

**Universität Duisburg-Essen,  
Fakultät für Wirtschaftswissenschaften**

**Modulhandbuch für den  
Masterstudiengang für das Lehramt an Gymnasien  
und Gesamtschulen  
Unterrichtsfach Informatik  
(PO2014)**

(LA Info GyGe Master 2014)

für das Wintersemester 2020/2021



# Inhalt

Einführung .....	1
Hinweise .....	1
Module .....	1
Leistungspunkte .....	1
Studienaufwand .....	1
Prüfungsleistungen und -anforderungen .....	1
Bildung der Fachnote .....	1
Studienverlaufsplan .....	1
Hinweise zu Lehrveranstaltungen von Juniorprofessorinnen und Juniorprofessoren, außerplanmäßigen Professorinnen und Professoren, Honorarprofessorinnen und Honorarprofessoren, Privatdozentinnen und Privatdozenten, promovierten wissenschaftlichen Mitarbeiterinnen und Mitarbeitern sowie Lehrbeauftragten .....	2
Prüferinnen und Prüfer .....	2
Prüfungstermine und Anmeldefristen .....	2
<b>Wahlpflichtbereich Informatik - 1.-3. Fachsemester, Pflicht .....</b>	<b>3</b>
Modul: Berechenbarkeit und Komplexität (6 Credits) .....	4
Vorlesung: Berechenbarkeit und Komplexität (3 Credits) .....	5
Übung: Berechenbarkeit und Komplexität (3 Credits) .....	5
Modul: Concurrency (6 Credits) .....	6
Vorlesung: Concurrency (3 Credits) .....	7
Übung: Concurrency (3 Credits) .....	7
Modul: Design und Architektur von Softwaresystemen (6 Credits) .....	8
Vorlesung: Design und Architektur von Softwaresystemen (3 Credits) .....	8
Übung: Design und Architektur von Softwaresystemen (3 Credits) .....	9
Modul (geplante Umstrukturierung): Diskrete Simulation (6 Credits) .....	10
Vorlesung: Diskrete Simulation (3 Credits) .....	10
Übung: Diskrete Simulation (3 Credits) .....	11
Modul: Distributed Objects & XML (6 Credits) .....	12
Vorlesung: Distributed Objects & XML (3 Credits) .....	12
Übung: Distributed Objects & XML (3 Credits) .....	13
Modul: Formale Methoden des Software Engineering (6 Credits) .....	14
Vorlesung: Formale Methoden des Software Engineering (3 Credits) .....	14
Übung: Formale Methoden des Software Engineering (3 Credits) .....	15
Modul: Informations- und Softwarevisualisierung (6 Credits) .....	16
Vorlesung: Informations- und Softwarevisualisierung (3 Credits) .....	17
Übung: Informations- und Softwarevisualisierung (3 Credits) .....	18
Modul: Kommunikationsnetze 2 (6 Credits) .....	19
Vorlesung: Kommunikationsnetze 2 (3 Credits) .....	20
Übung: Kommunikationsnetze 2 (3 Credits) .....	20
Modul: Konzepte und Implementierung Objektorientierter Programmiersprachen (6 Credits) .....	21
Vorlesung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits) .....	21
Übung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits) .....	22
Modul: Mathematische Algorithmen der Informatik (6 Credits) .....	23
Vorlesung mit integrierter Übung: Mathematische Algorithmen der Informatik (6 Credits) .....	23
Modul: Mensch-Computer Interaktion (6 Credits) .....	24
Vorlesung mit integrierter Übung: Mensch-Computer Interaktion (6 Credits) .....	24
Modul: No-Frills Software Engineering (6 Credits) .....	25
Vorlesung mit praktischer Übung: No-Frills Software Engineering (6 Credits) .....	26
Modul: Paradigmen und Konzepte der Softwareentwicklung (6 Credits) .....	27
Vorlesung mit integrierter Übung: Paradigmen und Konzepte der Softwareentwicklung (6 Credits) .....	27
Modul: Programmieren in C/C++ (6 Credits) .....	28
Vorlesung: Programmieren in C/C++ (3 Credits) .....	28
Übung: Programmieren in C/C++ (3 Credits) .....	29
Modul: Requirements Engineering und Management 1 (6 Credits) .....	30
Vorlesung: Requirements Engineering und Management 1 (3 Credits) .....	31
Übung: Requirements Engineering und Management 1 (3 Credits) .....	32
Modul: Requirements Engineering und Management 2 (6 Credits) .....	33
Vorlesung: Requirements Engineering und Management 2 (3 Credits) .....	33
Übung: Requirements Engineering und Management 2 (3 Credits) .....	34
Modul: Secure Software Systems (6 Credits) .....	35
Vorlesung: Secure Software Systems (3 Credits) .....	35
Übung: Secure Software Systems (3 Credits) .....	36
Modul (auslaufend): Software-Engineering für mobile Systeme (6 Credits) .....	37
Vorlesung mit integrierter Übung: Software-Engineering für mobile Systeme (6 Credits) .....	38
Modul: Software-Qualitätssicherung (6 Credits) .....	39
Vorlesung: Software-Qualitätssicherung (3 Credits) .....	40
Übung: Software-Qualitätssicherung (3 Credits) .....	40
<b>Pflichtbereich Didaktik der Informatik - 1. Fachsemester, Pflicht .....</b>	<b>41</b>
Modul: Didaktik der Informatik II (6 Credits) .....	42
Vorlesung: Didaktik der Informatik II (3 Credits) .....	43
Übung: Didaktik der Informatik II (3 Credits) .....	44

<b>Wahlpflichtbereich Didaktik der Informatik - 2.-3. Fachsemester, Pflicht</b> .....	<b>45</b>
Modul: Aktuelle Beiträge zur Didaktik der Informatik (3 Credits) .....	46
Seminar: Aktuelle Beiträge zur Didaktik der Informatik (3 Credits) .....	46
Modul: Informatik in der Sekundarstufe I (3 Credits) .....	47
Seminar: Informatik in der Sekundarstufe I (3 Credits) .....	47
Modul: Informatiksysteme aus fachdidaktischer Sicht (3 Credits) .....	48
Seminar: Informatiksysteme aus fachdidaktischer Sicht (3 Credits) .....	48
Modul: Methodeneinsatz im Informatikunterricht (3 Credits) .....	49
Seminar: Methodeneinsatz im Informatikunterricht (3 Credits) .....	49
Modul: Schülerlabor Informatik (3 Credits) .....	50
Seminar: Schülerlabor Informatik (3 Credits) .....	50
Modul: Sprachbildung im Informatikunterricht (3 Credits) .....	51
Seminar: Sprachbildung im Informatikunterricht (3 Credits) .....	52
<b>Begleitmodul zum Praxissemester "Schule und Unterricht forschend verstehen" - 2. Fachsemester, Pflicht</b> .....	<b>53</b>
Modul: Praxissemester (5 Credits) .....	54
Seminar: Begleitung des Praxissemesters (5 Credits) .....	54
Modul: Praxissemester (2 Credits) .....	56
Seminar: Begleitung des Praxissemesters (2 Credits) .....	56
<b>Pflichtbereich Masterarbeit einschließlich Begleitmodul - 4. Fachsemester, Pflicht</b> .....	<b>57</b>
Modul: Professionelles Handeln wissenschaftsbasiert weiterentwickeln (3 Credits) .....	58
Seminar: Professionelles Handeln wissenschaftsbasiert weiterentwickeln aus der Perspektive der Informatik und der Didaktik der Informatik (3 Credits) .....	58
Modul: Masterarbeit (Master LA Info GyGe) (20 Credits) .....	59
Abschlussarbeit: Didaktik der Informatik (20 Credits) .....	59
Abschlussarbeit: Networked Embedded Systems (20 Credits) .....	59
Abschlussarbeit: Software Systems Engineering (20 Credits) .....	59
Abschlussarbeit: Software-Engineering, insb. mobile Anwendungen (20 Credits) .....	60
Abschlussarbeit: Spezifikation von Softwaresystemen (20 Credits) .....	60
Abschlussarbeit: Technik der Rechnernetze (20 Credits) .....	60
Abschlussarbeit: Mensch-Computer Interaktion (20 Credits) .....	60
Abschlussarbeit: Sichere Software Systeme (20 Credits) .....	60
Abschlussarbeit: Visualisierung (20 Credits) .....	60

## Einführung

## Hinweise

Dieses Modulhandbuch dient als kommentiertes Veranstaltungsverzeichnis für die Studierenden und gleichzeitig als Unterlage für die Akkreditierungsbehörde. Alle inhaltlichen und organisatorischen Angaben der Modulbeschreibungen beruhen auf Angaben der Dozenten. Beachten Sie, dass immer Änderungen möglich sind, und das Modulhandbuch daher jährlich überarbeitet wird.

## Module

Unter Modularisierung versteht man die Zusammenfassung von Stoffgebieten zu thematisch und zeitlich abgerundeten, in sich geschlossenen und mit sog. "Credits" versehenen abprüfbaren Einheiten. Module können verschiedene Lehr- und Lernformen umfassen und die Inhalte können sich auf ein einzelnes Semester oder auch auf ein ganzes eines Studienjahr verteilen. Wenn alle zu einem Modul gehörigen Prüfungsleistungen erbracht sind, werden dem Prüfungskonto sog. Credits (=Cr) gutgeschrieben und es wird die Note des Moduls berechnet.

## Leistungspunkte

Die Credits (manchmal auch Leistungspunkte oder Kreditpunkte genannt) werden nach dem Standard ECTS vergeben (European Credit Transfer System = Europäisches System zur Anrechnung von Studienleistungen). Das European Credit Transfer System dient der Erfassung der von den Studierenden erbrachten Leistungen sowie der Anerkennung von Prüfungsleistungen aus anderen Studiengängen. Pro Studienjahr sollen 60 Credits erworben werden. Auf der Grundlage von erworbenen Credits und der dabei erzielten Noten (Grade Points) werden die gewichteten Durchschnittsnoten (Grade Point Averages) der Module und die Noten der Masterprüfung insgesamt berechnet.

