählen und anzuwenden, • Feedback zu reflektieren und zu bewerten, z. B. aus Code-Reviews mit Experten, und das erhaltene Feedback selbstständig zu nutzen, um ihre Lösungen zu verbessern.								
3	Inhalte							
<i>Modulbeschreibung:</i> In diesem Modul erhalten die Studierenden einen Überblick über die Architekturen komplexer Web-Anwendungen und analysieren deren Unterschiede und Einsatzgebiete. Sie lernen, wie entsprechende Web-Anwendungen durch die Auswahl und den Einsatz geeigneter client- und serverseitiger Technologien realisiert werden können. <i>Modulstruktur:</i> Das Modul umfasst die folgenden Themen: 1. Kurze Wiederholung der Grundlagen der Erstellung von Web-Seiten mit HTML, CSS und JavaScript (Bachelor-Stoff) 2. Identifikation, Analyse und Differenzierung von Architekturen moderner Web-Anwendungen:								

	1. Architekturmuster wie MVC und seine Varianten (MVVM, MVP usw.) 2. Anfragebasierte und komponentenbasierte Web-Frameworks 3. Single vs. Multi Page Applications, serverseitiges Rendering, clientseitiges Rendering, hybride Ansätze (z. B. Rehydrierung, Resumability) 4. Reaktive Programmierung/Streaming 3. Vertiefung serverseitiger Technologien zur Entwicklung von Web-Anwendungen (z.B. mit Java, JavaScript) 4. Vertiefung clientseitiger Konzepte und Technologien zur Entwicklung von Web-Anwendungen (z.B. komponentenorientierte Entwicklung, State Management, Routing) 5. Überblick über aktuelle Entwicklungen bei Web-Standards und in der Forschung (z.B. Web Components, WebAssembly)
4	Lehrformen • Flipped/Inverted Classroom: • Online-E-Learning-Materialien mit interaktiven Folien und Videos (asynchrones Selbststudium) • Interaktive Präsenzveranstaltungen für Aufgaben und Übungen anhand von Praxis- und Forschungsbeispielen (z.B. Coding, Gruppenübungen, Lightning Talks), für zusätzliche Vertiefung und zur Beantwortung und Diskussion von Fragen • Projektorientiertes Praktikum: Projektaufgabe, die über das gesamte Semester in Teams bearbeitet wird • Gastvorträge mit Experten und aktuellen Themen aus Forschung und Industrie
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (Umfang: 60%, Dauer: 60-90 Minuten) oder mündliche Prüfung (Umfang 60%, Dauer: 20-25 Minuten); semesterbegleitende Prüfungsleistungen (projektbezogene Arbeit, Umfang: 40%)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan); bestandene semesterbegleitende Prüfungsleistungen
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Simpson, Kyle (2015-2020): You Don't Know JS (Yet), Volume 1-6, O'Reilly/Independently published • Ullenboom, Christian (2024): Spring Boot and Spring Framework 6, Rheinwerk Computing • Jacobson, Daniel; Brail, Greg; Woods, Dan (2011): APIs: A Strategy Guide: Creating Channels with Application Programming Interfaces, O'Reilly • Masse, Mark (2011): REST API Design Rulebook: Designing Consistent Restful Web Service Interfaces, O'Reilly • Porcello, Eve; Banks, Alex (2018): Learning GraphQL: Declarative Data Fetching for Modern Web Apps, O'Reilly • Bass, Len; Clements, Paul; Kazman, Rick (2021): Software Architecture in Practice, SEI Series in Software Engineering, Fourth Edition, Addison-Wesley Professional • Osmani, Addy (2023): Learning JavaScript Design Patterns: A JavaScript and React Developer's Guide, Second Edition, O'Reilly <i>Relevante Standards:</i>

- Ecma International (2025): ECMA-262: ECMAScript® 2025 language specification, 16th Edition, <https://tc39.es/ecma262/>
- WHATWG (2025): HTML Living Standard, <https://html.spec.whatwg.org/>
- WHATWG (2025): DOM Living Standard, <https://dom.spec.whatwg.org>
- WHATWG (2025): Fetch Living Standard, <https://fetch.spec.whatwg.org>
- GraphQL Foundation (2025): GraphQL Specification, <http://spec.graphql.org>

Nummer		Bezeichnung						
46918		Anwendungen der KI						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h	Selbst- studium 90 h					
-	Anwendungen der KI			Vorlesung	60			4
-	Anwendungen der KI			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Fach- und Methodenkompetenz:								
- Absolvent*innen des Moduls beherrschen grundlegende und weiterführende Konzepte der Künstlichen Intelligenz und sind in der Lage, aktuelle Entwicklungen und Methoden der Künstlichen Intelligenz auf konkrete praktische Fragestellungen der Wirtschaftsinformatik anzuwenden - Die Teilnehmer*innen sind in der Lage, den Nutzen und die Grenzen der betrachteten Inhalte und Methoden in Bezug auf konkrete praktische Anwendungen der Wirtschaftsinformatik sicher einzuschätzen. - Die Teilnehmer*innen sind sicher im Umgang mit aktuellen Programmbibliotheken und können diese projektorientiert auf konkrete Fragestellungen anwenden.								
Selbstkompetenz:								
Die Teilnehmer*innen sind in der Lage, sich eigenständig mit aktuellen Entwicklungen im Bereich der Künstlichen Intelligenz und seinen Spezialisierungen und aktuellen Anwendungen im Bereich der Wirtschaftsinformatik auseinanderzusetzen und die Kernaussagen nachzuvollziehen.								
Sozialkompetenz:								
- Die Teilnehmer*innen sind in der Lage, Diskussionen zu wissenschaftlichen Fragestellungen (insbesondere hinsichtlich der Anwendbarkeit der vermittelten Inhalte für ihr Studiengebiet) zu führen. - Die Teilnehmer*innen erfassen die Relevanz der vermittelten Inhalte für ihr Studiengebiet und sind fähig, diese Relevanz adäquat zu kommunizieren. - Die Teilnehmer*innen können in projektorientierter Gruppenarbeit gemeinsam Herausforderungen der Projektaufgaben diskutieren, mögliche alternative Vorgehensweisen identifizieren und begründete Vorgehensweisen festlegen, umsetzen und evaluieren.								
3	Inhalte							
Die Veranstaltung beinhaltet folgende Themenbereiche:								
Grundlagenteil:								
- Einführung in die Programmiersprache Python und ausgewählte Bibliotheken zur Vorbereitung und Manipulation von Daten - Grundbegriffe der Künstlichen Intelligenz (Agententheorie, Umgebungen, Anwendungsbereiche, Suchverfahren, Markov Decision Processes, Constraint Satisfaction Problems, etc.) - Einführung in maschinelles Lernen (neuronale Netze, Lernparadigmen, Anwendungen etc.)								
Hauptteil:								
- Klassische Optimierungsverfahren der KI und zugehörige Lösungsansätze am Beispiel des Traveling Salesman Problems - Einführung in das Gebiet des Natural Language Processings mit Fokus auf autonomen textbasierten Dialogsystemen (chatbots) und Sentiment Analysis								

	- Reinforcement Learning: aktuelle Verfahren und Grenzen sowie exemplarische Anwendung in den Bereichen Optimierungsverfahren und autonome textbasierte Dialogsysteme - Betrachtung ausgewählter Themen aus dem Bereich KI Ethik mit Fokus auf Anwendungen und Relevanz in der Wirtschaftsinformatik Projektorientierter Teil: - Projektorientierte praktische Anwendung der erlernten Inhalte auf konkrete, im Bereich der Wirtschaftsinformatik relevante Themenbereiche. Mögliche in der Lehrveranstaltung auszuwählende Beispiele: Entwicklung autonom agierender Chatbots im Kundensupport, Lösung konkreter Optimierungsprobleme beispielsweise in der Lagerhaltung, etc. Hierbei sollen in der praktischen Anwendung die in der Lehrveranstaltung betrachteten Verfahren angewendet und bewertet werden.
4	Lehrformen - Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion - vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation
5	Teilnahmevoraussetzungen Siehe jeweils gültige Prüfungsordnung (BPO/MPO) des Studiengangs.
6	Prüfungsformen Die Prüfung besteht aus zwei Teilen: - Mündliche Prüfung (30 Minuten) mit einem Anteil von 50% an der Gesamtnote - Projektarbeit mit mündlicher Prüfung mit einem Anteil von 50% an der Gesamtnote
7	Voraussetzungen für die Vergabe von Kreditpunkten - Bestandene mündliche Prüfung - Bestandene Projektarbeit Jeweils mit der Note 4.0 oder besser.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sebastian Bab Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Auswahl an Literaturhinweisen: Stuart Russell und Peter Norvig, Artificial Intelligence: A Modern Approach, Global Edition, Pearson 2021

Nummer		Bezeichnung						
46863		Autonome mobile Systeme						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Autonome mobile Systeme			Vorlesung	60			2
-	Autonome mobile Systeme			Übung	20			1
-	Autonome mobile Systeme			Praktikum	15			1
2 Lernergebnisse (learning outcomes) / Kompetenzen								
<u>Fach- und Methodenkompetenz:</u>								
Nach Abschluss der Lehrveranstaltung sind die Studierenden in der Lage,								
• Methoden und Algorithmen der autonomen mobilen Systeme zu verstehen und anzuwenden • Zustandsregler und Zustandsbeobachter zu entwerfen und zu implementieren • Algorithmen zur Zustandschätzung dynamischer Systeme anzuwenden • Algorithmen zur Lokalisierung, Pfadplanung und Kollisionsvermeidung autonomer mobiler Systeme anzuwenden und zu implementieren								
<u>Fachübergreifende Methodenkompetenz:</u>								
• Analyse von dynamischen Systemen • Mathematische Modellbildung dynamischer Systeme • Simulation von dynamischen Systemen mit Matlab/Simulink								
3 Inhalte								
• Grundlagen dynamischer Systeme • Beschreibung zeitdiskreter Systeme, z-Transformation • Zustandsraumdarstellung • Regelung mittels Zustandsrückführung • Zustandsbeobachtung • Zustandsschätzung dynamischer Systeme • Methode der kleinsten Fehlerquadrate • Kalman Filter, Erweitertes Kalman Filter, Unscented Kalman Filter • Monte Carlo Methoden • Autonome mobile Systeme • Grundlagen der Lokalisierung mobiler Systeme • Lokalisierung mittels Kalman Filter • Lokalisierung mittels Partikel Filter • Umgebungsmodelle und Kartierung • Navigation und Pfadplanung • Kollisionsvermeidung • Ausgewählte Problemstellungen aus aktuellen Forschungsprojekten								
4 Lehrformen								
• Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • vorlesungsbegleitendes Praktikum • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation								

5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit oder mündliche Prüfung (gemäß akt. Prüfungsplan) • semesterbegleitende Prüfungsleistungen
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Christof Röhrig Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Sebastian Thrun, Wolfram Burgard, Dieter Fox: Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), MIT Press, 2005 • Siegwart, Roland; Nourbakhsh, Illah R.: Introduction to Autonomous Mobile Robots, MIT Press, 2nd Edition, 2011 • Karsten Berns, Ewald von Puttkamer: Autonomous Land Vehicles: Steps towards Service Robots, Vieweg + Teubner Verlag, 2009 • Hertzberg, Joachim; Lingemann, Kai; Nüchter, Andreas: Mobile Roboter - Eine Einführung aus Sicht der Informatik, Springer Vieweg Verlag, 2012 • Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, Sebastian Thrun: Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents), MIT Press, 2005 • Unbehauen, Heinz: Regelungstechnik II, Vieweg Verlag, 9. Auflage, 2007 • Lunze, Jan: Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung, Springer Verlag, 6. Auflage, 2010

Nummer		Bezeichnung						
46001		Big Data Analytics						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
						60 h	120 h	
-	Big Data Analytics			Vorlesung	60			2
-	Big Data Analytics			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreicher Teilnahme an den Modulveranstaltungen und dem Abschließen der Projektarbeit sind die Studierenden in der Lage ...								
• Anforderungen an Big Data Anwendungen zu beurteilen und geeignete Verfahren zur Datenanalyse auszuwählen • Big Data Anwendungen von klassischen Anwendungen abzugrenzen und zu erläutern • anhand von Anwendungsszenarien zu entscheiden, welche Verfahren der Datenanalyse großer Datenmengen (sowohl statischer als auch Echtzeitdaten) in Big Data Anwendungen eingesetzt werden • exemplarisch Big Data Anwendungen zu konzipieren, zu entwerfen, zu implementieren und kritisch zu reflektieren. • die Grundlagen der Datenanalyse von großen Datenmengen und Echtzeitdaten zu erklären								
3	Inhalte							
• Grundlagen zum Thema Big Data • Herausforderungen und Architektur von Big Data Anwendungen • Grundlagen von Echtzeitanwendungen im Big Data Kontext und deren Besonderheiten • Grenzen von klassischen Lösungskonzepten im Big Data Kontext (CAP-Theorem, vertikales / horizontales Skalieren), NoSQL Datenbanken und Datenvisualisierung • Ethische, soziale und wirtschaftliche Bewertung von Big Data-Anwendungen • Grundlagen und Verfahren zur Datenanalyse von großen Datenmengen • Verfahren zum Verteilen von Daten (MapReduce, RDD) • Management Rechenclustern und verteilte Datenhaltung • Methoden und Technologien der Datenanalyse großer Datenmengen • Grundlagen und Verfahren zur Datenanalyse in Echtzeitsystemen • Konzeption und Entwurf von Datenstrom Anwendungen (Datenstrom-Managementsysteme, Datenstrom-Designmuster) • Methoden und Technologien der Datenstromanalyse (Streaming Analytics) • Anomalie-Erkennung und Überwachung von Datenströmen • Unterschiede und Gemeinsamkeiten bei der Datenanalyse vor großen Datenmengen bzw. Daten in Echtzeitsystemen, insbesondere der betrachteten Verfahren								
4	Lehrformen							
• Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitendes Praktikum • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • vorlesungsbergleitendes Projekt								
5	Teilnahmevoraussetzungen							
Keine								
6	Prüfungsformen							
Die Modulprüfung besteht aus zwei Teilen:								

	1. zu 60% aus einer mündlichen Prüfung (20-30 Minuten, gemäß akt. Prüfungsplan), in der die Studierenden Anwendungsszenarien analysieren, verschiedene theoretische Grundlagen erläutern und situativ anwenden können und 2. zu 40% aus einer semesterbegleitenden Prüfungsleistung (Projektarbeit mit abschließendem Referat), mit der die Studierenden zeigen sollen, dass sie die in der Vorlesung gelernten Technologien anwenden, reflektieren und bewerten können.
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung und bestandene projektbezogene Arbeit (mit Dokumentation und Präsentation), jeweils mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Klaus Kaiser Prof. Dr. Inga Saatz Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Kleppmann, Designing data-intensive applications, O'Reilly Media (2017) • Bifet, Machine learning for data stream, MIT-Press (2017) • Ellis, Real-time analytics, Wiley & Sons (2014) • Parsian, Data Algorithmns, O'Reilly Media (2015) • Arundel und Domingus, Cloud Native DevOps mit Kubernetes, dpunkt.verlag (2019) Weitere aktuelle Literatur wird in der Vorlesung bekannt gegeben.