## Studienaufwand

Jede Lehrveranstaltung ist mit Credits versehen, die dem jeweils erforderlichen Studienaufwand (Workload) entsprechen. Ein Credit entspricht dabei einem Studienaufwand von 30 Stunden effektiver Studienzeit; dies umfasst Präsenzzeiten, Vor- und Nachbereitung sowie die Prüfungsvorbereitungen. Ein Studienjahr umfasst 60 Credits, was 1800 Arbeitsstunden pro Jahr entspricht. Der Umfang von Lehrveranstaltungen und die zugehörigen Credits der einzelnen Lehrveranstaltungen sind in den Modulbeschreibungen festgelegt. Bei dem erfolgreichen Abschluss eines Moduls werden die für dieses Modul vorgesehenen Credits dem Bonuspunktekonto des bzw. der Studierenden gutgeschrieben.

## Prüfungsleistungen und -anforderungen


Die zu erbringenden Prüfungsleistungen können den jeweiligen Modulbeschreibungen entnommen werden. Die Prüfungsdauer bzw. der Umfang schriftlicher Arbeiten orientieren sich an den Vorgaben der Prüfungsordnung für diesen Studiengang. Die konkreten Prüfungsanforderungen werden von den Dozentinnen und Dozenten spätestens zu Beginn der Lehrveranstaltung bekannt gegeben. Das gleiche gilt im Falle von Studienleistungen, insbesondere wenn sie Voraussetzung zur Teilnahme an der Prüfung bzw. für den Modulabschluss sind.

## Bildung der Fachnote

Der Stellenwert der einzelnen Modulnoten bei der Bildung der Fachnote ergibt sich aus §28 der Prüfungsordnung.

## Studienverlaufsplan

Studienbeginn: WS oder SS																															
3 Cr	4. FS			Masterarbeit												1 Cr															
		Begleitung MA-Arbeit*	Begleitung MA-Arbeit*													1 Cr															
12 Cr	3. FS	WP Info 2														1 Cr															
																WP DDI 3	WP DDI 2	1 Cr													
2 Cr	2. FS															1 Cr															
																WP DDI 1	5 bzw. 2 Cr.	1 Cr													
12 Cr	1. FS	WP Info 1														1 Cr															
																Didaktik der Informatik II		1 Cr													
53 Credits		Masterstudium																													
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> <td style="width: 50px; text-align: center;">1 Cr</td> </tr> </table>																1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr
1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr	1 Cr																
* Begleitveranstaltung im Fach der Master-Arbeit																															

ERKLÄRUNG:			
FARB- ZU- ORD- NUNG:	<b>Bereiche</b>		
	<b>Informatik</b>	<b>Didaktik</b>	<b>Praxisbegleitung</b>
	Die Farben entsprechen den Studien-Bereichen. Aus den verschiedenen Bereichen sind die Module zu wählen.		
BE- GRIFFE	<b>WP</b> = Wahlpflichtbereich <b>Info</b> = Informatik <b>DDI</b> = Didaktik der Informatik	<b>Cr = Credit</b> Punktesystem nach dem sich die Note bemisst; gibt außerdem Auskunft über den <i>Workload</i> .  1 Cr = 30 h Workload	<b>Workload</b> = Arbeitsaufwand in h; beinhaltet Lehrveranstaltungen, Vor- und Nachbereitung, Prüfungsvorbereitung und Prüfungen etc.   1 Einheit = 1 Credit
	Der Studienverlaufsplan ist erstellt gemäß Modulhandbuch; er ist eine Empfehlung und dient der Orientierung.		

## Hinweise zu Lehrveranstaltungen von Juniorprofessorinnen und Juniorprofessoren, außerplanmäßigen Professorinnen und Professoren, Honorarprofessorinnen und Honorarprofessoren, Privatdozentinnen und Privatdozenten, promovierten wissenschaftlichen Mitarbeiterinnen und Mitarbeitern sowie Lehrbeauftragten

Veranstaltungen und Prüfungen von Juniorprofessorinnen und Juniorprofessoren, außerplanmäßigen Professorinnen und Professoren, Honorarprofessorinnen und Honorarprofessoren, Privatdozentinnen und Privatdozenten, promovierten wissenschaftlichen Mitarbeiterinnen und Mitarbeitern sowie Lehrbeauftragten, mit Ausnahme von Veranstaltungen und Prüfungen des Pflichtbereichs, stellen ein freiwilliges Zusatzangebot der Fakultät für Wirtschaftswissenschaften im angegebenen Semester dar. Es besteht kein Rechtsanspruch der Studierenden auf wiederholte Durchführung der Veranstaltung und Prüfung im Folgesemester oder weiteren Semestern. Informieren Sie sich jeweils vor Vorlesungsbeginn über das aktuelle Angebot. Erstmalige Angebote an Lehrveranstaltungen stehen unter dem Vorbehalt der Genehmigung und/oder Finanzierung.

### Prüferinnen und Prüfer

An der Fakultät für Wirtschaftswissenschaften gilt der Grundsatz „wer lehrt, der prüft“. Prüferinnen und/oder Prüfer sind daher die in der jeweiligen Modulbeschreibung genannten Lehrperson/en. Bei Veranstaltungskombinationen aus Vorlesung und (i.d.R.) Übung ist die Lehrperson der Vorlesung die Prüferin oder der Prüfer. Bei mehreren Lehrpersonen, welche die Veranstaltung im semesterweisen Wechsel durchführen, ist die oder der im jeweiligen Semester Lehrende in den zugehörigen Prüfungen auch Prüferin oder Prüfer. Dies gilt unbeschadet der ergänzenden Bestellung von Prüferinnen und Prüfern durch den Prüfungsausschuss.

### Prüfungstermine und Anmeldefristen

Bitte informieren Sie sich rechtzeitig auf den Seiten des [Bereichs Prüfungswesen](#) über die Prüfungstermine und die Anmeldefristen, insb. auch bei Sonderprüfungen die außerhalb der regulären Prüfungszeiträume liegen.

## Wahlpflichtbereich Informatik - 1.-3. Fachsemester, Pflicht

Es sind zwei Module im Umfang von mind. 12 Credits zu belegen. Ein eventuell entstehender Überhang entfällt. Die folgenden Module können nur gewählt werden, wenn sie nicht im Bachelorstudium (nach PO 2011) absolviert wurden:

- Berechenbarkeit und Komplexität
- Concurrency
- Design und Architektur von Softwaresystemen
- Distributed Objects & XML
- Fehlertolerante verteilte Systeme
- Kommunikationsnetze 2
- Konzepte und Implementierung Objektorientierter Programmiersprachen
- Programmieren in C/C++
- Requirements Engineering und Management 1

<b>Modul: Berechenbarkeit und Komplexität (6 Credits)</b>	
Name im Diploma Supplement	Theoretical Computer Science
Verantwortlich	Prof. Dr. Barbara König
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• verfügen über die Kompetenz, Sachverhalte der theoretischen Informatik formal zu beschreiben und zu analysieren, insbesondere mit Bezug auf die Gebiete Berechenbarkeitstheorie und Komplexität</li> <li>• beherrschen Berechnungsmodelle wie Turing-Maschinen, LOOP-, WHILE-, GOTO-Programme, primitiv rekursive und mu-rekursive Funktionen</li> <li>• sind in der Lage, durch den Beweis der Äquivalenz dieser Berechnungsmodelle die Churchsche These nachzuvollziehen</li> <li>• verstehen Begriffe wie Unentscheidbarkeit und Reduzierbarkeit und können diese in einem Informatikkontext anwenden</li> <li>• kennen wichtige unentscheidbare Probleme (Halteproblem, Postsches Korrespondenzproblem, etc.)</li> <li>• besitzen die Fähigkeit, die Unentscheidbarkeit einer Problemstellung formal zu beweisen</li> <li>• kennen verschiedene Komplexitätsklassen sowie das P=NP-Problem und das Konzept der (NP-)Vollständigkeit</li> <li>• können die Komplexität von Problemen mit den bekannten Komplexitätsformeln abschätzen und sind in der Lage, Reduktionen formal durchzuführen</li> <li>• besitzen ein tieferes Verständnis für zentrale Konzepte der theoretischen Informatik</li> <li>• sind dadurch in der Lage, informatische Probleme mit formalen Methoden der theoretischen Informatik zu behandeln und zu lösen</li> </ul>
Praxisrelevanz	Dieses Modul vermittelt wesentliche Grundlagen, die für weite Bereich der praktischen Informatik relevant sind und ohne deren Kenntnis weder effektive noch effiziente Lösungen erstellt werden können.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 120 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Kernstudium &gt;Pflichtbereich II: Informatik &gt;3.-4. Fachsemester, Pflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Berechenbarkeit und Komplexität (3 Credits)</li> <li>• Übung: Berechenbarkeit und Komplexität (3 Credits)</li> </ul>
WIWI-M0043 Modul: Berechenbarkeit und Komplexität	