Nummer		Bezeichnung					
46862		Design and Modeling of Complex Software Architectures					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Entwurf und Modellierung komplexer Software-Architekturen / Design and Modeling of Complex Software Architectures		Vorlesung	60			4
-	Entwurf und Modellierung komplexer Software-Architekturen / Design and Modeling of Complex Software Architectures		Praktikum	15			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
<i>Wissen und Verstehen:</i> Nach Abschluss dieses Moduls werden die Studierenden in der Lage sein,							
• Grundprinzipien des Softwareentwurfs zu unterscheiden, • relevante Werkzeuge und Methoden für den domänenorientierten Entwurf zu unterscheiden und zu kategorisieren, • aktuelle Forschungsansätze zur Modellierung von Softwarearchitekturen zu benennen und einzuordnen.							
<i>Einsatz, Anwendung und Erzeugung von Wissen:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• eine komplexe Problemstellung/Domäne zu analysieren und in Teilproblem/-domänen zu zerlegen, • eine komplexe Software-Entwurfsaufgabe im Rahmen eines mehrwöchigen Projekts zu realisieren, • adäquate Prinzipien des Softwaredesigns auszuwählen und auf konkrete Anwendungsszenarien anzuwenden, • zentrale Muster auf Ebene der Makro- und Mikroarchitektur zu unterscheiden, zu analysieren und anzuwenden, • geeignete Methoden für den domänengetriebenen Entwurf auszuwählen, zu kombinieren und anzuwenden.							
<i>Kommunikation und Kooperation:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• Lösungen kooperativ im Team zu entwickeln und umzusetzen, • geeignete Methoden für die interdisziplinäre Entwicklung von Lösungen auszuwählen und anzuwenden, insbesondere zusammen mit Domänenexpert*innen ohne technischen Hintergrund, • ihre Ideen und Lösungen in verschiedenen Formaten wie Gruppenpräsentationen, Code Reviews, Lightning Talks oder Pitches insbesondere vor einem Fachpublikum (z.B. Gästen/Partner*innen aus der Industrie oder aus Forschungsprojekten) zu präsentieren, zu erläutern und zu diskutieren.							
<i>Wissenschaftliches Selbstverständnis/Professionalität:</i> Nach Abschluss dieses Moduls sind die Studierenden in der Lage,							
• Best Practices aus Industrie und Wissenschaft für den Software-Entwurf auszuwählen und anzuwenden, • Feedback, insbesondere von nicht-technischen Domänenexpert*innen, zu reflektieren und zu bewerten und das erhaltene Feedback selbstständig zur Verbesserung ihrer Lösungskonzepte umzusetzen.							

3	Inhalte <i>Modulbeschreibung:</i> In diesem Modul vertiefen Studierende ihre Kompetenzen im Entwurf von Software-Architekturen komplexer Systeme. Die Studierenden lernen, wie sie durch Auswahl und Anwendung geeigneter Prinzipien, Muster und Methoden eine skalierbare, robuste und wartbare Software-Architektur domänengetrieben entwerfen können. Die Analyse und Diskussion solcher Software-Architekturen erfolgt anhand von Praxisbeispielen und konkreten Lösungen aus Forschungsprojekten. <i>Modulstruktur:</i> Das Modul umfasst die folgenden Themen: 1. Kurze Wiederholung des Bachelor-Stoffs zum Software-Entwurf (z.B. Entwurfsmuster nach Gamma et al., Separation of Concerns, Schichtenarchitektur) 2. Vertiefte Aspekte des Software-Entwurfs: 1. Prinzipien (z.B. Lose Kopplung - hohe Kohäsion, SOLID) 2. Architekturmuster (z.B. Ports and Adapter, CQRS) 3. Methoden (z.B. Domain-Driven Design, WAM-Ansatz) 3. Charakteristika und Muster moderner Architekturstile (z.B. modulare Architekturen, ereignisbasierte Architekturen, Microservice-Architekturen) 4. Modellgetriebener Entwurf, Entwicklung und Rekonstruktion von Software-Architekturen
4	Lehrformen • Flipped/Inverted Classroom: • Online-E-Learning-Materialien mit interaktiven Folien und Videos (asynchrones Selbststudium) • Interaktive Präsenzveranstaltungen für Aufgaben und Übungen anhand von Praxis- und Forschungsbeispielen (z.B. Coding, Gruppenübungen, Lightning Talks), für zusätzliche Vertiefung und zur Beantwortung und Diskussion von Fragen • Projektorientiertes Praktikum: Projektaufgabe, die über das gesamte Semester in Teams bearbeitet wird • Gastvorträge mit Experten und aktuellen Themen aus Forschung und Industrie
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (Umfang: 60%, Dauer: 60-90 Minuten) oder mündliche Prüfung (Umfang 60%, Dauer: 20-25 Minuten); semesterbegleitende Prüfungsleistungen (projektbezogene Arbeit, Umfang: 40%)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan); bestandene semesterbegleitende Prüfungsleistungen
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Vernon, Vernon (2016): Domain-Driven Design Distilled, Addison-Wesley • Evans, Eric (2003): Domain-Driven Design: Tackling Complexity in the Heart of Software, Addison-Wesley • Richardson, Chris (2018): Microservice Patterns: With examples in Java, Manning • Martin, Robert C. (2017): Clean Architecture: A Craftsman's Guide to Software Structure and Design, Pearson

- Lilienthal, Carola (2019): Sustainable Software Architecture: Analyze and Reduce Technical Debt; dpunkt.verlag
- Bass, Len; Clements, Paul; Kazman, Rick (2021): Software Architecture in Practice, SEI Series in Software Engineering, Fourth Edition, Addison-Wesley Professional
- Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1994): Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley
- Combemale, Benoit; France, Robert; Jézéquel, Jean-Marc; Rumpe, Bernhard; Steel, James; Vojtisek, Didier (2016): Engineering Modeling Languages. CRC Press
- Rademacher, Florian (2022). A language ecosystem for modeling microservice architecture, Phd Thesis, <https://dx.doi.org/doi:10.17170/kobra-202209306919>

Nummer		Bezeichnung					
46816		Echtzeitsysteme					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
	-	Echtzeitsysteme	Vorlesung	60			4
-	Echtzeitsysteme	Praktikum	15			2	
2	Lernergebnisse (learning outcomes) / Kompetenzen						
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:							
Wissen und Verstehen:							
- die technischen Parameter, die für die Auswahl von Planungsverfahren relevant sind, zu ermitteln und anhand derer ein geeignetes Verfahren situationsgerecht auszuwählen, zu implementieren und zu validieren - die besonderen zeitlichen Aspekte von synchronisierten Prozessen und von verteilten Echtzeitsystemen in Entwurf und Implementierung zu adressieren.							
Einsatz, Anwendung und Erzeugung von Wissen:							
- Prozessmengen hinsichtlich Ausführbarkeit zu analysieren und deren Zeitverhalten zu bewerten - Systeme zu konzipieren, die in ihrem zeitlichen Verhalten vorhersagbar sind							
Kommunikation und Kooperation:							
- sich wissenschaftliche Erkenntnisse zu erschließen und - technische Entwürfe von Echtzeitsystemen zu präsentieren und - diese in interdisziplinären Teams umzusetzen.							
Wissenschaftliches Selbstverständnis / Professionalität:							
- die Entwurfsphilosophie hinter Echtzeitsystemen zu artikulieren - ethische und gesellschaftliche Auswirkungen von sicherheitskritischen Echtzeitsystemen zu reflektieren							
3	Inhalte						
- Der Zeitbegriff: mathematische Reduktion der Zeit, Aufbau und Eigenschaften einer Uhr, Zeitstandards - Einführung in die Planung: Prozessparameter, WCET, Brauchbarkeit, Prozesspräzedenz, Prozessanomalien - Prioritätsbasierte Planungsverfahren für aperiodische Prozesse: Earliest Due Date, Earliest Deadline First, Least Laxity First - Prioritätsbasierte Planungsverfahren für periodische Prozesse: Rate und Deadline Monotonic Scheduling, Einplanungstests (LL-Test, kritisches Intervall, RT-Test), Earliest Deadline First - Zeitbasierte Planungsverfahren: äußerer und innerer Zyklus, Anforderungen nach Baker&Shaw, Implementierung des Cyclic Executives - Planungsverfahren für synchronisierte Prozesse: Prioritätsumkehr, Verfahren (Nicht-präemptive kritische Abschnitte, Prioritätsvererbung, Prioritätsobergrenzen), Berechnung von Blockierungszeiten - Echtzeitbetriebssysteme: u.a. Architektur, Scheduler, Umgang mit Interrupts, Fallstudie - Verteilte Systeme: Synchronisation von Uhren, Echtzeiteigenschaften verschiedener Medienzugriffsverfahren, aktuelle Echtzeit-Protokolle							

4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • mündliche Prüfung, 30min (70%) • semesterbegleitende Prüfungsleistung, als projektbezogene Arbeit mit Präsentation (30%)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Erik Kamsties Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Dieter Zöbel. Echtzeitsysteme Grundlagen der Planung, Springer, 2020. • Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications, Springer, 2022 • Jane Liu. Real-Time Systems, Prentice Hall, 2000. • Peter Marwedel. Eingebettete Systeme - Grundlagen Eingebetteter Systeme in Cyber-Physikalischen Systemen, Springer, 2021. • Burns, A., Wellings, A.; Real-Time Systems and Programming Languages; Pearson, 2009. • Heinz Wörn und Dieter Brinkschulte. Echtzeitsysteme, Springer, 2005.

Nummer		Bezeichnung						
46850		Entwicklung software-intensiver Systeme						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Entwicklung software-intensiver Systeme			Vorlesung	60			2
-	Entwicklung software-intensiver Systeme			Übung	20			2
2 Lernergebnisse (learning outcomes) / Kompetenzen								
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:								
Wissen und Verstehen								
- zentrale Konzepte, Methoden und aktuelle Entwicklungen in der Systementwicklung software-intensiver Systeme zu analysieren und in einen fachlichen Kontext einzuordnen - Architektur- und Designprinzipien software-intensiver Systeme kritisch zu reflektieren und deren Auswirkungen auf Skalierbarkeit, Wartbarkeit und Sicherheit zu bewerten								
Einsatz, Anwendung und Erzeugung von Wissen								
- moderne Methoden des Konfigurationsmanagements, der Entwicklung, Integration und Bereitstellung software-intensiver Systeme praktisch anzuwenden und deren Effizienz in realen Projekten zu evaluieren - Test-, Verifikations- und Validierungsverfahren gezielt auszuwählen, anzupassen und auf komplexe Systementwicklungsprojekte zu übertragen - innovative Entwicklungsansätze und Technologien im Bereich DevOps, CI/CD und Dependable Systems zu erforschen, kritisch zu hinterfragen und für spezifische Anwendungsfälle zu adaptieren - wissenschaftlich fundierte Entscheidungen bei der Entwicklung sicherheitskritischer Systeme zu treffen und deren gesellschaftliche sowie ethische Implikationen abzuwägen								
Kommunikation und Kooperation								
- Ergebnisse technischer Analysen und Systementwicklungsprojekte zielgruppengerecht zu dokumentieren, zu präsentieren und kritisch zu diskutieren - in interdisziplinären Teams effektiv zusammenzuarbeiten, Entwicklungsstrategien abzustimmen und Konfliktpotenziale in der Systementwicklung zu reflektieren								
Wissenschaftliches Selbstverständnis / Professionalität								
- ihr eigenes professionelles Handeln in der Systementwicklung unter Berücksichtigung wissenschaftlicher Methoden und ethischer Standards zu reflektieren und weiterzuentwickeln - sich selbstständig neues Wissen zu aktuellen Trends und Technologien der Systementwicklung anzueignen und dieses in praktischen Projekten anzuwenden								

3	Inhalte • Charakterisierung und Abgrenzung Software-intensiver Systeme • Anwendungen, Referenzarchitekturen und Lebenszyklen in verschiedenen Industrien • Systemorientierung (Systembegriff, Sytemkontexte, Systemebenen/Bausteine) • Entwicklungsprozesse für komplexe Software-intensive Systeme (traditionell und agil) • Architektur und Design: Prinzipien und Best Practices für Softwarearchitekturen. • Implementierung und Integration: DevOps-Ansätze, CI/CD-Pipelines, Virtualisierung und Containertechnologien. • Test, Verifikation und Validierung komplexer und eingebetteter Systeme aus Hard- und Software • Deployment: Bereitstellungsstrategien mit modernen Tools • Funktionale Sicherheit: Risiko, -reduktion, SIL und Sicherheit, Normen, Kultur in verschiedenen Industrien • Dependable Systems: Konzepte zu Zuverlässigkeit, Verfügbarkeit, Wartbarkeit und Sicherheit Software-intensiver Systeme. • Architekturmuster und Prozesse für Verfügbarkeit und Funktionale Sicherheit
4	Lehrformen • Das Modul wird in Seminarform abgehalten und kombiniert Vorlesungen, praktische Übungen und Projektarbeit. • Lösung von praxisorientierten Übungsaufgaben, einzeln oder in Teams • Bearbeitung von Programmieraufgaben am Computer in Einzel- oder Gruppenarbeit • vorlesungsbegleitende Projektarbeit mit Abschlusspräsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Mündliche Prüfung (Umfang 100%, Dauer 20-30 min)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Sommerville: Software Engineering ,10. Aufl. , Pearson Deutschland, 2018. • Liggesmeyer, Rombach: Software Engineering eingebetteter Systeme, Spektrum Akademischer Verlag, 2005. • Bass, Clements, Kazman: Software Architecture in Practice, 3rd ed., Addison Wesley, 2013. • Delligatti: SysML Distilled, Addison Wesley, 2014. • Zörner: Software-Architekturen dokumentieren und kommunizieren, 2. Aufl., Hanser, 2015. • Starke, Hruschka: Arc42 in Aktion, Hanser, 2016. • Grünfelder: Software-Test für Embedded Systems, 2. Aufl., dpunkt Verlag, 2017. • Grenning: Test Driven Development of Embedded C, O'Reilly, 2011. • MISRA, MISRA C:2012, Guidelines for the use of the C language in critical systems, HORIBA MIRA Limited, 2015. • Hobbs: Embedded Software Development for Safety-Critical Systems, CRC Press (2016). • verschiedene Normen (werden in der Veranstaltung bekannt gegeben)

Nummer		Bezeichnung					
46914		Fundamente der Programmierung					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Konzepte in Programmiersprachen / Fundamente der Programmierung		Vorlesung	60			4
-	Konzepte in Programmiersprachen / Fundamente der Programmierung		Praktikum	15			2
2 Lernergebnisse (learning outcomes) / Kompetenzen							
Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierende in der Lage:							
<i>Wissen und Verstehen:</i>							
- wesentliche Eigenschaften funktionaler Programmiersprachen zu benennen. - das Konzept der Funktionsabschlüsse (Closure) zu erläutern. - den syntaktischen Aufbau von Lambda-Ausdrücke darzustellen. - den Normalisierungsprozess von Lambda-Ausdrücken zu erläutern. - grundlegende Programmkonstrukte als Lambda-Ausdrücke zu formulieren. - unterschiedliche Ansätze der Typisierung von Lambda-Ausdrücken zu erläutern. - die Eigenschaften von Dependent Typs zu erläutern.							
<i>Einsatz, Anwendung und Erzeugung von Wissen:</i>							
- die Grundregeln des natürlichen Schließens anzuwenden - Schlussregeln der konstruktiven Logik den Grundkonstrukten typisierter Programme gegenüberzustellen - elementare Beweise als Programme in einem Beweisunterstützungssystem zu schreiben							
<i>Kommunikation und Kooperation:</i>							
- zwischen unterschiedlichen formalen Beschreibungen zu vermitteln - präzise zu formulieren - logisch zu argumentieren							
<i>Wissenschaftliches Selbstverständnis / Professionalität:</i>							
- Anforderungen formal zu spezifizieren - die Zuverlässigkeit von Teilsystemen über Korrektheitsbeweise sicherzustellen - Abstraktionskonzepte zu beurteilen							
3 Inhalte							
<i>Modulbeschreibung:</i>							
Auf Basis des Curry-Howard-Isomorphismus werden fundamentale Konzepte der Programmierung diskutiert.							
<i>Modulstruktur:</i>							
- Grundlagen der praktischen Funktionalen Programmierung - Lambda-Kalkül							

	• Typisierung im Lambda-Kalkül • Natürliches Schließen • Propositions-as-Types • Dependent Types • Programmierung von Beweisen • Formalisierung ausgewählter Grundlagen der Programmierung
4	Lehrformen • Seminaristische Vorlesungen, mit Tafelanschrieb und Live-Programmierung • Situationsbedingter Wechsel zwischen Vorlesung, praktischen Übungen und Diskussion von Lösungen
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Schriftliche Klausurarbeit (60 - 90 Minuten) oder mündliche Prüfung (gemäß akt. Prüfungsplan)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Klausurarbeit oder bestandene mündliche Prüfung (gemäß akt. Prüfungsplan)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Dirk Wiesmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Bove A., Dybjer P.; "Dependent types at work. In Language Engineering and Rigorous Software Development", LNCS 5520, 57-99, Springer, Berlin, 2008 • Hindley, J.R., Seldin J.P.; "Lambda-Calculus and Combinators, an Introduction", Cambridge University Press, Cambridge, UK, 2008 • Hutton G.; "Programming in Haskell. Cambridge", University Press, Cambridge, UK, 2016 • Wadler P.; "Proposition as types", Commun. ACM, 58(12):75-84, Nov, 2015 • Wadler P., Kokke W., Siek J.G.; "Programming language foundations in Agda", 2021

Nummer		Bezeichnung					
46860		Internet der Dinge					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Internet der Dinge		Vorlesung	60			4
-	Internet der Dinge		Übung	20			2
-	Internet der Dinge		Praktikum	15			1
2	Lernergebnisse (learning outcomes) / Kompetenzen Fach- und Methodenkompetenz: Nach Abschluss der Lehrveranstaltung sind die Studierenden in der Lage, • Das Konzept des Internets der Dinge (Internet of Things, IoT) einzuordnen und von Machine 2 Machine Communication (m2m) und Industrie 4.0 abzugrenzen • Anwendungsfelder von IoT zu kennen und derer Anforderungen an Technologie und Architektur anzugeben • Technologien, Architekturen und Protokolle des IoT zu verstehen und vorhandene IoT-Systeme zu analysieren • Drahtlose Funktechnologien wie UWB, LoRaWAN, Z-Wave, ZigBee, Bluetooth Smart hinsichtlich Reichweite, Datenrate, Interoperabilität und Stromverbrauch einzuordnen • Routingprotokolle für die Ad-Hoc-Vernetzung wie OLSR, AODV, DSR zur verstehen und in eigene Systeme zu implementieren • Architekturen, Technologien und Protokolle für vorgegebene IoT-Anwendungen auszuwählen und in eigenen Systemen zu implementieren • Neue Architekturen und Routingprotokolle für spezielle IoT-Anwendungen zu entwerfen und zu implementieren						
3	Inhalte • Einleitung • Motivation, Definition, Abgrenzung zu m2m, Industrie 4.0 • Anwendungsgebiete und deren Anforderungen • Übersicht Schichtenmodelle: ISO/OSI, TCP/IP, IPv6 und 6LoWPAN, Bluetooth Smart • Übersicht Funkübertragung: ISM-Bänder, lizenzierte Bänder, UWB • Einordnung Technologien: IEEE 802.15.4, Bluetooth Smart, RFID, LoRaWAN • Architekturen und Protokolle des IoTs • Protokolle der Anwendungsschicht: CoAP, MQTT, GATT • Protokoll-Gateways der Anwendungsschicht: REST-HTTP/CoAP, REST-HTTP/GATT • Topologien: Stern u. Baum-Topologien mit zentralem Gateway, Mesh-Networking, Multi-Gateway • Routing-Protokolle: OLSR, AODV, DSR • IPv6, 6LoWPAN • Grundlagen der digitalen Kommunikation • Abtastung von Signalen, Nyquist Abtasttheorem • Kodierung, Modulation, Kanalkapazität Shannon Fano • Mehrfachzugriffsverfahren: ALOHA, CSMA/CA, FDMA, TDMA, CDMA, OFDM						

	• Grundlagen Funkübertragung: Antennen, Freiraumdämpfung, Fresnelsche Zone, • Beispielhafte Anwendungsgebiete • Smart Home • Szenarien und deren Anforderungen • Technologien: Z-Wave, ZigBee, EnOcean • Beispielhafte Umsetzung anhand eines aktuellen AAL-Forschungsprojektes • Logistik • Szenario Tracking & Tracing • Technologien: RFID, LoRaWAN, UWB • Beispielhafte Umsetzung anhand eines aktuellen Forschungsprojektes
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • vorlesungsbegleitendes Praktikum • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit oder mündliche Prüfung (gemäß akt. Prüfungsplan) • semesterbegleitende Prüfungsleistungen
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Christof Röhrig Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Jan Höller: From machine-to-machine to the internet of things - introduction to a new age of intelligence, Elsevier, 2014 • Peter Waher: Learning Internet of Things - explore and learn about Internet of Things with the help of engaging and enlightening tutorials designed for Raspberry Pi, Packt Publishing, Birmingham, 2015 • Ralf Gessler, Thomas Krause: Wireless-Netzwerke für den Nahbereich, Eingebettete Funksysteme, Vergleich von standardisierten und proprietären Verfahren, Vieweg+Teubner, 2009 • Martin Meyer: Kommunikationstechnik, Konzepte der modernen Nachrichtenübertragung, Vieweg+Teubner, 4. Auflage, 2011. • Andrew S. Tanenbaum, David J. Wetherall: Computernetzwerke, 5. Auflage, Pearson Studium, 2012

Nummer		Bezeichnung						
46839		Machine Learning						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
			Kontakt- zeit 60 h			Selbst- studium 120 h		
-		Maschinelles Lernen / Machine Learning		Vorlesung	60			4
-		Maschinelles Lernen / Machine Learning		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Die Lehrveranstaltung beschäftigt sich mit der Entwicklung und Analyse von maschinellen Lernverfahren in Anwendungen der Informatik, der Medizinischen Informatik bzw. für allgemeine Informationssysteme. Nach erfolgreicher Teilnahme an der Lehrveranstaltung haben die Studierenden die folgenden Kompetenzen erworben: Kenntnisse (Wissen): - kennen die wichtigsten Fachbegriffe des maschinellen Lernens und können damit lernende Systeme erklären. - Sie können den Einsatz maschineller Lernverfahren für eigene Anwendungsaufgaben implementieren und evaluieren. Dazu kennen die Studierenden typische Anwendungen dieser Methoden. - kennen Projektmanagementmethoden für Anwendungen des Maschinellen Lernens wie CRISP-DM - kennen Erklärungskomponenten für maschinelles Lernen und können die Ergebnisse interpretieren - kennen typische Probleme des maschinellen Lernens wie Overfitting und Information Leakage und können diese vermeiden - kennen die theoretischen Grenzen maschineller Lernsysteme und können diese formal beschreiben und zur Abschätzung der Grenzen eigener Anwendungen nutzen. Fertigkeiten - können maschinelle Lernsysteme für konkrete Anwendungskontexte der Informatik entwerfen, implementieren und analysieren. - können ethische Grundlagen und Grenzen maschineller Lernsysteme hinterfragen und diskutieren - können geeignete Methoden des maschinellen Lernens für Anwendungen auswählen - können einfache Deep Learning Lösungen mit JupyterHub implementieren - können Miniprojekte mit Industriebezug selbstständig im Team planen, durchführen, analysieren und dokumentieren. Kompetenzen (Selbst- und Sozialkompetenz) - können Ideen und Lösungsvorschläge mündlich und schriftlich formulieren - können Aufgaben in Übungen und Praktika selbstständig lösen und die Ergebnisse präsentieren - können sich theoretische Inhalte zum Thema Maschinelles Lernen aus wissenschaftlicher Literatur selbstständig erarbeiten und präsentieren - können in den Übungs- und Projektphasen kooperativ Lösungen erarbeiten - können in den Projektphasen kooperativ Aufgaben zur Lösung planen, verteilen und gemeinsam durchführen - können in Diskussionen zielorientiert argumentieren und mit Kritik sachlich umgehen - können Ergebnisse von Gruppenarbeiten gemeinsam präsentieren - können Projektergebnisse bewerten und Verbesserungsvorschläge formulieren - können Missverständnisse zwischen Gesprächspartnern erkennen und abbauen								

3	Inhalte - Grundbegriffe des Maschinellen Lernens - Nutzung von KNime für das Maschinelle Lernen - Design von Evaluationsstudien für Maschinelle Lernverfahren und Durchführung solcher Studien - Lineare Modelle - Verschiedene Modelle überwachter und unüberwachter Neuronaler Netzwerke - Lernverfahren für strukturierte Daten: Support Vektor Maschine, Entscheidungsbäume, Random Forest, Gradient Boosting Machines (GBM) - Nächster Nachbarverfahren, Lazy Learning und Embeddings - Bayesische Netzwerke - Unüberwachte Lernverfahren (k-means) - Kombinationsmodelle (Ensembles, Boosting Machines) - Deep Learning Modelle (Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), Transformer Architekturen z.B. BERT, Visual Transformer) - Deep Learning Konzepte - Transferlernen, Datenvermehrung, Generative Adversarial Networks (GAN) - Deep Learning - Parallelisierung mit GPUs, Implementierung auf mobilen Plattformen mit geringen Ressourcen - Large Language Models und Nutzung von Embeddings, Retrieval Augmented Generation (RAG) - Theoretische Konzepte: Bias-Varianz Dilemma, No Free Lunch Theorem - Erklärbarkeit von Modellen - Anwendungen mit Daten verschiedener Modalitäten (Text, Image, Sound), Word2Vec, FastText, Transformer - Methoden zur Verbesserung der Generalisierungsleistung (Regularisierung, Feature Selektion, Dimensionsreduktion, Komplexitätsanpassung) - Problemlösung am Beispiel studiengangsbezogener Miniprojekte aus industriellen Anwendungen (studentische Miniprojekte in 2-3er Teams)
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation • Umgedrehter Unterricht (inverted classroom)
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit (70% der Prüfungsleistung) • semesterbegleitende Prüfungsleistungen (30% der Prüfungsleistung)
7	Voraussetzungen für die Vergabe von Kreditpunkten • bestandene Klausurarbeit • erfolgreiche semesterbegleitende Prüfungsleistung (Miniprojekt mit Präsentation)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr.-Ing. Christoph M. Friedrich Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur - I. Witten, E. Frank, M. Hall und C. J. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4. Auflage, Morgan Kaufmann (2017) - elektronische Version im Intranet verfügbar - C. M. Bishop, Pattern Recognition and Machine Learning, Springer (2006)