Vorlesung: Berechenbarkeit und Komplexität (3 Credits)			
Name im Diploma Supplement	Theoretical Computer Science		
Anbieter	Fachgebiet Theoretische Informatik <a href="http://www.ti.inf.uni-due.de/">http://www.ti.inf.uni-due.de/</a>		
Lehrperson	Prof. Dr. Barbara König		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Kenntnisse der Modellierungsmethoden der Informatik werden nachdrücklich empfohlen.			
<b>Abstract</b> Die Vorlesung gibt eine Einführung in die theoretische Informatik, insbesondere in die Gebiete Berechenbarkeit und Komplexität.			
<b>Lehrinhalte</b> Die Berechenbarkeits- und Komplexitätstheorie ist eine wichtige Grundlage der Informatik. Hierbei geht es um Fragestellungen der Form: was kann überhaupt berechnet werden? Wie teuer ist diese Berechnung? Mit dem P-NP-Problem erläutert dieses Gebiet auch das wichtigste bisher ungelöste Problem der theoretischen Informatik. Im Rahmen dieser Veranstaltung werden grundlegende Kenntnisse zu den Bereichen Berechenbarkeit und Komplexität vermittelt. Inhalte im Einzelnen: <ul style="list-style-type: none"> <li>• Berechenbarkeit (Turing-Maschinen, Intuitiver Berechenbarkeitsbegriff, Churchsche These, LOOP-, WHILE-, GOTO-Berechenbarkeit, Primitiv rekursive und mu-rekursive Funktionen, Ackermannfunktion, Halteproblem, Unentscheidbarkeit, Reduktionen, Postisches Korrespondenzproblem, Weitere unentscheidbare Probleme)</li> <li>• Komplexität (Komplexitätsklassen, P-NP-Problem, NP-Vollständigkeit, Weitere NP-vollständige Probleme, Randomisierung, Primzahltests).</li> </ul> Hinweis: <ul style="list-style-type: none"> <li>• Die Vorlesung wird wechselweise in Duisburg und Essen durchgeführt, jeweils mit Übertragung an den anderen Campus. Achten Sie auf die Modalitäten für die Essener Teilnehmer.</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• U. Schöning: "Theoretische Informatik - kurzgefasst", Spektrum, Akademischer Verlag (4. Auflage), 2000</li> <li>• Skript zur Vorlesung, siehe Homepage (<a href="http://www.informatik.uni-duisburg.de/AGThInf/">http://www.informatik.uni-duisburg.de/AGThInf/</a>)</li> </ul>			
<b>didaktisches Konzept</b> Vorlesung mit Folien und Erklärung komplexer Inhalte mit stiftbasierter Eingabe auf dem TabletPC; Videoübertragung an den anderen Campus; Bereitstellung von Vorlesungsvideos			
WIWI-C0006 <b>Vorlesung: Berechenbarkeit und Komplexität</b> im Modul WIWI-M0043: Berechenbarkeit und Komplexität			

Übung: Berechenbarkeit und Komplexität (3 Credits)			
Name im Diploma Supplement	Theoretical Computer Science		
Anbieter	Fachgebiet Theoretische Informatik <a href="http://www.ti.inf.uni-due.de/">http://www.ti.inf.uni-due.de/</a>		
Lehrperson	Prof. Dr. Barbara König		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> keines			
<b>Abstract</b> Übungen zu theoretischer Informatik, insbesondere zu den Gebieten Berechenbarkeit und Komplexität			
<b>Lehrinhalte</b> Es werden die Inhalte der Vorlesung durch Übungen vertieft.			
<b>Literaturangaben</b> Siehe Literaturangaben der Vorlesung.			
<b>didaktisches Konzept</b> Erarbeiten der Vorlesungsinhalte mit den Tutoren; Vorstellung der Lösung der Übungsaufgaben; Korrektur und Bewertung der von den Studierenden abgegebenen Lösungen			
WIWI-C0005 <b>Übung: Berechenbarkeit und Komplexität</b> im Modul WIWI-M0043: Berechenbarkeit und Komplexität			



<b>Modul: Concurrency (6 Credits)</b>	
Name im Diploma Supplement	Concurrency
Verantwortlich	Jun.-Prof. Dr. Lucas Davi
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• haben ein Verständnis für die Modellierung nebenläufiger Systeme</li> <li>• kennen unterschiedliche Spezifikationstechniken für zustandsbasierte Systeme (kommunizierende endliche Automaten und Ablaufmodelle)</li> <li>• können Modelle erstellen und hinsichtlich ihrer Korrektheitseigenschaften formal untersuchen</li> <li>• beherrschen Grundprinzip und Werkzeuge zum Model Checking für synchrone zeitbehafete Automatenetze</li> <li>• können die Modellierung auf Probleme des nebenläufigen Rechnens anwenden</li> <li>• kennen die Synchronisationskonzepte des nebenläufigen Rechnens</li> <li>• sind in der Lage, effiziente nebenläufige System zu konstruieren und die Vor- und Nachteile verschiedener Lösungsansätze vergleichend zu bewerten</li> <li>• sind in der Lage, das nicht deterministische Verhalten von nebenläufigen Systemen durch Ablaufmodelle darzustellen und formal zu spezifizieren</li> <li>• können Eigenschaften von Modellen formal beschreiben und diese durch modellgestützte Verfahren verifizieren oder falsifizieren</li> <li>• beherrschen den Umgang mit Werkzeugen zur Erstellung und Behandlung (Animation, Verifikation, Model Checking) von Modellen nebenläufiger Systeme können praktische Aufgabenstellungen hinsichtlich Nebenläufigkeit analysieren und einer Implementierung zugänglich machen</li> <li>• können nebenläufige Threads für die bekannten „klassischen“ Modelle programmieren und darüber hinaus auch Threads zur Implementierung beliebiger Synchronisationsalgorithmen einsetzen</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90-120 Minuten). Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme Prüfungsvorleistung oder aber Bestandteil der Prüfung ist. Ist letzteres der Fall, so bilden die Teilleistungen zusammen mit der Abschlussprüfung eine zusammengesetzte Prüfung mit einer Endnote. Bestandene Prüfungsvorleistungen/Teilleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Kernstudium &gt;Pflichtbereich II: Informatik &gt;2.-3. Fachsemester, Pflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Bachelor 2013&gt;Informatik &gt;Liste 1 &gt;1.-6. Fachsemester, Wahlpflicht</li> <li>• TechMathe Bachelor 2013&gt;Wahlpflichtbereich &gt;Profil "Software Systems Engineering" &gt;1.-6. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Concurrency (3 Credits)</li> <li>• Übung: Concurrency (3 Credits)</li> </ul>
WIWI-M0795 Modul: Concurrency	

Vorlesung: Concurrency (3 Credits)			
Name im Diploma Supplement	Concurrency		
Anbieter	Juniorprofessur für Sichere Software Systeme <a href="https://www.syssec.wiwi.uni-due.de/">https://www.syssec.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Lucas Davi		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Elementare Java-Programmierung			
<b>Lehrinhalte</b> In der Vorlesung werden Concurrency-Konzepte sowohl unter Modellierungs- wie auch unter Programmierungsgesichtspunkten behandelt. Es werden grundlegenden Konzepte der Nebenläufigkeit, die in der Hardware, im Betriebssystem, in Programmen, in verteilten Systemen und in Rechnernetzen eine wichtige Rolle spielen, in einheitlicher Weise behandelt. Themen sind: <ul style="list-style-type: none"> <li>• Einführung in die Modellierung mit Automaten und Transitionssystemen, Eigenschaften von Automaten, Erweiterungen von Automaten (Variabel, Spontanübergänge, Nicht-Determinismus)</li> <li>• Kommunizierende Endliche Automaten: synchrone und asynchrone Automatenkommunikation, Verhaltensspezifikation mit Automaten, Weg/Zeit-Diagramme, Prinzipie der Protokollspezifikation und der automatenbasierten Erreichbarkeitsanalyse, Algorithmen zur Erreichbarkeitsanalyse, erschöpfende und partielle Exploration;</li> <li>• Synchrone und zeitbehaftete Automaten, Einführung in Timed Computational Tree Logic und Model Checking mit UPPAAL</li> <li>• Beispiel aus den Bereichen Concurrent Execution, Mutual Execution, Deadlock, Safety and Liveness Properties.</li> <li>• Programmmodell, Ausführungsmodell, Ablaufmodell, Schreibkonflikt, Datenabhängigkeit, Verklemmung</li> <li>• Parallelisierung in der Hardware, Parallelisierung im Grundsystem, Parallelisierung auf Programmebene, Operatoren zur Synchronisation und Kommunikation</li> <li>• Implementierung nebenläufiger Systeme in Java, insbesondere Threads in Java, Interaktion zwischen Threads, Monitore, Wartebedingungen in Programmen, Vermeidung von Verklemmung, Grundmuster des nebenläufigen Rechnens für bestimmte Anwendungen, Echtzeitprozesse, Programmierstile für Echtzeitsysteme (zeitbasierte und ereignisorientierte Ansätze)</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• B. Müller-Clostermann: Folienskript zu "Concurrency - Teil I"; siehe Homepage (als pdf-Dokument erhältlich)</li> <li>• K. Echte: Folienskript zu "Concurrency - Teil II"; siehe Homepage (als pdf-Dokument erhältlich)</li> <li>• J. Magee, J. Kramer: Concurrency – State models and Java programs; Wiley, 1999</li> <li>• P. Ziesche: Nebenläufige &amp; Verteilte Programmierung, W3L-Verlag, 2005</li> <li>• A. Tanenbaum: Moderne Betriebssysteme; Hanser-Verlag</li> <li>• R. G. Hertwich, G. Hommel: Nebenläufige Programme; Springer-Verlag</li> </ul> Weiteres Begleitmaterial und Skripte, jeweils aktuelle Hinweise finden Sie auf der Homepage der Dozenten.			
WIWI-C1100 Vorlesung: Concurrency im Modul WIWI-M0795: Concurrency			

Übung: Concurrency (3 Credits)			
Name im Diploma Supplement	Concurrency		
Anbieter	Juniorprofessur für Sichere Software Systeme <a href="https://www.syssec.wiwi.uni-due.de/">https://www.syssec.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Lucas Davi		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> elementare Java-Programmierung			
<b>Lehrinhalte</b> <ul style="list-style-type: none"> <li>• Grundkonzepte der Nebenläufigkeit</li> <li>• Modellierung von Nebenläufigkeit mit gekoppelten Automaten</li> <li>• Synchrone und asynchrone Kopplung, zeitbehaftete Automaten</li> <li>• Verhaltensspezifikation und Model Checking mit Computational Tree Logic</li> <li>• Modellierung durch Ablaufmodelle</li> <li>• Identifikation von Schreib- und Lesekonflikten sowie Datenabhängigkeiten</li> <li>• Modellierung von Kommunikations- und Synchronisationsmustern</li> <li>• Erweiterung der Programmierfähigkeit von sequentiellen auf nebenläufige Programme anhand des Thread-Konzepts von Java und dem mit synchronized-Methoden verbundenen Monitorkonzept</li> <li>• darauf aufbauend: Implementierung von höheren Konzepten der Kommunikation und Synchronisation (wie sie teilweise in Java schon vordefiniert sind, z. B. Locking)</li> <li>• Abschließend wird ein Programmrahmen für eine allgemeine Synchronisationsmethode erstellt</li> </ul>			
<b>Literaturangaben</b> siehe Vorlesung			
WIWI-C0373 Übung: Concurrency im Modul WIWI-M0795: Concurrency			

Modul: Design und Architektur von Softwaresystemen (6 Credits)	
Name im Diploma Supplement	Design and Architecture of Software Systems
Verantwortlich	Prof. Dr. Michael Goedicke
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen Eigenschaften objekt- und komponentenbasierter Softwaresysteme und verfügen über detaillierte Kenntnisse der Prinzipien der objektorientierten Modellierung</li> <li>• sind vertraut mit der UML-Sprachdefinition sowie UML-Diagrammtypen und besitzen die Fähigkeit zur Erstellung und Analyse von ausgewählten UML-Diagrammtypen (mit Schwerpunkt Klassendiagramme)</li> <li>• verfügen über Kenntnisse der Systemspezifikation auf Objekt- und Klassenebene, insbesondere des Konzepts des „Design by Contract“ und dessen Umsetzung in Spezifikations- und Programmiersprachen, um konkrete Designaufgaben auf Objekt- und Klassenebene zu lösen</li> <li>• sind in der Lage, Fragestellungen des Entwurfs von Mikroarchitekturen zu diskutieren und durch die Anwendung von Entwurfsmustern oder vergleichbaren Techniken zu beantworten, um Mikroarchitekturen zielgerichtet zu entwerfen</li> <li>• sind mit den Prinzipien komponentenbasierter Anwendungen vertraut, kennen UML-Diagrammtypen um solche Systeme angemessen zu beschreiben, können komponentenbasierte Architekturen entwerfen und mit Hilfe von Komponentenframeworks realisieren</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 60-90 Minuten). Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob erfolgreich abgelegte Testate als Prüfungsvorleistung verlangt werden. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;1.-4. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Design und Architektur von Softwaresystemen (3 Credits)</li> <li>• Übung: Design und Architektur von Softwaresystemen (3 Credits)</li> </ul>
WIWI-M0341 Modul: Design und Architektur von Softwaresystemen	

Vorlesung: Design und Architektur von Softwaresystemen (3 Credits)			
Name im Diploma Supplement	Design and Architecture of Software Systems		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke Dr. Michael Striewe		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse und Fähigkeiten in der Modellierung von Informatiksystemen, Programmierung und Software Engineering			
<b>Lehrinhalte</b> Die Vorlesung führt zunächst in die grundlegenden Fragestellungen des Designs komplexer Softwaresysteme ein und behandelt Fragestellungen des Designs von Softwaresystemen in aufsteigendem Umfang vom Design auf Objektebene bis zum Entwurf großer komponentenbasierter Systeme. Im Einzelnen werden folgende Themen angesprochen: <ol style="list-style-type: none"> <li>1. Grundlagen des Systemdesigns und der Systemkomposition</li> <li>2. Sichten und Modelle für den Designprozess großer Systeme</li> <li>3. Design und Spezifikation von Systemen auf Objektebene</li> <li>4. Entwurf von Mikroarchitekturen und Makroarchitekturen</li> <li>5. Architekturbeschreibungssprachen</li> <li>6. Bewertung von Architekturen</li> </ol>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• C. Ghezzi, M. Jazayeri, D. Mandrioli: Fundamentals of Software Engineering</li> <li>• Literaturhinweise zu den einzelnen Themen werden in der Vorlesung bekannt gegeben.</li> </ul>			
WIWI-C0329 Vorlesung: Design und Architektur von Softwaresystemen im Modul WIWI-M0341: Design und Architektur von Softwaresystemen			

Übung: Design und Architektur von Softwaresystemen (3 Credits)			
Name im Diploma Supplement	Design and Architecture of Software Systems		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke Dr. Michael Striewe		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen siehe Vorlesung			
Lehrinhalte Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung.			
Literaturangaben Siehe Literaturangaben der Vorlesung.			
WIWI-C0328 Übung: Design und Architektur von Softwaresystemen im Modul WIWI-M0341: Design und Architektur von Softwaresystemen			

Modul (geplante Umstrukturierung): Diskrete Simulation (6 Credits)	
Wichtige Änderungen im Modul	Modul wird voraussichtlich wieder angeboten (ggf. mit anderen inhaltlichen Schwerpunkten) sobald die Professur Nachfolge Prof. Hoßfeld wiederbesetzt ist.
Name im Diploma Supplement	Discrete Simulation
Verantwortlich	Prof. Dr. Michael Goedicke
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• können ein Simulationskonzept erstellen</li> <li>• können ein Simulationsprogramm implementieren</li> <li>• können für relevante Fragestellungen aus der Praxis Simulationsmodelle erstellen</li> <li>• können Simulationsexperiment durchführen und auswerten</li> </ul>
Praxisrelevanz	Die Leistungsbewertung von Systemen und relevante praktische Fragestellungen können oft nur durch Simulationen beantwortet werden. Dadurch ist ein starker Praxisbezug gegeben.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90-120 Minuten) oder mündlichen Prüfung (in der Regel: 20-40 Minuten); die konkrete Prüfungsform – Klausur versus mündliche Prüfung – wird innerhalb der ersten Wochen der Vorlesungszeit von der zuständigen Dozentin oder dem zuständigen Dozenten festgelegt.  Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme Prüfungsvorleistung oder aber Bestandteil der Prüfung ist. Ist letzteres der Fall, so bilden die Teilleistungen zusammen mit der Abschlussprüfung eine zusammengesetzte Prüfung mit einer Endnote. Bestandene Prüfungsvorleistungen/Teilleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Wilnf Bachelor 2010-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Vertiefungsrichtung "Technik und Sicherheit betrieblicher Kommunikationssysteme" &gt;5.-6. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Diskrete Simulation (3 Credits)</li> <li>• Übung: Diskrete Simulation (3 Credits)</li> </ul>
WIWI-M0720 Modul: Diskrete Simulation	

Vorlesung: Diskrete Simulation (3 Credits)			
Name im Diploma Supplement	Discrete Simulation		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörerschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Hilfreich sind Grundkenntnisse aus der Statistik und Stochastik sowie die Vorlesungen „Stochastik für Informatiker“ und „Modelle der Informatik“.			
<b>Abstract</b> Es wird eine Übersicht über Techniken der diskreten, ereignisorientierten Simulation vermittelt, wobei das Erstellen von Simulationsmodellen und die statistische Auswertung von Simulationsexperimenten im Vordergrund stehen.			
<b>Lehrinhalte</b> Es wird ein Grundverständnis von unterschiedlichen Simulationstechniken wie Ereignisorientierter Simulation oder Monte-Carlo Simulation vermittelt. Zu den Grundlagen zählen das Design von Simulationsstudien und –experimenten, sowie die Konzepte der diskreten Simulation. Die Implementierung einer eigenen Simulation steht hierbei im Fokus, welche die Erfassung von Statistiken sowie die Erzeugung von Zufallszahlen beinhaltet. Insbesondere soll auch ein Verständnis der statistischen Auswertung von Simulationsstudien und Interpretation der Simulationsergebnisse vermittelt werden. Die Erzeugung und Simulation von speziellen Zufallsprozessen, wie räumliche Punktprozesse, werden ebenfalls vorgestellt.			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• A. M. Law; W. D. Kelton: Simulation Modeling and Analysis; McGrawHill, 2006</li> <li>• J. Banks; S. S. Carson; B. L. Nelson; D. M. Nicol: Handbook of Simulation; John Wiley, 2013</li> <li>• H. Kobayashi; B. L. Mark: System Modeling and Analysis: Foundations of System Performance Evaluation; Prentice Hall, 2008</li> </ul> Literaturangaben und Links werden im Semester auf der Webseite des Lehrstuhls zur Verfügung gestellt.			
<b>didaktisches Konzept</b> Vorlesung			
WIWI-C0937 Vorlesung: Diskrete Simulation im Modul WIWI-M0720: Diskrete Simulation			

Übung: Diskrete Simulation (3 Credits)			
Name im Diploma Supplement	Discrete Simulation		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse in der Programmierung und Modelle der Informatik.			
<b>Abstract</b> In den Übungen wenden die Studierenden den in der Vorlesung erarbeiteten Stoff in im Rahmen von Programmierübungen und im Rahmen von theoretischen Fragen praktisch an. Das Ziel der Übungen ist die praktische Umsetzung der in der Vorlesung erarbeiteten Konzepte. Insbesondere werden unterschiedliche Aspekte für Computer-gestützte Simulation implementiert werden.			
<b>Lehrinhalte</b> Die Implementierung einer eigenen Simulation steht im Fokus, welche die Erfassung von Statistiken sowie die Erzeugung von Zufallszahlen beinhaltet. Insbesondere soll auch ein Verständnis der statistischen Auswertung von Simulationsstudien und Interpretation der Simulationsergebnisse vermittelt werden.			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Übungsblätter im Semester online erhältlich.</li> </ul> Siehe Literaturangaben der Vorlesung.			
<b>didaktisches Konzept</b> Die Übungen bestehen aus: <ul style="list-style-type: none"> <li>• Programmierübungen, bei denen die Implementierung einer eigenen Simulation im Fokus steht und die von den Studierenden einzeln oder in Gruppen bearbeitet werden. Anhand von Aufgabenblättern werden die Studierenden an die Implementierung von Simulationen und die Durchführung von Simulationsstudien herangeführt. Das Programmieraufgabenblatt wird in der Übung besprochen.</li> <li>• Theoretische Übungsaufgaben zum Vorlesungsstoff werden als Aufgabenblatt ausgegeben, die von den Studierenden bearbeitet werden. Die Lösungen zum Aufgabenblatt werden in der Übung vorgestellt und besprochen werden.</li> <li>• Präsenzübungen, bei denen Aufgabenstellungen zum Vorlesungsstoff von den Studierenden bearbeitet und anschließend besprochen werden.</li> </ul> Daneben werden weitere Internetbasierte Übungsmöglichkeiten angeboten, über die der Vorlesungsstoff intensiv nachbereitet werden kann.			
WIWI-C0938 Übung: Diskrete Simulation im Modul WIWI-M0720: Diskrete Simulation			