- E. Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning), Revised and Updated Edition, MIT Press (2021)
- I. Goodfellow, Y. Bengio und A. Courville: Deep Learning, MIT Press (2016)

Nummer		Bezeichnung					
46852		Mobile Multi-Agenten-Systeme					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
	-	Verteilte und mobile Systeme / Mobile Multi-Agenten-Systeme	Vorlesung	60			4
	-	Verteilte und mobile Systeme / Mobile Multi-Agenten-Systeme	Praktikum	15			2
2							
Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:							
Wissen und Verstehen							
- zentrale Konzepte, Architekturen und Interaktionsmechanismen von mobilen Multiagentensystemen (MAS) zu erklären und sie von klassischen verteilten Systemen abzugrenzen. - Methoden der Koordination, Kooperation und Konsensfindung in mobilen MAS zu analysieren und deren Einsatz in realen Anwendungen zu bewerten.							
Einsatz, Anwendung und Erzeugung von Wissen							
- Planungs-, Lern- und Selbstorganisationsmechanismen in mobilen MAS zu entwerfen, zu implementieren und auf Problemstellungen wie Schwarmintelligenz oder autonome Systeme anzuwenden. - Mobile MAS mit geeigneten Simulationstools zu modellieren, zu testen und ihre Leistungsfähigkeit zu evaluieren.							
Kommunikation und Kooperation							
- wissenschaftliche Erkenntnisse und technische Konzepte zu mobilen MAS klar zu präsentieren und in interdisziplinären Teams anzuwenden. - ethische und gesellschaftliche Auswirkungen intelligenter mobiler MAS kritisch zu reflektieren und verantwortungsbewusst zu gestalten.							
Wissenschaftliches Selbstverständnis / Professionalität							
- sich eigenständig in aktuelle Forschungsthemen zu mobilen MAS einzuarbeiten, neue Fragestellungen zu entwickeln und geeignete Lösungsansätze abzuleiten. - wissenschaftlich fundierte Entscheidungen bei der Entwicklung und Anwendung von mobilen MAS zu treffen und dabei technische sowie ethische Rahmenbedingungen zu berücksichtigen.							
3							
Inhalte							
Grundlagen von Multiagentensystemen (MAS) - Definition und Konzepte von Agenten und Multiagentensystemen - Eigenschaften und Klassifikation von Agenten - Abgrenzung zu klassischen verteilten Systemen Agentenarchitekturen und -modelle - Reaktive, deliberative und hybride Architekturen - Entscheidungsfindung und Autonomie von Agenten - Architekturen für intelligente Agenten Kommunikation und Interaktion in MAS - Agent Communication Languages (ACL) - Sprechakttheorie und Ontologien							

	• Mechanismen der Koordination und Kooperation • Konsensfindung und Verhandlung in MAS • Nash-Gleichgewicht und kooperative Strategien • Verhandlungsprotokolle (Aufgaben- und Wertorientierung) • Anwendungen in Auktionen und elektronischen Märkten • Planung und Lernen in Multiagentensystemen • Zentrale vs. verteilte Planung • Reinforcement Learning für Agentensysteme • Selbstorganisation und evolutionäre Algorithmen • Modellierung von Umgebungen für MAS • Einfluss von deterministischen und dynamischen Umgebungen • Simulation von Agentensystemen mit Mesa (Python) • Herausforderungen in offenen und komplexen Systemen • Kollektive Bewegung und Schwarmintelligenz • Prinzipien der Schwarmintelligenz (bspw. Boid-Modell) • Selbstorganisierte Formationen in Multiagentensystemen • Anwendungen: Drohnenschwärme, autonome Fahrzeuge, Such- und Rettungsmissionen • Anwendungsfälle und Fallstudien • Multiagentensysteme in Robotik und Logistik • Agentensysteme in der Spieleindustrie und Simulation sozialer Systeme • Intelligente Agenten für Smart Cities und IoT • Praktische Übungen und semesterbegleitendes Projekt • Implementierung von Agenten mit Mesa (Python) • Entwicklung eines kooperativen Multiagentensystems • Evaluierung verschiedener Interaktions- und Planungsmechanismen
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • Bearbeitung von Programmieraufgaben am Rechner in Einzel- oder Teamarbeit • studienbegleitende Prüfungsleistung mit abschließender Präsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • mündliche Prüfung [Umfang: 75%](20-30 min) • studienbegleitende Prüfungsleistung [Umfang: 25%]
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung einschließlich SBL mit der Note 4,0 oder besser
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund

10 Literatur

- Wooldridge, M.: *An Introduction to MultiAgent Systems*, 2nd ed., John Wiley & Sons, 2009
- Russell, S., & Norvig, P.: *Artificial Intelligence: A Modern Approach*, 4th ed, Pearson, 2021
- Shoham, Y., & Leyton-Brown, K.: *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*, Cambridge University Press, 2009
- Mesa Framework Documentation: <https://mesa.readthedocs.io>

Nummer		Bezeichnung						
46910		Requirements Engineering						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Requirements Engineering			Vorlesung	60			4
-	Requirements Engineering			Übung	20			2
2								
Lernergebnisse (learning outcomes) / Kompetenzen								
Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage:								
Wissen und Verstehen:								
- die Rolle des Requirements Engineering (RE) im Kontext heutiger Herausforderungen in Projekten zu erläutern, z.B. der „VUCA world“ (Volatility, Uncertainty, Complexity, Ambiguity) - verschiedene Kreativitäts- und Erhebungstechniken geeignet zu kombinieren um Anforderungen zu ermitteln, sowie Anforderungen geeignet zu dokumentieren und zu validieren - zu erläutern, wie verschiedene Aktivität im RE mit generativer KI unterstützt werden können und wo Grenzen liegen								
Einsatz, Anwendung und Erzeugung von Wissen:								
- Einen RE Prozess in eine Organisation einzupassen und Vorgehensweisen, Techniken und Tools im RE entsprechend den Rahmenbedingungen eines Projekts auszuwählen. - Anforderungen zielorientiert zu erheben, zu analysieren, zu spezifizieren und zu validieren. - Mit spezifischen Themen von Anforderungen wie Variabilität und Adaptivität umzugehen.								
Kommunikation und Zusammenarbeit:								
- Sich Techniken und Methoden im RE selbst zu erschließen und an andere zu vermitteln - effektiv mit verschiedenen Stakeholdern, z.B. Kunden, Entwicklern und Endbenutzern, zu kommunizieren, um Anforderungen zu entwickeln, zu verfeinern und zu validieren - in Teams zusammenzuarbeiten, um Konzepte und Lösungen zu entwickeln und dabei unterschiedliche Perspektiven und Interessen abzuwägen								
Wissenschaftliches Selbstverständnis / Professionalität:								
- RE Kompetenz in einem domänenübergreifenden Team aufzubauen und zu leiten - die ethische und professionelle Verantwortung zu erkennen, die mit der Umsetzung der Bedürfnisse der Stakeholder in erfolgreiche Systeme verbunden ist - Praktiken im RE kritisch zu bewerten und Verbesserungen zu identifizieren								
3								
Inhalte								
Requirements Engineering (RE) ist die erste Aktivität in der Software-, System- und Serviceentwicklung. Anforderungen sind der Ausgangspunkt für Design, Verifizierung und Validierung sowie für den Test und die Integration von Systemen. Konfigurations- und Änderungsanforderungsmanagement sind mit RE verbunden. Die Definition von Anforderungen und der strukturierte Umgang mit Anforderungen ist nach wie vor ein wichtiger Bereich für die Forschung zu Werkzeugen und Methoden – insbesondere für große und komplexe mechatronische Systeme. In diesem Modul erhalten die Studierenden spezifisches Wissen über den aktuellen Stand der Technik und die wichtigsten zukünftigen Herausforderungen im Bereich RE.								

	• Einführung in das Requirements Engineering • Definition, Relevanz und Herausforderungen • Einbettung abhängig von Systemtyp und Projekteigenschaften • Frameworks (z. B. Jacksons WRSPM-Modell) • Anforderungsermittlung • Stakeholder-Identifizierung • Kreativität und Innovation • Interviews, Fokusgruppen und Ethnographie • Brainstorming und kollaborative Workshops • Anforderungsdokumentation • Standards für Anforderungsspezifikationen (SRS) • Informelle Methoden: Prototypen, Storyboards • Modellierungsanforderungen: i-star, UML, User Stories • Tools: JIRA, Confluence, ReqIF • Validierung und Verifizierung • Qualitätsmerkmale: Vollständigkeit, Konsistenz, Richtigkeit • Prototyping und Benutzerfeedback • Strategien zum Testen von Anforderungen • Anforderungsmanagement • Priorisierungstechniken: z.B. MoSCoW, Kano, Weighted Scoring • Traceability • Auswirkungsanalyse für Änderungen • Versionierung und Änderungsmanagement • Verbesserung des RE-Prozesses • Fortgeschrittene Themen • Software-Produktlinien, adaptive Systeme und Crowd-basierte Systeme • Domänenspezifische Sprachen • Generative KI und natürliche Sprachverarbeitung im RE
4	Lehrformen Vorlesung mit Tafelanschrieb und Projektion
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • schriftliche Klausurarbeit, Dauer 90 min (70%) • semesterbegleitende Prüfungsleistung, als projektbezogene Arbeit mit Präsentation (30%)
7	Voraussetzungen für die Vergabe von Kreditpunkten • bestandene Klausurarbeit • erfolgreiches Miniprojekt (projektbezogene Arbeit)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Erik Kamsties Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund

10 Literatur

- Klaus Pohl. Requirements Engineering: Fundamentals, Principles and Techniques. Springer, 2025
- Klaus Pohl und Chris Rupp: Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level, 2015
- Brian Berenbach, Daniel Paulish, Juergen Kazmeier, Arnold Rudorfer. Software and Systems Requirements Engineering In Practice, McGraw-Hill, March 2009
- Klaus Pohl, Günter Böckle und Frank J. van der Linden. Software Product Line Engineering: Foundations, Principles and Techniques, Springer, Januar 2011
- Søren Lausen. Software Requirements - Styles and Techniques, Addison-Wesley, 2002.
- Ellen Gottesdiener. Requirements by Collaboration - Workshops for Defining Needs. Addison-Wesley, 2002

Nummer		Bezeichnung					
46000		Software-Entwicklungsprozesse					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Software-Entwicklungsprozesse		Vorlesung	60			2
-	Software-Entwicklungsprozesse		Praktikum	60			2
2 Lernergebnisse (learning outcomes) / Kompetenzen							
Nach der erfolgreichen Teilnahme an diesem Modul sind die Studierende in der Lage:							
Wissen und Verstehen:							
• ein hierarchisches Begriffsmodell zu Klassifizierung von Softwareentwicklungsprozesse zu beschreiben. • bekannte Prozessmodelle zu beschreiben. • Ansätze zur Bestimmung der Prozessqualität zu erläutern. • gegebene Prozessbeschreibungen zu klassifizieren. • die wesentlichen Unterschiede bekannte Prozessmodell zu erläutern. • den Einfluss unterschiedlicher Stakeholder in Entwicklungsprozessen zu beschreiben.							
Einsatz, Anwendung und Erzeugung von Wissen:							
• auf der Grundlage von Rahmenbedingungen ein geeignetes Vorgehensmodell auszuwählen. • Alternativen auf Grundlagen einer Risikobetrachtung zu bewerten. • auf Grundlage von Rahmenbedingungen ein Prozessmodell anzupassen bzw. zu konzipieren.							
Kommunikation und Kooperation							
• alternative Ansätze der Prozessgestaltung zu diskutieren. • den Informationsaustausch in Software-Projekten zu gestalten. • die Teamarbeit durch geeignete Prozesse zu unterstützen.							
Wissenschaftliches Selbstverständnis / Professionalität							
• ein Projektumfeld zu analysieren. • bestehende Prozesse zu beurteilen. • Software-Entwicklungsprozesse zu optimieren und zu gestalten.							
3 Inhalte							
• Vorgehensmodelle • Prozessmodelle • Merkmale bekannter Prozessmodelle • Prozess-Frameworks • Faktor Mensch • Ökonomie von Software-Projekten • Kommunikation in der Softwareentwicklung • Artefakte und Dokumentation • Prozess- und Produktqualität • Integration von Projekt-, Risiko-, Qualitäts-, Problem-, Änderungs- und Anforderungs-Management • Tailoring • Prozessentwurf							

4	Lehrformen • Seminaristische Vorlesung mit Tafelanschrieb und Projektion • Situationsbedingter Wechsel zwischen Vorlesung, praktischen Übungen und der Diskussion von Fallstudien
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Schriftliche Klausurarbeit (60 - 90 Minuten) oder mündliche Prüfung.
7	Voraussetzungen für die Vergabe von Kreditpunkten • Bestandene Klausurarbeit.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Dirk Wiesmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Ludewig K, Lichter H.; "Software Engineering: Grundlagen, Menschen, Prozesse, Techniken", dpunkt.verlag, Heidelberg, 2023 • Balzert H, Ebert Ch.; "Lehrbuch der Software-Technik, Software-Management", Spektrum Akademischer Verlag, Heidelberg, 2008 • Sommerville I.; "Software Engineering", Pearson, Boston, 2016 • Chrissis M.B., Konrad M., Shrum S; "CMMI for Development", Addison-Wesley, Upper Saddle River, NJ, 2011 • Dawson C, Dawson R.; "Software development process models: a technique for evaluation and decision-making", Knowl Process Manag. 21(1), 43-53, 2014