Modul: Distributed Objects & XML (6 Credits)	
Name im Diploma Supplement	Distributed Objects, XML & UML
Verantwortlich	Prof. Dr. Michael Goedicke
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• besitzen Kompetenzen in der Entwicklung und Bewertung verteilter, komponentenbasierter Anwendungssysteme</li> <li>• kennen die relevanten Grundlagen bezüglich der Unterschiede sowie Vor- und Nachteile zentral bzw. verteilt organisierter Systeme</li> <li>• können darauf aufbauend mit konkreten Plattformen und Frameworks arbeiten, die die Grundlage für verteilte und komponentenbasierte Systeme darstellen</li> <li>• kennen Eigenschaften objekt- und komponentenbasierter Softwaresysteme</li> <li>• kennen Eigenschaften verteilter und heterogener Softwaresysteme, die orts- und plattform-übergreifend arbeiten</li> <li>• können mit den grundlegenden Eigenschaften relevanter Protokolle, Sprachen und Frameworks für verteilte Softwaresysteme umgehen</li> <li>• können auf dieser Basis mit Plattformen für serverseitige, komponentenorientierte Systeme grundlegende Anwendungen entwickeln</li> <li>• sind in der Lage, die Technologien und ihr Zusammenspiel zu bewerten und deren Vor- und Nachteile abwägen</li> <li>• können mit objektorientierter Middleware Verbindungen herstellen und Daten zwischen laufenden Anwendungen auf verschiedenen Rechnern im Netzwerk austauschen</li> <li>• können serverseitige Anwendungen auf Basis der Java Enterprise Plattform mit Webkomponenten, Geschäftslogikkomponenten und Persistenzkomponenten entwickeln</li> <li>• sind in der Lage, das Zusammenspiel dieser Technologien zur Entwicklung vollständiger Anwendungen zu bewerten</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 60-90 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Network Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Network Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Network Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Distributed Objects &amp; XML (3 Credits)</li> <li>• Übung: Distributed Objects &amp; XML (3 Credits)</li> </ul>
WIWI-M0336 Modul: Distributed Objects & XML	

Vorlesung: Distributed Objects & XML (3 Credits)			
Name im Diploma Supplement	Distributed Objects, XML & UML		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke Dr. Michael Striewe		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse in Programmierung, Kommunikationsnetze und Software Engineering			
<b>Lehrinhalte</b> Es wird eine Übersicht über den Aufbau von verteilten Systemen gegeben, die auf einer objektorientierten Strukturierung beruhen. Des Weiteren wird eine Übersicht über die Beschreibung von Strukturen gegeben, für deren Formulierung XML eingesetzt werden kann. <ol style="list-style-type: none"> <li>1. Grundlegende Eigenschaften von verteilten Systemen, Anforderungen an verteilte Systeme, Kriterien für verteilte und zentral organisierte Systeme, Nichtfunktionale Eigenschaften</li> <li>2. Design Verteilter Objektsysteme, UML, MetaObjektmodell verteilter Objektsysteme, Objektlebenszyklus in verteilten Systemen</li> <li>3. Prinzipien objektorientierter Middleware, Einordnung in OSI Referenzmodell, Typen von Middleware, RPC, IDL, Einfluss von Middleware Plattform auf SW-Architektur</li> <li>4. Java RMI / Corba, Interfaces, Remote Objects, SW-Architektur, jeweils für die beiden Vertreter mit Bewertung der Vor- und Nachteile</li> <li>5. Heterogenität und XML in Verteilten Systemen, Aspekte und Probleme der Heterogenität, XML Struktur und Anwendungen, insbesondere in verteilten Systemen (SOAP, RDF, Schema)</li> <li>6. Application Server, Ziele von ASP, standardisierte Realisierung von nichtfunktionalen Eigenschaften der Verteilung mit ASP (hier J2EE), Struktur von J2EE, Anwendungen, Bewertung</li> </ol>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• W. Emmerich: Engineering Distributed Objects; Wiley 2000</li> </ul>			
WIWI-C0327 Vorlesung: Distributed Objects & XML im Modul WIWI-M0336: Distributed Objects & XML			



## Übung: Distributed Objects & XML (3 Credits)

Name im Diploma Supplement	Distributed Objects, XML & UML		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke Dr. Michael Striewe		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen siehe Vorlesung			
Lehrinhalte Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung.			
Literaturangaben Siehe Literaturangaben der Vorlesung.			
WIWI-C0326 Übung: Distributed Objects & XML im Modul WIWI-M0336: Distributed Objects & XML			

Modul: Formale Methoden des Software Engineering (6 Credits)	
Name im Diploma Supplement	Formal Methods in Software Engineering
Verantwortlich	Prof. Dr. Michael Goedicke
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen die Grundbegriffe der syntaxbasierten Semantikdefinition für formale (Spezifikations-)Sprachen sowie die zugehörigen Methoden</li> <li>• besitzen fundierte Kenntnisse zur formalen, logikbasierten Spezifikation von Softwaresystemen sowie zu den Möglichkeiten und Grenzen der Analyse solcher formaler logikbasierter Spezifikationen</li> <li>• können natürlichsprachliche Aussagen als logische Aussagen formulieren und diese auswerten sowie formale Beweise aufstellen</li> <li>• beherrschen Verfahren zur Beschreibung und Modellierung von parallelen sowie unendlich laufenden Transitionssystemen und können Softwaresysteme zustandsbasiert zu modellieren</li> <li>• können den gesamten Prozess von der formalen Spezifikation der Anforderungen über die geeignete Modellierung eines Systems bis hin zur Verifikation des Modells erläutern und in jedem Schritt geeignete Verfahren anwenden</li> <li>• können die Konzepte der automatischen Verifikation erläutern, die zugehörigen Algorithmen skizzieren, erläutern und anwenden</li> <li>• kennen grundsätzliche Grenzen der automatischen Verifikation und können Systeme benennen, die nicht automatisch verifiziert werden können bzw. Maßnahmen benennen, die eine automatische Verifikation ermöglichen</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Wilnf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Formale Methoden des Software Engineering (3 Credits)</li> <li>• Übung: Formale Methoden des Software Engineering (3 Credits)</li> </ul>
WIWI-M0262 Modul: Formale Methoden des Software Engineering	

Vorlesung: Formale Methoden des Software Engineering (3 Credits)			
Name im Diploma Supplement	Formal Methods in Software Engineering		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende und vertiefte Kenntnisse in Programmierung und Software Engineering			
<b>Lehrinhalte</b> Es wird eine Übersicht über die Themen formale Spezifikation und Analyseverfahren von Softwaresystemen gegeben, die folgende Inhalte umfasst: <ol style="list-style-type: none"> <li>1. Grundlagen der Sprachdefinition</li> <li>2. Logik als formale Sprache und Kalküle für Model Checking und automatisches Beweisen</li> <li>3. Systemmodellierung durch Transitionssysteme</li> <li>4. Konzepte und Algorithmen des Model Checking</li> <li>5. Bewertungskriterien für die Anwendung von automatischen Beweisverfahren in der Softwaretechnik</li> </ol>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• D. Peled: Software Reliability Methods; Springer, 2001</li> <li>• J. Magee, J. Kramer: Concurrency: State Models Java Programs Wiley, 1999</li> <li>• Chin-Liang Chang, Richard Char-Tung Lee: Symbolic Logic and Mechanical Theorem Proving (Computer Science Classics); Academic Press, 1973</li> </ul>			
WIWI-C0325 Vorlesung: Formale Methoden des Software Engineering im Modul WIWI-M0262: Formale Methoden des Software Engineering			

Übung: Formale Methoden des Software Engineering (3 Credits)			
Name im Diploma Supplement	Formal Methods in Software Engineering		
Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>		
Lehrperson	Prof. Dr. Michael Goedicke		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen siehe Vorlesung			
Lehrinhalte Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung.			
Literaturangaben Siehe Literaturangaben der Vorlesung.			
WIWI-C0324 Übung: Formale Methoden des Software Engineering im Modul WIWI-M0262: Formale Methoden des Software Engineering			

<b>Modul: Informations- und Softwarevisualisierung (6 Credits)</b>	
Name im Diploma Supplement	Information and Software Visualization
Verantwortlich	Jun.-Prof. Dr. Fabian Beck
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• erkennen die Möglichkeiten und Grenzen der Datenvisualisierung und können diese auf konkrete Anwendungsbeispiele übertragen</li> <li>• verstehen Grundlagen der visuellen Wahrnehmung und Kognition sowie deren Implikationen auf die visuelle Darstellung von Daten</li> <li>• besitzen einen fundierten Überblick über Möglichkeiten zur visuellen Repräsentation abstrakter Daten und können Visualisierungstechniken auf neue Problemstellungen anpassen und Gestaltungsentscheidungen begründen</li> <li>• beherrschen die Integration von Visualisierungstechniken mit Interaktionstechniken und algorithmischen Lösungen</li> <li>• erkennen gewinnbringende Einsatzmöglichkeiten interaktiver Visualisierungssysteme in der Softwaretechnik</li> <li>• können interaktive Visualisierungssysteme als Software umsetzen</li> <li>• können quantitative und qualitative Nutzerstudien planen, durchführen und auswerten</li> </ul>
Praxisrelevanz	In der Praxis müssen Daten nicht nur verarbeitet, sondern auch den Nutzern verständlich präsentiert werden. Die Visualisierung übernimmt eine Mittlerrolle zwischen Mensch und Computer. Visualisierungen kommunizieren Daten verständlich und erlauben es dem Nutzer darüber hinaus, sie interaktiv zu analysieren. Die vorgestellten Visualisierungstechniken werden an Praxisbeispielen insbesondere aus der Softwaretechnik illustriert, sind aber auch auf andere Anwendungen übertragbar.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten). Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme an der Übung (mindestens 50% der Übungspunkte) als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Network Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• Wilnf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Informations- und Softwarevisualisierung (3 Credits)</li> <li>• Übung: Informations- und Softwarevisualisierung (3 Credits)</li> </ul>
WIWI-M0787 Modul: Informations- und Softwarevisualisierung	