Nummer		Bezeichnung						
46908		Usability Engineering						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
englisch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Usability Engineering			Vorlesung	60			2
-	Usability Engineering			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen							
Wissen und Verstehen Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • systematisch Gestaltungslösungen zu entwickeln und auf deren Gebrauchstauglichkeit zu prüfen • Forschungsdesigns zu erstellen, durchzuführen und auszuwerten								
Einsatz, Anwendung und Erzeugung von Wissen Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • methodische Abwägungen vorzunehmen und Auswahlen zu begründen								
Kommunikation und Kooperation Durch die erfolgreiche Teilnahme am Modul werden Studierende • gemeinsam als Kleingruppe ein eigenes Forschungsvorhaben planen und umsetzen • das Vorgehen in einem gemeinsamen Projekt einüben								
Wissenschaftliches Selbstverständnis / Professionalität Nach erfolgreicher Teilnahme am Modul sind die Studierenden in der Lage • über die Methodik ihres Fachgebiets und darüber hinaus zu reflektieren und konkrete Arbeiten auf deren Methodik zu untersuchen								
3	Inhalte							
1. Grundbegriffe Usability 2. Bezüge und Anknüpfungspunkte zu Mensch-Computer-Interaktion 3. Usability Engineering – frühe Ansätze 4. Studiendesign und Evaluationsmethoden 5. Projektarbeit 6. Spezialkapitel: Usability bezogen auf einen aktuellen Forschungsgegenstand								
4	Lehrformen							
seminaristische Vorlesung mit Präsentation, Gruppenarbeit und Arbeitsaufträgen								
5	Teilnahmevoraussetzungen							
Keine								

6	Prüfungsformen Hausarbeit mit Referat (80%) Semesterbegleitleistung durch Arbeitsaufträge (20%)
7	Voraussetzungen für die Vergabe von Kreditpunkten insgesamt bestandene Leistung aus • Hausarbeit und Referat (80%) • Semesterbegleitleistung (20%)
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Andreas Harrer Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Dix, A., Finlay, J., Abowd, G., Beale, R. (2004): Human-Computer Interaction. Pearson Education Limited. DIN EN ISO 9241 Wilde, T., Hess, T. (2007): Forschungsmethoden der Wirtschaftsinformatik. Wirtsch. Inform. 49, 280–287. Aktuelle Forschungspapiere zum Thema Usability und User Experience Standardisierte Fragebogeninstrumente zum Thema Usability und User Experience

Nummer		Bezeichnung						
46002		Zeitreihenanalyse						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Zeitreihenanalyse			Vorlesung	60			2
-	Zeitreihenanalyse			Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme an den Modulveranstaltungen und dem Abschließen der Projektarbeit sind die Studierenden in der Lage: • Methoden zur Modellierung, Analyse und Vorhersage von Zeitreihen umfassend zu verstehen und anzuwenden • Klassische und moderne Ansätze der Zeitreihenanalyse auf unterschiedliche Anwendungsbereiche auszuwählen und zu implementieren. • Zeitreihendaten mit fortgeschrittenen Programmierertools zu analysieren, zu modellieren und zu visualisieren • Unsicherheiten und Vorhersagefehler in Modellen zu bewerten und adäquate Maßnahmen zur Verbesserung vorzuschlagen • Projekte zur Zeitreihenanalyse selbstständig zu planen, durchzuführen, kritisch zu reflektieren, zu dokumentieren und zu präsentieren							
3	Inhalte Einführung in Zeitreihen • Stationarität, Trendanalysen • Autokorrelation und Saisonalität • Glättungs- und Filterverfahren • Umgang mit fehlenden und extremen Werten Klassische Methoden der Zeitreihenanalyse • Lineare Modelle: AR, MA, ARMA, ARIMA • Erweiterte Modelle: SARIMA, GARCH • Modellanpassung: Modellidentifikation, Schätzung der Modellparameter, Validierungsmethoden und Leistungsmetriken • Modellvorhersage und Modelldiagnose • State-Space-Modelle (Zustandsraummodelle) und Kalman-Filter Weitere Methoden der Zeitreihenanalyse • Maschinelles Lernen für Zeitreihen (z. B. LSTMs, Prophet, XGBoost) • Regressionsmodelle für Zeitreihen Softwarepakete in R bzw. Python Praktische Anwendungen • Zeitreihenanalyse in Wirtschaft, Klima- und Wettervorhersagen, Finanz- und Gesundheitsdaten • Praxisprojekt: Analyse realer Zeitreihendaten							
4	Lehrformen • seminaristischer Unterricht mit Flipchart, Smartboard oder Projektion • Projektarbeit begleitendes Praktikum • Bearbeitung von Projektarbeit in Teamarbeit • Projektarbeit mit abschließender Präsentation							

5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Seminaristische Ausarbeitung und Vorstellung eines Themas • Projektarbeit mit abschließender Präsentation
7	Voraussetzungen für die Vergabe von Kreditpunkten • erfolgreiche Projektarbeit • erfolgreiche mündliche Präsentation • erfolgreich durchgeführte Ausarbeitung und Präsentation eines Themas
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Nadja Bauer Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Schlittgen, R., Sattarhoff, C (2020): Angewandte Zeitreihenanalyse mit R, 4., erweiterte und aktualisierte Auflage. Walter de Gruyter GmbH, Berlin/Boston. • Vogel, J. (2015): Prognose von Zeitreihen. Eine Einführung für Wirtschaftswissenschaftler. Springer Gabler, Wiesbaden • Box, G.E.P., Jenkins, G.M. & Reinsel (1994). Time Series Analysis: Forecasting and Control. Revised Edition. Holden Day, San Francisco. • Brockwell, J.P. & Davis, R.A. (1996). Introduction to Time Series and Forecasting. Springer, New York

Nummer		Bezeichnung					
46858		Projektmanagement					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Projektmanagement		seminaristische Veranstaltung	35			4
2 Lernergebnisse (learning outcomes) / Kompetenzen							
Nach erfolgreicher Teilnahme an diesem Modul sind die Studierenden in der Lage							
<u>Wissen und Verstehen</u>							
• Zentrale Konzepte des Projektmanagements zu benennen und zu erläutern. • Aktuelle Standards im Projektmanagement zu beschreiben. • Die Aufgaben und das Berufsbild eines IT-Projektmanagers wiederzugeben. • Konzepte des Qualitäts- und Risikomanagements zu erklären.							
<u>Einsatz, Anwendung und Erzeugung von Wissen</u>							
• Ein Planungsprojekt eigenständig durchzuführen und dabei vertiefte Methoden der Projektplanung einzusetzen. • Kenntnisse in den Methoden der Projektsteuerung (insbesondere Zeit- und Kostenmanagement) anzuwenden. • Für ein komplexes Projekt geeignete Methoden und Werkzeuge des Projektmanagements auszuwählen und diese im Team anzuwenden. • Methoden des Projektmanagements auf andere Aufgabenbereiche eines Wirtschaftsinformatikers zu übertragen.							
<u>Kommunikation und Kooperation</u>							
• Spezielle Methoden und Werkzeuge zur Unterstützung der Kooperation und Kommunikation im Projekt (z.B. Mind Mapping, CSCW-Tools, Entscheidungstabellen) zu nutzen und diese Werkzeuge sinnvoll miteinander zu verknüpfen. • Im Team zusammenzuarbeiten, um die Kenntnisse aus allen Phasen der Veranstaltung in einem komplexen Projekt anzuwenden.							
<u>Wissenschaftliches Selbstverständnis / Professionalität</u>							
• Die Übertragbarkeit von Projektmanagement-Methoden auf andere Aufgaben eines Wirtschaftsinformatikers zu erkennen. • Das eigene Handeln in Bezug auf die Aufgaben und das Berufsbild des IT-Projektmanagers einzuordnen.							
3 Inhalte							
• Grundkonzepte des Projektmanagements • Methoden und Werkzeuge der Projektplanung • Methoden und Werkzeuge der Projektsteuerung (Zeitmanagement, Kostenmanagement) • Methoden und Werkzeuge für ein Qualitätsmanagement in Projekten (Normen, Qualitätssysteme) • Methoden und Werkzeuge für das Risikomanagement in Projekten (Risikoabschätzung, Risikoüberwachung und handhabung)							

	Methoden und Werkzeuge für die Unterstützung von Kommunikation und Kooperation in Projektgruppen
4	Lehrformen - Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion - Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit - eigenständige Durchführung eines Projektes
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen - schriftliche Klausurarbeit - semesterbegleitende Prüfungsleistungen in Form von Übernahme einer Teilüprojektleitung
7	Voraussetzungen für die Vergabe von Kreditpunkten - bestandene Klausurarbeit - erfolgreiche Bearbeitung des Teilprojektes
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Robert Rettinger Prof. Dr. Dino Schönberg Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur Tiemeyer, Ernst (Hrsg.): <i>Handbuch IT-Projektmanagement. Vorgehensmodelle, Managementinstrumente, Good Practices.</i> 4., überarbeitete und erweiterte Auflage. Hanser, 2023. Wieczorrek, Hans W. / Mertens, Peter: <i>Management von IT-Projekten: Von der Planung zur Realisierung.</i> 5., überarbeitete Auflage. Springer Vieweg, 2021. Timinger, Holger: <i>Mit traditionellem, agilem und hybridem Vorgehen zum Erfolg.</i> Wiley-VCH, 2024

Nummer		Bezeichnung					
47723		Personalführung					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Personalführung		seminaristische Veranstaltung	35			4
-	Personalführung		Praktikum	35			2
2 Lernergebnisse (learning outcomes) / Kompetenzen							
<u>Fach- und Methodenkompetenz:</u>							
- Die Studierenden können die spezifischen Aufgaben von Führungskräften erläutern und gegen Fachaufgaben abgrenzen. - Die Studierenden kennen ausgewählte psychologische Grundlagen der Führung sowie ausgewählte Führungstheorien. - Die Studierenden kennen ausgewählte Führungsmethoden und könne diese im Rahmen von Fallbeispielen und Rollenspielen anwenden. - Die Studierenden können Fallbeschreibungen zu typischen Führungssituationen analysieren und Lösungsvorschläge auf Basis der gelernten Theorie entwickeln und argumentieren.							
<u>Fachübergreifende Methodenkompetenz:</u>							
Die Kenntnisse der psychologischen Grundlagen, die Fähigkeit (Konflikt-)Situationen analysieren zu können sowie die kommunikativen Fertigkeiten können die Studierenden in jedweder beruflichen Situation sinnvoll einsetzen.							
<u>Sozialkompetenz:</u>							
- Gruppenarbeiten fördern die Fähigkeit, mit anderen (fremden) Studierenden Lösungen zu erarbeiten. - Rollenspiele stärken die Fähigkeiten im konstruktiven Umgang mit Feedback und trainieren die Beobachtungsgabe für kommunikative (Konflikt-)Situationen.							
<u>Berufsfeldorientierung:</u>							
Durch Gastbeiträge von PersonalleiterInnen sowie von Führungskräften aus der Praxis erfahren die Studierenden, welche Anforderungen an Führungskräfte in Berufsfeldern der Informatik gestellt werden.							
3 Inhalte							
- Führungsrollen - Führungsaufgaben - Delegation und Zielvereinbarung - Motivation - Führungsstile - Teamstrukturen - Persönlichkeitseigenschaften - Gesprächsführung (Laterale) Führung in Projekten - Changemanagement - Führung im Wandel							

4	Lehrformen • seminaristischer Unterricht mit Flipchart, Smartboard oder Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit • Gruppenarbeit • Einzelarbeit • Fallstudien • Rollenspiele • Übungen oder Projekte auf der Basis von praxisnahen Beispielen
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen • Semesterbegleitende Prüfungsleistungen (Umfang 1/3) + mündliche Prüfung (Umfang 2/3)
7	Voraussetzungen für die Vergabe von Kreditpunkten Semesterbegleitende Prüfungsleistungen und mündliche Prüfung müssen in Summe bestanden sein.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Sinje Teschler-Nunkesser Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • BLESSIN, B. & WICK, A. 2014. Führen und Führen lassen, Konstanz und München, UVK Verlagsgesellschaft mbH. • FREY, D. & SCHMALZRIED, L. 2013. Philosophie der Führung, Gute Führung lernen von Kant, Aristoteles, Popper & Co, Berlin, Heidelberg, Springer-Verlag. • GERRIG, R. J. 2015. Psychologie, Halbergmoos, Pearson. • GROTE, S. & GOYK, R. (eds.) 2018. Führungsinstrumente aus dem Silicon Valley Konzepte und Kompetenzen: Springer Gabler. • NERDINGER, F. W., BLICKLE, G. & SCHAPER, N. 2014. Arbeits- und Organisationspsychologie, Berlin, Heidelberg, Springer-Verlag. • PASCHEN, M. 2014. Psychologie der Menschenführung, Berlin, Heidelberg, Springer-Verlag. • VON ROSENSTIEL, L., REGNET, E. & DOMSCH, M. E. (eds.) 2014. Führung von Mitarbeitern - Handbuch für erfolgreiches Personalmanagement, Stuttgart: Schäffer-Poeschel Verlag. • STÖWE, C. & KEROMOSEMITO, L. 2013. Führen ohne Hierarchie - Laterale Führung, Wiesbaden, Springer.