Vorlesung: Informations- und Softwarevisualisierung (3 Credits)			
Name im Diploma Supplement	Information and Software Visualization		
Anbieter	Juniorprofessur für Visualisierung <a href="https://www.vis.wiwi.uni-due.de/">https://www.vis.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Fabian Beck		
SWS	2	Sprache	deutsch/englisch
Turnus	Sommersemester	maximale Hörschaft	50
<b>empfohlenes Vorwissen</b> Grundkenntnisse in Programmierung, Datenstrukturen und Algorithmen, Software Engineering			
<b>Abstract</b> Erst Visualisierungen machen komplexe Daten wie multivariate Zeitreihen und Netzwerkstrukturen lesbar und verständlich. Techniken für eine solche Datenvisualisierung erzeugen aus einem gegebenen Datensatz automatisch eine visuelle Repräsentation, die häufig in eine interaktive Nutzeroberfläche eingebunden ist. Diese Vorlesung gibt einen breiten Überblick zu Methoden der Informationsvisualisierung. Ansätze der visuellen Analytik kombinieren Visualisierungen mit Interaktionsmechanismen und algorithmischen Lösungen. Als spezieller Anwendungsfall solcher interaktiven Visualisierungen geht die Veranstaltung vertieft auf Softwarevisualisierungen ein. Hier werden Entwickler beim Verstehen, Erweitern und Verbessern von Softwaresystemen unterstützt. Die Veranstaltung beleuchtet darüber hinaus, wie Benutzer Visualisierungen wahrnehmen und verarbeiten und wie Benutzerverhalten in Studien wissenschaftlich untersucht werden kann.			
<b>Lehrinhalte</b> Grundlagen: <ul style="list-style-type: none"> <li>• Diagramme und Infografiken</li> <li>• Visuelle Wahrnehmung und Kognition</li> </ul> Informationsvisualisierung: <ul style="list-style-type: none"> <li>• Multivariate Daten und Mengen</li> <li>• Hierarchien und Netzwerke</li> <li>• Zeitreihen und dynamische Daten</li> <li>• Dokumente und Medien</li> </ul> Visuelle Analytik: <ul style="list-style-type: none"> <li>• Interaktionstechniken</li> <li>• Integration algorithmischer Lösungen</li> </ul> Softwarevisualisierung <ul style="list-style-type: none"> <li>• Struktur und Architektur</li> <li>• Evolution von Software</li> <li>• Ausführungsverhalten</li> </ul> Evaluation: <ul style="list-style-type: none"> <li>• Quantitative Nutzerstudien</li> <li>• Qualitative Nutzerstudien</li> </ul>			
<b>Literaturangaben</b> Literatur wird in der Veranstaltung bzw. auf der Homepage der Arbeitsgruppe bekannt gegeben.			
<b>didaktisches Konzept</b> Die Vorlesung wird möglichst interaktiv gestaltet. Diskussionsfragen, Abstimmungen und kurze praktische Übungen wechseln sich mit vorgetragenen Inhalten ab. Die Vorlesung ist eng mit einer Übung verzahnt, in der die Inhalte der Vorlesung auf praktische Beispiele angewandt und implementiert werden.			
WIWI-C1022 <b>Vorlesung: Informations- und Softwarevisualisierung</b> im Modul WIWI-M0787: Informations- und Softwarevisualisierung			

Übung: Informations- und Softwarevisualisierung (3 Credits)			
Name im Diploma Supplement	Information and Software Visualization		
Anbieter	Juniorprofessur für Visualisierung <a href="https://www.vis.wiwi.uni-due.de/">https://www.vis.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Fabian Beck		
SWS	2	Sprache	deutsch/englisch
Turnus	Sommersemester	maximale Hörschaft	50
empfohlenes Vorwissen siehe Vorlesung			
<b>Lehrinhalte</b> Praktische Übungen zu den Inhalten der Vorlesung, unter anderem: <ul style="list-style-type: none"> <li>• Konzeption und Diskussion von Visualisierungen</li> <li>• Implementierung von Visualisierungstechniken</li> <li>• Reflektion aktueller Forschungsliteratur</li> </ul>			
<b>Literaturangaben</b> siehe Vorlesung			
<b>didaktisches Konzept</b> Die Studierenden erarbeiten in bewerteten Übungsaufgaben selbständig Visualisierungskonzepte für konkrete Problemstellungen und implementieren diese als interaktive Software. Impulsreferate, gemeinsame Besprechungen der Ergebnisse und vertiefende Diskussionen prägen den Charakter der Veranstaltung.			
WIWI-C1023 <b>Übung: Informations- und Softwarevisualisierung</b> im Modul WIWI-M0787: Informations- und Softwarevisualisierung			

<b>Modul: Kommunikationsnetze 2 (6 Credits)</b>	
<b>Name im Diploma Supplement</b>	Communication Networks 2
<b>Verantwortlich</b>	Prof. Dr. Pedro José Marrón
<b>Voraussetzungen</b>	Siehe Prüfungsordnung.
<b>Workload</b>	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 60 Stunden</li> <li>• Prüfungsvorbereitung: 60 Stunden</li> </ul>
<b>Dauer</b>	Das Modul erstreckt sich über 1 Semester.
<b>Qualifikationsziele</b>	Die Studierenden <ul style="list-style-type: none"> <li>• haben einen qualifizierten Überblick über aktuelle Funktionen in TCP/IP-basierten sowie drahtlosen Netzen und die zugehörigen Kommunikationsprotokolle,</li> <li>• kennen die grundlegenden Algorithmen, die in den vorgestellten Protokollen verwendet werden,</li> <li>• können anhand gestellter Anforderungen eine geeignete Technologieauswahl vornehmen,</li> <li>• können die in der Vorlesung vorgestellten Konzepte und Protokolle im realen System umsetzen,</li> <li>• verstehen die dabei anfallenden Konfigurationsaufgaben und können diese ausführen.</li> </ul>
<b>Praxisrelevanz</b>	Kenntnisse zu den unterschiedlichen Typen von Kommunikationsnetzen und deren Protokollarchitekturen sind für eine sinnvolle Technologieauswahl in der Praxis notwendig.
<b>Prüfungsmodalitäten</b>	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90-120 Minuten) oder mündlichen Prüfung (in der Regel: 30 Minuten); die konkrete Prüfungsform - Klausur versus mündliche Prüfung - wird innerhalb der ersten Wochen der Vorlesungszeit von der zuständigen Dozentin oder dem zuständigen Dozenten festgelegt.  Prüfungsvorleistung: Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme an der Übung (mindestens 50% der Übungspunkte) als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
<b>Verwendung in Studiengängen</b>	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 2 (Informatik) &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Bachelor 2013&gt;Wahlpflichtbereich &gt;Profil "Network Systems Engineering" &gt;1.-6. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• Wilnf Bachelor 2010-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Vertiefungsrichtung "Technik und Sicherheit betrieblicher Kommunikationssysteme" &gt;5.-6. Fachsemester, Wahlpflicht</li> </ul>
<b>Bestandteile</b>	<ul style="list-style-type: none"> <li>• Vorlesung: Kommunikationsnetze 2 (3 Credits)</li> <li>• Übung: Kommunikationsnetze 2 (3 Credits)</li> </ul>
WIWI-M0221 Modul: <b>Kommunikationsnetze 2</b>	



Vorlesung: Kommunikationsnetze 2 (3 Credits)			
Name im Diploma Supplement	Communication Networks 2		
Anbieter	Networked Embedded Systems <a href="http://www.nes.uni-due.de/">http://www.nes.uni-due.de/</a>		
Lehrperson	Prof. Dr. Pedro José Marrón		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Erforderliche Module: Kommunikationsnetze 1			
<b>Abstract</b> Kommunikation ist ein Querschnittsthema das heutzutage alle Bereiche der praktischen Informatik beeinflusst. Aufbauend auf der Vorlesung "Kommunikationsnetze 1" werden in dieser Vorlesung weitere Aspekte, Funktionen und Kommunikationsprotokolle TCP/IP-basierter Netze behandelt. Dabei werden einerseits bereits in "Kommunikationsnetze 1" angesprochene Themen vertieft, andererseits werden aber auch dort nicht behandelte, für das heutige Internet wichtige Themenbereiche, wie bspw. drahtlose Netze und deren Kommunikation behandelt.			
<b>Lehrinhalte</b> <ul style="list-style-type: none"> <li>• Überblick über Grundbegriffe der technischen Kommunikation, der geschichteten Protokollarchitekturen und das OSI-Referenzmodell.</li> <li>• Routing und Routing-Protokolle: Link State Routing, Distance Vector Routing, RIP, OSPF, BGP.</li> <li>• Mechanismen und Protokolle der Transportschicht: UDP, TCP, SCTP, DCCP, Automatic Repeat Request, Flow Control, Congestion Control.</li> <li>• Infrastruktur-Protokolle: NAT, PAT, DHCP, DNS.</li> <li>• Drahtlose und mobile Netzwerke: IEEE 802.11, IEEE 802.15.4, Bluetooth, Mobilfunk.</li> <li>• Internet der Dinge: 6LoWPAN, RPL, CoAP, MQTT.</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Vorlesungsfolien „Kommunikationsnetze 2“ (im Semester online erhältlich)</li> <li>• J. Kurose, K. Ross: Computer Networking: A Top-Down Approach</li> <li>• A. Tannenbaum: Computer Networks</li> <li>• Weitere Literaturangaben und Links werden im Semester zur Verfügung gestellt.</li> </ul>			
<b>didaktisches Konzept</b> Vorlesung			
WIWI-C0384 Vorlesung: <b>Kommunikationsnetze 2</b> im Modul WIWI-M0221: Kommunikationsnetze 2			