Nummer		Bezeichnung					
46877		Organisatorisch/rechtliche Aspekte von IT-Beschaffungen					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h			Selbst- studium 120 h		
-	Organisatorisch/rechtliche Aspekte von IT-Beschaffung		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage: • Prozesse und Aktivitäten von IT-Beschaffungsprozessen anzuwenden, • die zentralen Verfahren, rechtlichen Rahmenbedingungen und relevanten Ausschreibungsrichtlinien für IT-Beschaffungsprojekte anzuwenden, • geeignete Methoden, Techniken, Sprachen und Werkzeuge u.a. des Projekt- und Requirements Managements anzuwenden, um IT-Beschaffungsprozesse adäquat durchführen zu können • relevante Ergebnisse eines IT-Beschaffungsprozesses in studentischen Projektgruppen zu erschaffen und zu präsentieren						
3	Inhalte Im Rahmen der Lehrveranstaltung werden geeignete Methoden, Sprachen, Techniken, Regelungen und Werkzeuge zur Durchführung von IT-Beschaffungsprozessen vorgestellt. Im praktischen Teil der Lehrveranstaltung wird ein solcher IT-Beschaffungsprozess simuliert und von einer studentischen Projektgruppe exemplarisch durchgeführt; oftmals in Kooperation mit regionalen IT-Unternehmen. Hierbei werden relevante Aktivitäten von der Anforderungserhebung, über die Marktrecherche und Evaluierung geeigneter Systeme, die Analyse der rechtlichen Rahmenbedingungen bis hin zur Erstellung von Ausschreibungsunterlagen durchgeführt. Vor diesem Hintergrund werden im Rahmen der Lehrveranstaltung folgende Themenbereiche und Inhalte vermittelt: • Projektmanagement • Projektplanung mit Vorgangsknotennetzplänen und Gantt-Diagrammen, Kosten- und Aufwands-Controlling • Anforderungserhebung und -bestimmung • Erhebungsmethoden wie schriftliche Befragung und semi-strukturiertes Interview mit Interview-Leitfaden • Praktische Durchführung durch das Projektteam (zumeist in Kooperation mit regionalen IT-Unternehmen) • Anforderungsanalyse, -spezifikation und -dokumentation • Aufbau und Erstellung von Anforderungsdokumenten und Pflichtenheften • Gliederungen und IEEE-Standards • Rechtliche Rahmenbedingungen eines IT-Beschaffungsprojekts • Rechte und Pflichten von Auftraggeber/Auftragnehmer • Aufbau und Erstellung von Ausschreibungsunterlagen: Formulare, Regelungen, Gesetze • EVB-IT, BVB-Rahmenwerke • Ausschreibungsrecht, Vergaberecht, Ausschreibungsbewertung • Öffentliche, beschränkte und freihändige Vergabe • Durchführung von Bietergesprächen und -präsentationen: Ablauf und Vorgehen						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Seminaristischer Unterricht mit Flipchart, Smartboard oder Projektion • Vorstellung und Präsentation dedizierter Lehrinhalte durch die Studierenden						

	• Präsentationen der Studierenden
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Die Modulprüfung besteht aus zwei Teilen, • für die jeweils 50 Punkte erreicht werden können und • die jeweils mit 25 Punkten bestanden werden müssen: Teil 1: während des Semesters übernehmen Sie als Mitglied einer studentischen Projektgruppe (6-8 Teilnehmer:innen) die relevanten Aktivitäten im Rahmen eines simulierten IT-Beschaffungsprozesses. Hierbei erstellen Sie in Teilgruppen definierte Ergebnisse, die Sie als Dokument über eine elektronische Plattform bereitstellen und anschließend im Rahmen der seminaristischen Lehrveranstaltung präsentieren. Für die semesterbegleitenden Prüfungsleistungen können Sie maximal 50 Punkte erreichen. Teil 2: in der vorlesungsfreien Zeit findet eine 60-minütige Klausur zu ausgewählten Themen und Inhalten der Lehrveranstaltung statt. Bei dieser Klausur sind ebenfalls maximal 50 Punkte erreichbar. Die Gesamtnote setzt sich aus der Addition der in Teil 1 und Teil 2 erreichten Punkte zusammen: • ab 50 Gesamtpunkten erreicht man die Note 4,0 (ausreichend) und • ab 95 Gesamtpunkten wird die Note 1,0 (sehr gut) vergeben
7	Voraussetzungen für die Vergabe von Kreditpunkten Sie müssen die beiden Teile der Prüfungsform (siehe Pkt. 6 Prüfungsformen) mit mindestens ausreichend bestanden haben, um die Leistungsüberprüfung des Lehrmoduls zu bestehen und die dafür vorgesehenen Kreditpunkte erreichen zu können.
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr.-Ing. Guy Vollmer Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Balzert, H. (2008): Lehrbuch der Softwaretechnik - Softwaremanagement, Heidelberg: Spektrum Akademischer Verlag. • Balzert, H. (2009): Lehrbuch der Softwaretechnik - Basiskonzepte und Requirements Engineering, 3. Auflage, Heidelberg: Spektrum Akademischer Verlag. • Keller-Stoltenhoff, Leitzen, Ley (2017): Handbuch für die IT-Beschaffung (Band 1 und 2), Heidelberg: Rehm-Verlag. • Mangold, P. (2009): IT-Projektmanagement kompakt, 3. erweiterte Auflage, Heidelberg: Spektrum Akademischer Verlag. • Spitzczok, N.; Vollmer, G., Weber-Schäfer, U. (2014): Pragmatisches IT-Projektmanagement, 2. überarbeitete Auflage, Heidelberg: dpunkt-Verlag. • Vollmer, G. (2024): Vorlesungsunterlagen zur seminaristischen Lehrveranstaltung "Organisatorische und rechtliche Aspekte der IT-Beschaffung" • Winkelhofer, G. (2005): Management- und Projekt-Methoden, 3. vollst. überarbeitete Auflage, Berlin, Heidelberg: Springer Verlag.

Nummer		Bezeichnung					
41527		Analoges und Digitales Wahrnehmen					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
-	Analoges und Digitales wahrnehmen		seminaristische Veranstaltung	35	60	120	4
2	Lernergebnisse (learning outcomes) / Kompetenzen Im Architekturstudium wird Wahrnehmung von Architektur als eigenständiges Fach begriffen, da die Wahrnehmung im Sinne des Sehens und Denkens die Grundvoraussetzung für den Entwurf und das Gestalten ist. Mit der gleichen Motivation wird in diesem Modul die Wahrnehmung für das Analoge und Digitale im Sinne der Architektur vermittelt werden. Wichtig ist hierbei insbesondere das Spannungsfeld zwischen Analogem und Digitalem, da digitale Lösungen stets in einen analogen Kontext eingebunden sind und mit diesem interagieren. Oft ist es sogar so, dass ein Großteil der Digitalität für den Endnutzer verborgen ist (z.B. sind in einer Smart City oder einem Smart Home die technischen Prozesse zur Überwachung und Steuerung des Gebäudes vollkommen unsichtbar). Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage ... <u>Wissen und Verstehen</u> • Design als eine ästhetische Form der praktischen Welterschließung zu erläutern und in Bezug zur Gestaltung von Digitalisierung zu setzen. • die Bedeutung von Wahrnehmung für die Gestaltung analoger und digitaler Produkte bzw. Lösungen zu erklären. • Qualitätsbegriffe verschiedener Disziplinen zu unterscheiden und gegenüberzustellen. • die Bedeutung von Vorbildung und Vorwissen für die Wahrnehmung von Sachverhalten und Qualitäten zu erläutern. <u>Einsatz, Anwendung und Erzeugung von Wissen</u> • eine gegebene digitale Lösung auf Basis etablierter Qualitätsmodelle für digitale Systeme systematisch zu analysieren. • die Qualitäten alternativer Entwürfe einer digitalen Lösung anhand definierter Kriterien zu bewerten. <u>Kommunikation und Kooperation</u> • ihre Einschätzung zur Qualität einer gegebenen digitalen Lösung zielgruppengerecht zu erläutern und zu begründen. • eigene Kenntnisse und Positionen in einen interdisziplinären Austausch einzubringen und unterschiedliche fachliche Perspektiven zu berücksichtigen. <u>Wissenschaftliches Selbstverständnis / Professionalität</u> • das eigene Werte- und Qualitätsbewusstsein kritisch zu reflektieren und weiterzuentwickeln. • die im Modul erworbenen Kompetenzen als transversale Schlüsselqualifikationen für ihr berufliches Handeln einzuordnen.						
3	Inhalte • Einführung in den Leitgedanken von D. Feige „Design ist eine ästhetische Form der praktischen Welterschließung“						

	• Ästhetik für analoge und digitale Produkte (bspw. Form- und Farbgebung, Typographie, Markenwahrnehmung, Barrierefreiheit, Architektur als Lebensraum) • Wahrnehmung des Digitalen als soziotechnisches System in Abgrenzung zu menschlichen und technischen Systemen • Infosphäre als Modell des Digitalen (Infosphäre = Raum von Information mit Agenten und Operationen) • Qualitäten digitaler Lösungen (Customer Experience, hedonische vs. pragmatische Qualität) und Qualitäten digitaler System (ISO9241, ISO25000)
4	Lehrformen seminaristischer Unterricht, Gruppenarbeiten, Peer-Teaching, Projektarbeit
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Modulprüfung 70% - Hausarbeit (20 – 30 Seiten) und 30% Referat mit mündlicher Prüfung (15 – 20 Minuten)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Modulprüfung und Teilnahme am Ausstellungsformat
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Dr. Kim Lauenroth Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • A. Abel, B. Rudolf: Architektur wahrnehmen. transcript, 2018. • J. Albers: Interaction of Color, Yale University Press, 1963. • C. Alexander: The Timeless Way of Building. (3 vols). New York: Oxford University Press, 1979. • C. Alexander et al.: A Pattern Language: Towns, Buildings, Construction. (3 vols). New York: Oxford University Press, 1977. • ISO 9241-210: Menschzentrierte Gestaltung interaktiver Systeme. In: DIN EN ISO 9241: Ergonomie der Mensch-System-Interaktion, 2020. • ISO 25000: Software-Engineering – Qualitätskriterien und Bewertung von Softwareprodukten (SQuaRE) – Leitfaden für SQuaRE, 2014. • D. Feige: Design - eine philosophische Analyse. Suhrkamp, 2019. • L. Floridi: Philosophy and Computing: An Introduction. Taylor & Francis, 1999. • W. Lidwell et al.: Universal Principles of Design. Rockport Publishers, 2003. • J. Maeda: How to Speak Machine: Computational Thinking for the Rest of Us. Portfolio, 2019. • D. Pye: Nature and Aesthetics of Design. Bloomsbury Publishing, 2008. • D. Rams: Weniger aber besser. Die Gestalten Verlag, 2014. • A. White: The Elements of Graphic Design: Space, Unity, Page Architecture, and Type. 2nd Edition. Allworth Press, 2011. • P. Zumthor: Architektur denken. Birkhäuser, 2005.

Nummer		Bezeichnung					
46003		Ausgewählte Aspekte der Informatik 1					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h			Selbst- studium 120 h		
-	Ausgewählte Aspekte der Informatik 1		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>In diesem Modul werden nicht regelmäßig angebotene Veranstaltungen verschiedener Themen der Eingebetteten Systeme zusammengefasst. Die Inhalte und Kompetenzen werden jedes Semester in einem Zusatzdokument veröffentlicht.</u> <u>Die Kompetenzen ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
3	Inhalte <u>Die Inhalte ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen • Schriftliche Klausurarbeit • projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung • mündliche Prüfungen • Hausarbeiten • Referate						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Prüfung						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund						

10 Literatur

siehe Zusatzdokument zur konkreten Veranstaltung

Nummer		Bezeichnung					
46004		Ausgewählte Aspekte der Informatik 2					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Ausgewählte Aspekte der Informatik 2		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>In diesem Modul werden nicht regelmäßig angebotene Veranstaltungen verschiedener Themen der Informatik zusammengefasst. Die Inhalte und Kompetenzen werden jedes Semester in einem Zusatzdokument veröffentlicht.</u> <u>Die Kompetenzen ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
3	Inhalte <u>Die Inhalte ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen • Schriftliche Klausurarbeit • projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung • mündliche Prüfungen • Hausarbeiten • Referate						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Prüfung						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund						

10 Literatur

siehe Zusatzdokument zur konkreten Veranstaltung

Nummer		Bezeichnung					
46005		Ausgewählte Aspekte der Informatik 3					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h			Selbst- studium 120 h		
-	Ausgewählte Aspekte der Informatik 3		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>In diesem Modul werden nicht regelmäßig angebotene Veranstaltungen verschiedener Themen der Informatik zusammengefasst. Die Inhalte und Kompetenzen werden jedes Semester in einem Zusatzdokument veröffentlicht.</u> <u>Die Kompetenzen ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
3	Inhalte <u>Die Inhalte ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen • Schriftliche Klausurarbeit • projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung • mündliche Prüfungen • Hausarbeiten • Referate						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Prüfung						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund						

10 Literatur

siehe Zusatzdokument zur konkreten Veranstaltung

Nummer		Bezeichnung					
46006		Ausgewählte Aspekte der Informatik 4					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
		Kontakt- zeit 60 h			Selbst- studium 120 h		
-	Ausgewählte Aspekte der Informatik 4		seminaristische Veranstaltung	35			4
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>In diesem Modul werden nicht regelmäßig angebotene Veranstaltungen verschiedener Themen der Informatik zusammengefasst. Die Inhalte und Kompetenzen werden jedes Semester in einem Zusatzdokument veröffentlicht.</u> <u>Die Kompetenzen ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
3	Inhalte <u>Die Inhalte ergeben sich aus dem veröffentlichten Zusatzdokument zu der konkreten Veranstaltung.</u>						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen • Schriftliche Klausurarbeit • projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung • mündliche Prüfungen • Hausarbeiten • Referate						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Prüfung						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund						

10 Literatur

siehe Zusatzdokument zur konkreten Veranstaltung

Nummer		Bezeichnung					
46857		Ausgewählte Aspekte der Informationssicherheit					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
englisch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 90 h	
-	Ausgewählte Aspekte der Informationssicherheit / Selected Aspects of Information Security		Vorlesung	60			4
-	Ausgewählte Aspekte der Informationssicherheit / Selected Aspects of Information Security		Praktikum	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen Die Studierenden sind in der Lage, - relevante Begriffe zu definieren, zu differenzieren und zu erklären. - die entscheidende Bedeutung der Standardisierung in der Informationssicherheit zu verstehen und methodisch umzusetzen. - praxisrelevante Methoden, Best Practices und Softwaretools anzuwenden. - Projektaufgaben selbstständig umzusetzen und Ergebnisse zu dokumentieren.						
3	Inhalte - Abhängig von den für das jeweilige Semester tatsächlich ausgewählten Themen. - Beispielhafte Themen: - Informationssicherheitsmanagementsysteme: Grundlagen, ISO/IEC 27000 Reihe, Bedrohungsmodellierung, Risikomanagement - Betriebssystemsicherheit: Capabilities, AppArmor, SELinux, Linux Härtung - Netzwerksicherheit: Firewall Systeme, Intrusion Detection/Prevention Systeme (IDS/IPS) - Software-Sicherheit: Penetration Testing, Static Application Security Testing (SAST) - Hardware-Sicherheit: CPU Security, Trusted Platform Module (TPM), Smartcards - Weitere Themen: Privacy, Biometrische Systeme Die Unterrichtssprache ist <i>englisch</i> .						
4	Lehrformen - Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion - Einzelarbeit - vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen - Projektbezogene Arbeit (100%)						
7	Voraussetzungen für die Vergabe von Kreditpunkten - erfolgreiche Projektarbeit						
8	Stellenwert der Note für die Endnote						

9	Modulbeauftragte/r Prof. Dr. Holger Schmidt Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur - Abhängig von den für das jeweilige Semester tatsächlich ausgewählten Themen.

Nummer		Bezeichnung					
46007		Fortgeschrittene Digitale Forensik					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
					60 h	120 h	4
-	Fortgeschrittene Digitale Forensik		Vorlesung	60			2
-	Fortgeschrittene Digitale Forensik		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
<u>Fach- und Methodenkompetenz:</u>							
Nach Abschluss der Lehrveranstaltung „Fortgeschrittene Digitale Forensik“ sind die Studierenden in der Lage, forensische Analysen auf komplexen IT-Systemen durchzuführen und dabei sowohl aktuelle Methoden als auch spezialisierte Werkzeuge situationsgerecht einzusetzen. Sie können anspruchsvolle Datenquellen (z.B. Speicherabbilder, Netzwerkverkehr, Cloud-Umgebungen) systematisch untersuchen, um digitale Spuren zweifelsfrei zu identifizieren und zu sichern. Die Studierenden kennen die relevanten rechtlichen und technischen Rahmenbedingungen und sind in der Lage, forensische Prozesse zu planen, zu koordinieren und zu dokumentieren. Sie können geeignete Strategien zur Erhaltung der Beweiskette auswählen und anwenden sowie Vor- und Nachteile verschiedener Analysemethoden beurteilen.							
<u>Sozialkompetenz:</u>							
Die Kooperations- und Teamfähigkeit wird im Rahmen von praxisnahen Fallstudien und Projektarbeiten gefördert. Die Studierenden lernen, innerhalb von Projektteams transparent zu kommunizieren sowie Ergebnisse und Vorgehensweisen gemeinsam zu reflektieren. Sie können im Austausch mit unterschiedlichen Fachdisziplinen (z.B. Recht, IT-Sicherheit, Management) zielorientiert argumentieren und mit Feedback oder konstruktiver Kritik souverän umgehen. Dabei entwickeln sie die Fähigkeit, Konflikte oder Missverständnisse frühzeitig zu erkennen, aktiv zu entschärfen und die Ergebnisse im Team strukturiert zu präsentieren.							
3	Inhalte						
1. Erweiterte Analysetechniken und Tools							
• Arbeitsspeicherforensik							
• Mobile Device Forensics (Android, iOS)							
2. Netzwerk- und Cloud-Forensik							
• Log- und Datenanalyse in virtualisierten und Cloud-Umgebungen							
• Netzwerkflussanalysen (z.B. NetFlow, Zeek)							
• Protokoll- und Paketanalysen in verteilten Systemen							
3. Forensik in komplexen IT-Infrastrukturen							
• Container- und Microservices-Forensik (z.B. Docker)							
• IoT- und Embedded-Systeme (z.B. Smarthome-Forensik)							
• Besonderheiten bei Hybrid- und Multi-Cloud-Strategien							
4. Incident Response und Notfallmanagement							
• Aufbau und Rollenverteilung in Incident-Response-Teams							
• Einbindung in bestehende Sicherheitskonzepte (SIEM, SOC)							

	- Forensische Reaktion auf aktuelle Angriffsszenarien (z.B. Ransomware, APTs) 5. Rechtliche Grundlagen und Compliance - Datenschutzgesetze (DSGVO, BDSG) - Cybercrime-Gesetzgebung und strafprozessuale Aspekte - Gerichtsfeste Beweiserhebung und Dokumentation 6. Forensische Dokumentation und Kommunikation - Strukturierte Erstellung forensischer Berichte und Gutachten - Kommunikation mit juristischen Instanzen und Management - Präsentationstechniken für komplexe forensische Sachverhalte 7. Aktuelle Forschung und Trends - KI-gestützte forensische Analysen (z.B. Machine Learning für Anomalieerkennung) - Zukünftige Herausforderungen
4	Lehrformen - Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion - vorlesungsbegleitendes Praktikum
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen mündliche Prüfung
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. rer. nat. Daniel Spiekermann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur File system forensic analysis Brian Carrier 978-0321268174 The art of memory forensics Michael Hale Ligh, Andrew Case, Jamie Levy , Aaron Walters 978-1118825099

Nummer		Bezeichnung					
41521		Grundlagen des Bauens und Entwerfens digitaler Lösungen					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Wintersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
	-	Grundlagen des Bauens und Entwerfens digitaler Lösungen	Vorlesung	60			4
-	Grundlagen des Bauens und Entwerfens digitaler Lösungen	Übung	20			2	
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>Fach- und Methodenkompetenz:</u> Nach Abschluss der Lehrveranstaltung sind die Studierenden in der Lage, Systeme konzipieren und entwickeln zu können, die in ihrem zeitlichen Verhalten vorhersagbar sind. Sie kennen die technischen Parameter, die für die Auswahl von Planungsverfahren relevant sind und können anhand der Vor- und Nachteile ein geeignetes Verfahren situationsgerecht auswählen und implementieren. Die Studierenden können sich die besonderen zeitlichen Aspekte von synchronisierten Prozessen und von verteilten Systemen in Entwurf und Implementierung erschließen. <u>Sozialkompetenz:</u> Kooperations- und Teamfähigkeit wird während der Übungs- und Projektphasen trainiert. Die/der Studierende/r kann in Diskussionen zielorientiert argumentieren und mit Kritik sachlich umgehen; er/sie/es kann vorhandene Missverständnisse zwischen Gesprächspartnern erkennen und abbauen. Ergebnisse aus Gruppenarbeit können gemeinsam präsentiert werden.						
3	Inhalte • Der Zeitbegriff: mathematische Reduktion der Zeit, Aufbau und Eigenschaften einer Uhr, Zeitstandards • Einführung in die Planung: Prozessparameter, WCET, Brauchbarkeit, Prozesspräzedenz, Prozessanomalien • Prioritätsbasierte Planungsverfahren für aperiodische Prozesse: Earliest Due Date, Earliest Deadline First, Least Laxity First • Prioritätsbasierte Planungsverfahren für periodische Prozesse: Rate und Deadline Monotonic Scheduling, Einplanungstests (LL-Test, kritisches Intervall, RT-Test), Earliest Deadline First • Zeitbasierte Planungsverfahren: äußerer und innerer Zyklus, Anforderungen nach Baker&Shaw, Implementierung des Cyclic Executives • Planungsverfahren für synchronisierte Prozesse: Prioritätsumkehr, Verfahren (Nicht-präemptive kritische Abschnitte, Prioritätsvererbung, Prioritätsobergrenzen), Berechnung von Blockierungszeiten • Echtzeitbetriebssysteme: u.a. Architektur, Scheduler, Umgang mit Interrupts • Verteilte Systeme: Synchronisation von Uhren, Echtzeiteigenschaften verschiedener Medienzugriffsverfahren, aktuelle Echtzeit-Protokolle						
4	Lehrformen • Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • vorlesungsbegleitende Übung • vorlesungsbegleitendes Praktikum						
5	Teilnahmevoraussetzungen Keine						

6	Prüfungsformen • mündliche Prüfung • semesterbegleitende Prüfungsleistungen
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung
8	Stellenwert der Note für die Endnote
9	Modulbeauftragte/r Prof. Dr. Dirk Wiesmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund
10	Literatur • Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications, Springer, 2011 • Dieter Zöbel. Echtzeitsysteme Grundlagen der Planung, Springer, 2008. • Jane Liu. Real-Time Systems, Prentice Hall, 2000. • Peter Marwedel. Eingebettete Systeme, Springer, 2007. • Heinz Wörn und Dieter Brinkschulte. Echtzeitsysteme, Springer, 2005. • Burns, A., Wellings, A.; Real-Time Systems and Programming Languages; Pearson Education Ltd., Third Ed. 2001.

Nummer		Bezeichnung					
46833		IT-Netze					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
					60 h	120 h	
-	IT-Netze / IT Nets		Vorlesung	60			2
-	IT-Netze / IT Nets		Übung	20			1
-	IT-Netze / IT Nets		Praktikum	20			1
2	Lernergebnisse (learning outcomes) / Kompetenzen <u>Fach- und Methodenkompetenz:</u> Der/die Studierende versteht die Prinzipien, Protokolle und die Architektur von Rechnernetzen sowie der darauf basierenden Anwendungen. Er/sie wendet Verfahren des Netzdesigns auf Layer 2 und Layer 3 an, führt die Konfiguration von Netzkomponenten (Router, Switch) durch und plant die Einrichtung von virtuellen Netzen an. Er/sie versteht den Entwurf und die Implementierung von Kommunikationsprotokollen und ist in der Lage, verteilte Systeme mit physikalischen und virtuellen Netzwerkkomponenten zu konzipieren und zu konfigurieren. <u>Sozialkompetenz:</u> Anhand von praktischen Vorführungen sowie eigens erworbenen Erfahrungswerten durch praktische Übungen kann er/sie typische und anerkannte Technologien und Verfahren in den Bereichen Datennetzkommunikation und Einsatz virtueller Netzsysteme bewerten.						
3	Inhalte • Kommunikations- und Referenzmodelle; • theoretische Verfahren zur Kapazitätsplanung und Berechnung basierend auf statistischen Modellen und Markov-Ketten; • Netzalgorithmen für Switching - Spanning Tree Protokoll - und Routing - Open Shortest Path First • Weitverkehrslösungen, wie Multi Protocol Label Switching; • Virtualisierte Netzgeräte am Beispiel von CumulusVX und OPNSense, • Netzwerkmanagement auf Basis von SNMP und Einsatz von Zabbix als Überwachungssystem; • Referenzarchitekturen für Unternehmensnetze und Rechenzentrumsnetze, • Netzaspekte in Cloud Computing						
4	Lehrformen • Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion • Lösung von praxisnahen Übungsaufgaben in Einzel- oder Teamarbeit						
5	Teilnahmevoraussetzungen Keine						
6	Prüfungsformen schriftliche Klausurarbeit						
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene Klausurarbeit						
8	Stellenwert der Note für die Endnote						

9	Modulbeauftragte/r Prof. Dr. Stephan Recker Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur • Larry L. Peterson Bruce S. Davie: Computer Networks: a system approach, 2.ed., Morgan Kaufmann • Douglas Comer / David L. Stevens: Internetworking with TCP/IP, Vol.1 und 2, Prentice Hall

Nummer		Bezeichnung					
46851		Multimodale Interaktion in ambienten Umgebungen					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch		1 Sem.	1 / 2;3	Findet nur im Sommersemester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit 60 h	Selbst- studium 120 h	
-	Multimodale Interaktion in ambienten Umgebungen		Vorlesung	60			4
-	Multimodale Interaktion in ambienten Umgebungen		Übung	20			2
2	Lernergebnisse (learning outcomes) / Kompetenzen						
Wissen und Verstehen							
Die Studierenden:							
• verfügen über ein vertieftes, spezialisiertes Wissen im Bereich multimodaler Mensch-Maschine-Interaktion (MMI) und deren Einbettung in vernetzte intelligente Umgebungen. • kennen aktuelle Forschungsfragen, theoretische Konzepte und technologische Entwicklungen im Bereich moderner Interaktionsformen, einschließlich sensorbasierter, sprachbasierter und tangibler Interaktion. • sind in der Lage, interdisziplinäre Zusammenhänge zwischen Interaktionstechnologien, kognitiven Prozessen und systemischen Anforderungen zu analysieren und kritisch zu reflektieren. • haben ein wissenschaftlich fundiertes Verständnis für die technischen, methodischen und ethischen Herausforderungen bei der Gestaltung multimodaler Benutzungsschnittstellen.							
Einsatz, Anwendung und Erzeugung von Wissen							
Die Studierenden sind in der Lage:							
• neuartige Interaktionsformen systematisch zu analysieren, ihre Potenziale und Limitationen abzuleiten und diese auf spezifische Anwendungsfelder zu übertragen. • Konzepte, Methoden und Modelle für multimodale Benutzungsschnittstellen zu entwickeln und forschungsorientiert weiterzuentwickeln. • eigene wissenschaftliche Fragestellungen zu formulieren, Hypothesen zu entwickeln und empirische oder experimentelle Methoden zur Validierung anzuwenden. • innovative Lösungen für Mensch-Maschine-Interaktionssysteme zu entwerfen, zu implementieren und kritisch zu evaluieren. • komplexe interaktive Systeme im Kontext von Ambient Assisted Living (AAL) zu gestalten und interdisziplinäre Anforderungen methodisch zu integrieren. • technologische Entwicklungen aus der Forschung in konkrete Anwendungen zu übertragen und neue Interaktionsparadigmen zu erschließen.							
Kommunikation und Kooperation							
Die Studierenden:							
• sind in der Lage, ihre wissenschaftlichen Erkenntnisse und technischen Lösungen sowohl für Fachkollegen als auch für ein interdisziplinäres Publikum klar und überzeugend zu kommunizieren. • können in interdisziplinären Teams anspruchsvolle Problemstellungen systematisch bearbeiten und forschungsgeleitete Lösungsansätze erarbeiten. • verfügen über ausgeprägte Diskurs- und Argumentationsfähigkeiten zur kritischen Reflexion wissenschaftlicher Arbeiten und technologischer Entwicklungen. • sind in der Lage, Forschungs- und Entwicklungsarbeiten professionell zu dokumentieren, zu präsentieren und kritisch zu diskutieren.							