Übung: Kommunikationsnetze 2 (3 Credits)			
Name im Diploma Supplement	Communication Networks 2		
Anbieter	Networked Embedded Systems <a href="http://www.nes.uni-due.de/">http://www.nes.uni-due.de/</a>		
Lehrperson	Prof. Dr. Pedro José Marrón		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> <i>Erforderliches Modul:</i> Kommunikationsnetze 1 <i>Notwendige Voraussetzungen:</i> Teilnahme an der Vorlesung „Kommunikationsnetze 2“, Programmierkenntnisse <i>Sinnvoll:</i> Grundkenntnisse im Umgang mit Unix-Betriebssystemen (z.B. Linux, FreeBSD, Solaris, MacOS X, ...)			
<b>Abstract</b> Siehe Abstract der Vorlesung.			
<b>Lehrinhalte</b> Die Übungen umfassen sowohl theoretische, als auch praktische Inhalte in Form von einerseits zu verwendenden und andererseits zu implementierenden Programmen, welche die in der Vorlesung vorgestellten Konzepte und Protokolle nutzen bzw. realisieren. Dadurch werden Möglichkeiten geschaffen, praktische Erfahrungen im Umgang mit und der Entwicklung von netzwerkbasierenden Anwendungen zu erwerben.			
<b>Literaturangaben</b> siehe Vorlesung			
<b>didaktisches Konzept</b> Theoretische Übungen behandeln und erweitern die in der Vorlesung besprochenen Inhalte. Hierzu werden Aufgabenblätter ausgegeben, welche nach deren Bearbeitung in der Übung besprochen werden. Praktische Übungen vertiefen die theoretischen Grundlagen durch die Verwendung und Implementierung von Protokollen und Anwendungen, deren Schwerpunkt die Netzwerkkommunikation darstellt. Dadurch können kennengelernte Konzepte und Protokolle im realen System erprobt werden, um Praxiskenntnisse im Umgang mit diesen zu erwerben.			
WIWI-C0383 Übung: <b>Kommunikationsnetze 2</b> im Modul WIWI-M0221: Kommunikationsnetze 2			

Modul: Konzepte und Implementierung Objektorientierter Programmiersprachen (6 Credits)	
Name im Diploma Supplement	Concepts and Implementation of Object-Oriented Programming Languages
Verantwortlich	Prof. Dr. Volker Gruhn
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen unterschiedliche Konzepte objektorientierter Programmiersprachen</li> <li>• sind in der Lage, die Semantik von Applikationen unter Verwendung ausgewählter Konstrukte zu bestimmen, als auch formale Beschreibungen ausgewählter Konstrukte zu erstellen</li> <li>• beherrschen den praktischen Umgang mit unterschiedlichen objektorientierten Programmiersprachkonstrukten und verstehen den Einfluß von solchen Konstrukten auf die resultierenden Architekturen</li> <li>• beherrschen den Umgang mit formalen Konstrukten für den Entwurf von Programmiersprachkonstrukten</li> <li>• können Fehler in Programmen anhand von formalen Beschreibungen identifizieren und Typfehler nachvollziehen</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits)</li> <li>• Übung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits)</li> </ul>
WIWI-M0218 Modul: Konzepte und Implementierung Objektorientierter Programmiersprachen	

Vorlesung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits)			
Name im Diploma Supplement	Concepts and Implementation of Object-oriented Programming Languages		
Anbieter	Lehrstuhl für Software-Engineering, insb. mobile Anwendungen <a href="http://www.se.wiwi.uni-due.de/">http://www.se.wiwi.uni-due.de/</a>		
Lehrperson	Dr. Stefan Hanenberg		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse einer objektorientierten Programmiersprache (z.B. Java)			
<b>Lehrinhalte</b> Die Objektorientierung spielt heutzutage in sehr vielen Bereichen der Softwareentwicklung eine entscheidende Rolle. So genannte „Enterprise-Frameworks“, welche für eine Vielzahl von Diensten verantwortlich sind, die in den heutigen „Business-Applications“ zum Einsatz kommen, basieren massiv auf objektorientierten Konzepten. Somit ist die Studie der zugrunde liegenden objektorientierten Konstrukte eine wichtige Voraussetzung um die entsprechenden Frameworks und ihre Architekturen zu verstehen und anzuwenden. Der Begriff der Objektorientierung vereinigt auf Programmiersprachenebene eine Menge von Konzepten, die in unterschiedlichen Programmiersprachen unterschiedliche Ausprägungen finden. Als Beispiel sei an dieser Stelle die Vererbung genannt, welche in unterschiedlichen Sprachen unterschiedlich implementiert ist (Einfachvererbung vs. Mehrfachvererbung, objektbasierte vs. klassenbasierte Vererbung, static dispatching vs. dynamic dispatching, multidispatching, etc.). Für die Anwendung einer Programmiersprache hat die Existenz bestimmter Konzepte erheblichen Einfluss auf die resultierenden Softwarearchitekturen. So hat zum Beispiel in Java die Nichtexistenz von multidispatching zur Konsequenz, dass der Entwickler gegebenenfalls in seinen Anwendungen Vorkehrungen treffen muss, welche die Ausführung der „richtigen“ Methoden garantieren. Die Vorlesung bietet einen breiten Überblick an unterschiedlichen Konzepten objektorientierter Programmiersprachen und deren Semantik, wobei insbesondere Typsysteme (und deren unterschiedliche Ausprägungen) in den Fokus der Betrachtung gezogen werden. Dazu werden gängige Techniken zur Beschreibung der formalen Semantik von Programmiersprachen (Lambda-Kalkül, Featherweight Java als Model für die Sprache Java, etc.) eingeführt und angewendet.			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Bruce, Kim B.: Foundations of Object-Oriented Languages, MIT Press, 2002.</li> <li>• Pierce, Benjamin C.: Types and Programming Languages, MIT Press, 2002.</li> <li>• Abadi, M.; Cardelli, L.: A Theory of Objects, Springer-Verlag, 1996.</li> </ul>			
WIWI-C0282 Vorlesung: Konzepte und Implementierung Objektorientierter Programmiersprachen im Modul WIWI-M0218: Konzepte und Implementierung Objektorientierter Programmiersprachen			

**Übung: Konzepte und Implementierung Objektorientierter Programmiersprachen (3 Credits)**

Name im Diploma Supplement	Concepts and Implementation of Object-oriented Programming Languages		
Anbieter	Lehrstuhl für Software-Engineering, insb. mobile Anwendungen <a href="http://www.se.wiwi.uni-due.de/">http://www.se.wiwi.uni-due.de/</a>		
Lehrperson	Dr. Stefan Hanenberg		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Keines			
<b>Lehrinhalte</b> Im Rahmen der Übung zur Vorlesung werden zum einen die in der Vorlesung behandelten Konzepte praktisch eingesetzt, als auch die Beschreibungstechniken der Sprachkonzepte angewendet. Konkret werden unter anderem das Dispatch-Verhalten von Java, Smalltalk und CLOS angewendet als auch das Typsystem für Featherweight Java eingesetzt.			
<b>Literaturangaben</b> Siehe Literaturangaben der Vorlesung.			
WIWI-C0281 Übung: Konzepte und Implementierung Objektorientierter Programmiersprachen im Modul WIWI-M0218: Konzepte und Implementierung Objektorientierter Programmiersprachen			

Modul: Mathematische Algorithmen der Informatik (6 Credits)	
Name im Diploma Supplement	Mathematical Algorithmus in Computer Science
Verantwortlich	Dipl. Math. Alexander Lewintan
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• besitzen fachliche Kompetenzen in grundlegenden mathematischen Themen und ihrer Umsetzung in programmierte Algorithmen</li> <li>• sind in der Lage, diese Themen zu erläutern und die Eigenschaften von zugehörigen Algorithmen und deren praktische Einsatzmöglichkeiten zu beurteilen</li> <li>• können geeignete mathematische Methoden auswählen, zugehörige Algorithmen entwickeln und implementieren</li> <li>• können diese Algorithmen praktisch erproben und die erzielten Ergebnisse interpretieren</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 2 (Informatik) &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-2. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung mit integrierter Übung: Mathematische Algorithmen der Informatik (6 Credits)</li> </ul>

WIWI-M0409 Modul: Mathematische Algorithmen der Informatik

Vorlesung mit integrierter Übung: Mathematische Algorithmen der Informatik (6 Credits)			
Name im Diploma Supplement	Mathematical Algorithmus in Computer Science		
Anbieter	Dipl.-Math. Alexander Lewintan <a href="http://www.icb.uni-due.de">www.icb.uni-due.de</a>		
Lehrperson	Dipl. Math. Alexander Lewintan		
SWS	4	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	20
<b>empfohlenes Vorwissen</b> Es werden Kenntnisse in Linearer Algebra erwartet, wie sie in der Regel in einem Informatik-Bachelorstudium vermittelt werden.			
<b>Abstract</b> In diesem Kurs werden verschiedene für Informatiker relevante mathematische Modelle aus der modernen Mathematik behandelt und geübt.			
<b>Qualifikationsziele</b> Die Studierenden <ul style="list-style-type: none"> <li>• überblicken die Hauptideen der modernen Mathematik</li> <li>• beherrschen deren praktische Anwendung in der Informatik</li> </ul>			
<b>Lehrinhalte</b> <ul style="list-style-type: none"> <li>• Elemente der Zahlentheorie <ul style="list-style-type: none"> <li>• Lineare diophantische Gleichung</li> <li>• Der Euklidische Algorithmus</li> <li>• Lineare Kongruenz</li> <li>• Primzahlen</li> </ul> </li> <li>• Elemente der Gruppen Theorie und RSA-Verfahren</li> <li>• Ringe und Körper, Körpererweiterung</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• S. Bosch: Algebra; Springer Verlag</li> <li>• H. Lüneburg: Gruppen, Ringe, Körper; R. Oldenbourg Verlag</li> <li>• K.-U. Witt: Algebraische und zahlentheoretische Grundlagen für die Informatik; Springer Vieweg Verlag</li> <li>• G. A. Jones and J. M. Jones: Elementary Number Theory; Springer Verlag</li> </ul>			
WIWI-C0590 <b>Vorlesung mit integrierter Übung: Mathematische Algorithmen der Informatik</b> im Modul WIWI-M0409: Mathematische Algorithmen der Informatik			