	- beherrschen Methoden der wissenschaftlichen Zusammenarbeit, einschließlich Peer-Review-Prozessen und kollaborativer Forschung Wissenschaftliches Selbstverständnis / Professionalität Die Studierenden: - haben die Fähigkeit, wissenschaftliche Theorien, Methoden und Konzepte kritisch zu hinterfragen und auf neue Problemstellungen im Bereich der Mensch-Maschine-Interaktion anzuwenden. - sind in der Lage, eigenständig Forschungsprojekte zu initiieren und innovative Ansätze zur Weiterentwicklung der MMI-Disziplin zu entwickeln. - reflektieren die gesellschaftlichen, ethischen und anwendungsbezogenen Auswirkungen von Interaktionstechnologien und berücksichtigen diese in ihren Lösungen. - verfügen über ein hohes Maß an Selbstständigkeit und Eigeninitiative im Wissenserwerb sowie in der Entwicklung neuer wissenschaftlicher und technologischer Konzepte. - verstehen sich als Teil der wissenschaftlichen Community und können ihre Expertise in Fachkonferenzen, Publikationen und interdisziplinären Forschungsprojekten einbringen.
3	Inhalte - Grundlagen der Interaktionsgestaltung aus Wahrnehmungs-, Arbeits- und Kognitionspsychologie; Theorien der Gestaltung: Distributed Cognition, Tätigkeitstheorie, Strukturierungstheorie; Interaktionsmodellierung - Beschreibung und Nutzung von Kontextinformationen für die Interaktionsdurchführung - Vertiefung in den folgenden technischen Bereichen: - Sensor-basierte Interaktionstechnologien, - Spracherkennung- und Steuerung, - Tangible Interaktion/Kamera-Projektor-Systeme; - Ambiente Umgebungen aus dem Bereich AAL, in den Aufgabenbereichen: - Sicherheit & Prävention (Hausnotruf, Beleuchtungssysteme,), - Gesundheit und Pflege (Vitalparameter Monitoring, Fitness-Tracker,), - Haushalt und Versorgung (Google Nest, Robotik, Service-Portale,), - Kommunikation und soziales Umfeld (Sprachsteuerung, Kommunikationslösungen,); - AAL-Plattformen und Internet of Things -Infrastrukturen als Grundlage für Multimodale Interaktion. - Vorgehen (Analyse, Konzeption, Methoden, Modelle) für die die Entwicklung multimodaler Benutzerschnittstellen. - Problemlösung am Beispiel einer selbst entwickelten multimodalen Benutzerschnittstelle aus dem Bereich AAL (studentische Projekte);
4	Lehrformen - Vorlesung in Interaktion mit den Studierenden, mit Tafelanschrieb und Projektion - Vorlesung im seminaristischen Stil, mit Tafelanschrieb und Projektion - vorlesungsbegleitende Projektarbeiten mit abschließender Präsentation
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen mündliche Prüfung (30 Minuten)
7	Voraussetzungen für die Vergabe von Kreditpunkten bestandene mündliche Prüfung (30 Minuten)
8	Stellenwert der Note für die Endnote

9	Modulbeauftragte/r Prof. Dr. Thomas Königsmann Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur • Rogers, I. (2012). HCI Theory: Classical, Modern, and Contemporary - Synthesis Lectures on Human-Centered Informatics. Morgan & Claypool. • Journal on Multimodal User Interfaces (2016), Volume 10, Springer International Publishing 2016 • BMBF/VDE Innovationspartnerschaft AAL (Hrsg.) 2011: Ambient Assisted Living (AAL) Komponenten, Projekte, Services Eine Bestandsaufnahme, VDE Verlag.

Nummer		Bezeichnung						
46995		Anerkannte Wahlpflichtprüfungsleistung						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls		CP
deutsch			1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach		6
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
						Kontakt- zeit	Selbst- studium	
2	Lernergebnisse (learning outcomes) / Kompetenzen							
3	Inhalte							
4	Lehrformen							
5	Teilnahmevoraussetzungen							
6	Prüfungsformen							
7	Voraussetzungen für die Vergabe von Kreditpunkten							
8	Stellenwert der Note für die Endnote							
9	Modulbeauftragte/r Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund							
10	Literatur							

Nummer		Bezeichnung					
46996		Anerkannte Wahlpflichtprüfungsleistung					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch			1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
2	Lernergebnisse (learning outcomes) / Kompetenzen						
3	Inhalte						
4	Lehrformen						
5	Teilnahmevoraussetzungen						
6	Prüfungsformen						
7	Voraussetzungen für die Vergabe von Kreditpunkten						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund						
10	Literatur						

Nummer		Bezeichnung					
46997		Anerkannte Wahlpflichtprüfungsleistung					
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP
deutsch			1 / 2;3	Findet in jedem Semester statt		Wahlpflichtfach	6
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
					Kontakt- zeit	Selbst- studium	
2	Lernergebnisse (learning outcomes) / Kompetenzen						
3	Inhalte						
4	Lehrformen						
5	Teilnahmevoraussetzungen						
6	Prüfungsformen						
7	Voraussetzungen für die Vergabe von Kreditpunkten						
8	Stellenwert der Note für die Endnote						
9	Modulbeauftragte/r Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im Studienportal der Fachhochschule Dortmund						
10	Literatur						

Nummer		Bezeichnung			
47581		Guided Research Project			
Sprache	Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls
deutsch	2 Sem.	1 / 2;3	Findet in jedem Semester statt		Pflichtfach
1	Veranstaltungen		Veranstaltungsart	geplante Gruppen- größe	Workload
					Kontakt- zeit
					Selbst- studium
					150 h
					570 h
-	Guided Research Project		Praktikum		
					10
					10
2	Lernergebnisse (learning outcomes) / Kompetenzen				
	<i>Wissen und Verstehen:</i> Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, - eine komplexe forschungs- und/oder praxisorientierte Problemstellung der Informatik zu verstehen und zu analysieren, und - wissenschaftliche und fachpraktische Methoden und Verfahren der Informatik zur Problemlösung zu differenzieren, zu selektieren und anzuwenden. <i>Einsatz, Anwendung und Erzeugung von Wissen:</i> Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, - geeignete theoretische oder konzeptuelle Ansätze zur Bearbeitung einer komplexen Aufgabenstellung auszuwählen, anzuwenden und zu bewerten, - Projektaufgaben zu planen, zu strukturieren und durch geeignete Methoden des Projektmanagements zu organisieren, - Ergebnisse schriftlich unter Berücksichtigung wissenschaftlicher Standards zu dokumentieren und kritisch zu reflektieren, und - Ergebnisse mündlich vor Fachpublikum zu präsentieren und zu diskutieren. <i>Kommunikation und Kooperation:</i> Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, - effektiv in einem Team zusammenzuarbeiten, um eine komplexe forschungs- oder praxisorientierte Aufgabenstellung zu bearbeiten, - Verantwortung für bestimmte Aufgabenbereiche zu übernehmen und aktiv zum Projekterfolg beizutragen, und - Konflikt- und Kommunikationsfähigkeiten zu stärken, indem Gruppenkonflikte konstruktiv gelöst werden. <i>Wissenschaftliches Selbstverständnis / Professionalität:</i> Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, - in teamorientierten Forschungs- und Entwicklungsprojekten zu arbeiten und sich an wissenschaftlichen sowie industriellen Standards zu orientieren, und - sich erfolgreich in größeren Projekten und interdisziplinären Arbeitsumfeldern einzubringen.				
3	Inhalte				
	Das Guided Research Project beinhaltet die Bearbeitung einer komplexen forschungs- oder praxisnahen Problemstellung der Informatik in Teamarbeit. Die Problemstellung wird von den Lehrenden gestellt und stammt in der Regel von laufenden Forschungsprojekten, Forschungseinrichtungen, Industrieunternehmen oder den Lehrgebieten der Lehrenden. Um eine fachliche Tiefe zu erreichen, die geeignet auf die Master-Thesis vorbereiten kann, und um die langfristige und nachhaltige Organisation und Abstimmung im Team sowie das verbindliche Übernehmen der Verantwortung für bestimmte Aktivitäten intensiv zu fördern, erstreckt sich das Guided Research Project über zwei Semester (Teil 1 + Teil 2).				

	Die projektspezifischen Fachinhalte und Grundlagen werden gemeinsam im Seminar zu Beginn des Projektes erarbeitet und durch die Bearbeitung des Projektes erweitert und vertieft. Darüber hinaus vermittelt das Guided Research Project folgende übergreifende Inhalte: • Einführung in das wissenschaftliche Arbeiten und projektorientierte Forschung • Analyse einer komplexen, forschungs- oder praxisnahen Problemstellung • Methoden und Werkzeuge zur strukturierten Bearbeitung wissenschaftlicher Projekte • Auswahl und Anwendung geeigneter theoretischer und praktischer Konzepte zur Lösung der Aufgabenstellung • Projektplanung, Meilensteinsetzung und Fortschrittskontrolle • Wissenschaftliche Dokumentation und Publikationsformate • Präsentation und Diskussion von Ergebnissen vor Fachpublikum • Teamarbeit, Projektmanagement und interdisziplinäre Zusammenarbeit • Reflexion der erarbeiteten Lösungen und Ableitung von Optimierungspotenzialen Das Guided Research Project trägt wesentlich zur Vorbereitung auf die Master-Thesis bei.
4	Lehrformen • Projektorientierte Bearbeitung einer komplexen Aufgabenstellung in Teamarbeit über zwei Semester • Seminar zur gemeinsamen Erarbeitung projektspezifischen Fachinhalte und Grundlagen bei Projektbeginn: Präsentation selbständig bearbeiteter Themen durch die Studierenden mit anschließender Diskussion • Kontinuierliches Coaching durch die betreuenden Lehrenden (z.B. durch regelmäßige Projekt-Meetings, Reviews) Darüber hinaus können weitere, im projektspezifischen Kontext erforderliche bzw. relevante, Lehrformen Anwendung finden.
5	Teilnahmevoraussetzungen Keine
6	Prüfungsformen Die Prüfung erfolgt in Form von zwei Teilprüfungen (Teil 1 und 2): Teil 1: • Referat (Umfang: 20%, Dauer: 20-30 Minuten) • Projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung (Umfang: 30%, Dauer mündl. Prüfung: 15-30 Minuten) Teil 2: • Projektbezogene Arbeit mit Dokumentation und Präsentation mit anschließender mündlicher Prüfung (Umfang: 50%, Dauer mündl. Prüfung: 15-30 Minuten)
7	Voraussetzungen für die Vergabe von Kreditpunkten Bestandene Modulprüfung, d.h. die Teilprüfungen Teil 1 und Teil 2 müssen jeweils mit der Note 4,0 oder besser bestanden sein
8	Stellenwert der Note für die Endnote

9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund
10	Literatur Projektspezifische Literatur wird von den Lehrenden gestellt und/oder von den Studierenden im Rahmen der Projektbearbeitung selbst recherchiert werden. Übergreifende Literatur: • Kirchner, Jens; Meyer, Sebastian (2022): Wissenschaftliche Arbeitstechniken für die MINT-Fächer, Springer Vieweg. ISBN: 978-3-6583-3911-1 • Carrera-Rivera, Angela; Ochoa, William; Larrinaga, Felix; Lada, Ganix (2022): How-to conduct a systematic literature review: A quick guide for computer science research, MethodsX, 9. DOI: doi.org/10.1016/j.mex.2022.101895 • Creswell, John W.; Creswell, J. David (2018): Research design: qualitative, quantitative, and mixed methods approaches, 5. Auflage, Los Angeles SAGE. ISBN: 978-1-5063-8676-8 • Shull, Forrest; Singer, Janice and Sjøberg, Dag I. K. (2008): Guide to Advanced Empirical Software Engineering. Springer, Berlin. ISBN: 978-1-8480-0043-8 • Dresch, Aline; Lacerda, Daniel Pacheco; Antunes, José Antônio Valle (2015): Design Science Research: Springer, Cham • Denning, Peter J. (2005): Is computer science science? Communications of the ACM 48, 4 (April 2005), 27–31. DOI: doi.org/10.1145/1053291.1053309

Nummer		Bezeichnung						
103		Thesis mit Kolloquium						
Sprache		Dauer	Studiensemester	Häufigkeit des Angebots		Art des Moduls	CP	
deutsch		1 Sem.	4	Findet in jedem Semester statt		Pflichtfach	30	
1	Veranstaltungen			Veranstaltungsart	geplante Gruppen- größe	Workload		SWS
			Kontakt- zeit			Selbst- studium		
2	Lernergebnisse (learning outcomes) / Kompetenzen							
3	Inhalte							
4	Lehrformen							
5	Teilnahmevoraussetzungen							
6	Prüfungsformen							
7	Voraussetzungen für die Vergabe von Kreditpunkten							
8	Stellenwert der Note für die Endnote							
9	Modulbeauftragte/r Prof. Dr. Sven Jörges Prof. Dr. Frank Künemund Lehrende/r siehe aktuelles Vorlesungsverzeichnis oder individuellen Studienplan im <u>Studienportal</u> der Fachhochschule Dortmund							
10	Literatur							