Modul: Mensch-Computer Interaktion (6 Credits)	
Name im Diploma Supplement	Human-Computer Interaction
Verantwortlich	Jun.-Prof. Dr. Stefan Schneegaß
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen die Methoden und Konzepte der Mensch-Computer-Interaktion.</li> <li>• können Benutzeroberflächen sinnvoll erstellen und bewerten.</li> <li>• verstehen menschliche Aspekte: Wahrnehmung, Motorik, Gedächtnis, Aufmerksamkeit und Problemlösung.</li> <li>• können verschiedene Techniken zur Evaluation von Benutzerschnittstellen wie beispielsweise Experten-Evaluation (Walkthrough, GOMS) oder Benutzer-Evaluation (Think-aloud, Interviews, Auswertung) anwenden.</li> <li>• kennen grundlegende Aspekte, Funktionsweisen und Besonderheiten verschiedener Spezialthemen der Mensch-Computer Interaktion (z.B. AR/VR Systeme, Mobile Interaktion, Wearable Computing, Eyetracking oder Brain-Computer Interfaces).</li> </ul>
Praxisrelevanz	Benutzungsschnittstellen werden immer mehr zum entscheidenden Faktor, der über Erfolg oder Misserfolg einer Anwendung bestimmt. Die Grundlagen der Mensch-Computer Interaktion zu verstehen und anwenden zu können hilft bei der Erstellung besserer und benutzbarere Systeme.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt von einer mündlichen (in der Regel 20-40 Minuten) oder schriftlichen Prüfung (in der Regel 60-90 Minuten). Die genauen Prüfungsmodalitäten werden in den ersten Vorlesungswochen je nach Teilnehmerzahl festgelegt.  Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme an der Übung (mind. 50% der Übungsblätter bearbeitet) als Prüfungsvorleistung Zulassungsvoraussetzung für die Modulprüfung ist. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Network Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung mit integrierter Übung: Mensch-Computer Interaktion (6 Credits)</li> </ul>

WIWI-M0791 Modul: Mensch-Computer Interaktion

Vorlesung mit integrierter Übung: Mensch-Computer Interaktion (6 Credits)			
Name im Diploma Supplement	Human-Computer Interaction		
Anbieter	Juniorprofessur für Mensch-Computer Interaktion <a href="https://www.hci.wiwi.uni-due.de/">https://www.hci.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Stefan Schneegaß		
SWS	4	Sprache	deutsch/englisch
Turnus	Wintersemester	maximale Hörschaft	50
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse der Programmierung.			
<b>Abstract</b> In dieser Vorlesung mit integrierter Übung erhalten Studierende einen Überblick über die Grundlagen sowie detaillierte Kenntnisse zu ausgewählte Spezialthemen der Mensch-Computer Interaktion.			
<b>Lehrinhalte</b> Die Vorlesung besteht aus zwei Teilen. Im ersten Teil werden die Grundlagen der Mensch-Computer Interaktion vermittelt. Im zweiten Teil spezielle Themen vertieft.  Die integrierte Übung besteht aus zwei Teilen. Im ersten Teil werden die Grundlegenden Inhalte aus der Vorlesung durch Übungsblätter vertieft. Im zweiten Teil der Übung werden die gewonnenen Kenntnisse in Gruppenarbeit angewandt. Hier werden jährlich wechselnde Projekte realisiert.			
<b>Literaturangaben</b> Literatur wird in der Veranstaltung bzw. auf der Homepage der Arbeitsgruppe bekannt gegeben.			
<b>didaktisches Konzept</b> Die Vorlesung wird interaktiv gestaltet und die Studierenden durch Diskussionsfragen, Abstimmungen und kurze praktische Aufgaben eingebunden.			

WIWI-C1097 Vorlesung mit integrierter Übung: Mensch-Computer Interaktion im Modul WIWI-M0791: Mensch-Computer Interaktion

<b>Modul: No-Frills Software Engineering (6 Credits)</b>	
Name im Diploma Supplement	No-Frills Software Engineering
Verantwortlich	Prof. Dr. Volker Gruhn
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• beherrschen die Grundprinzipien des "No-Frills Software Engineering" und können diese anwenden</li> <li>• besitzen die Fähigkeit, die Konzepte des "No-Frills Software Engineering" von anderen Vorgehensweisen der SW-Entwicklung abzugrenzen</li> <li>• können Konzepte des "No-Frills Software Engineering" selbständig auswählen und anwenden</li> <li>• analysieren und beurteilen Probleme und Risiken der industriellen SW-Entwicklung und ihre Konsequenzen für Entwicklungsprozesse</li> <li>• analysieren Praxisszenarien und wenden selbständig geeignete Methoden und Werkzeuge des "No-Frills Software Engineering" an</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung <ul style="list-style-type: none"> <li>• in der Gestalt einer Klausur (in der Regel: 90-120 Minuten) oder</li> <li>• in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten).</li> </ul> Die konkrete Prüfungsform – Klausur versus mündliche Prüfung – wird innerhalb der ersten Wochen der Vorlesungszeit von der zuständigen Dozentin oder dem zuständigen Dozenten festgelegt. Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Erstellung eines Portfolios als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Die bzw. der Dozierende nimmt zu Beginn der Lehrveranstaltung die Präzisierung von Art und Umfang der Prüfungsvorleistungen vor. Bestandene Testate haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Wilnf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung mit praktischer Übung: No-Frills Software Engineering (6 Credits)</li> </ul>
WIWI-M0411 Modul: No-Frills Software Engineering	

Vorlesung mit praktischer Übung: No-Frills Software Engineering (6 Credits)			
Name im Diploma Supplement	No-Frills Software Engineering		
Anbieter	Lehrstuhl für Software-Engineering, insb. mobile Anwendungen <a href="http://www.se.wiwi.uni-due.de/">http://www.se.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Volker Gruhn Dr. Marc Hesenius Wilhelm Koop		
SWS	4	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Modellierung, Programmierung, Software-Engineering-Grundlagen			
<b>Abstract</b> Die Vorlesung vermittelt einen Überblick über das Thema „No-Frills Software Engineering“ (NSFE) und festigt das vermittelte Wissen durch die Einbettung praktischer Übungen, Diskussion von Anwendungsszenarien sowie durch begleitende Fallstudien. Das No-Frills-Prinzip (dt.: „ohne Schnickschnack“) auf die Softwareentwicklung angewendet bedeutet die Vereinfachung von Softwareprozessen durch die Konzentration auf die Hauptaktivitäten und deren Umsetzung mit pragmatischen Prinzipien des Software Engineerings.			
<b>Lehrinhalte</b> Nach der Einordnung und Abgrenzung des NFSE zu etablierten Vorgehensmodellen werden die Grundsätze des NFSE detailliert vorgestellt und diskutiert. Die folgenden Themen werden in der Vorlesung unter anderem behandelt: <ul style="list-style-type: none"> <li>• Probleme und Risiken der industriellen SW-Entwicklung und ihre Konsequenzen für Entwicklungsprozesse</li> <li>• Bedeutung von Problemverständnis und Domänenwissen im SW-Prozess</li> <li>• Wertorientierung in der Softwareentwicklung</li> <li>• Anwendung von modernen Konzepten des Software-Engineerings</li> <li>• Flexibilität und Agilität</li> <li>• Kommunikation und Interaktion im Projektteam</li> <li>• Hauptaktivitäten des NFSE</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Volker Gruhn, Clemens Schäfer: No-Frills Software Engineering for Business Information Systems Experience Report. Proceedings of the 2009 Conference on New Trends in Software Methodologies, Tools and Techniques (SoMeT_09), p. 93-105. IOS Press, Amsterdam.</li> <li>• Barry W. Boehm, Kevin J. Sullivan: Software economics: a roadmap. Proceedings of the Conference on The Future of Software Engineering, p. 319-343, Limerick, Ireland, ACM 2000</li> <li>• Bill Curtis, Herb Krasner, Neil Iscoe: A field study of the software design process for large systems. Communications of the ACM, v.31 n.11, p.1268-1287, Nov. 1988</li> <li>• K. Kautz, S. Madsen, and J. Nørbjerg. Persistent problems and practices in information systems development. Information Systems Journal, 17(3):217-239, 2007.</li> <li>• M. M. Lehman: Uncertainty in computer application and its control through the engineering of software. Journal of Software Maintenance: Research and Practice, v.1 n.1, p.3-27, Sept. 1989</li> </ul>			
<b>didaktisches Konzept</b> In der Veranstaltung werden Vorlesung, praktische Übungen und Methoden zur gemeinsamen Erarbeitung kombiniert. Eine Trennung von Vorlesung und Übung ist nicht vorgesehen.			
WIWI-C0593 Vorlesung mit praktischer Übung: No-Frills Software Engineering im Modul WIWI-M0411: No-Frills Software Engineering			



Modul: Paradigmen und Konzepte der Softwareentwicklung (6 Credits)	
Name im Diploma Supplement	Paradigms and Concepts of Software Engineering
Verantwortlich	Prof. Dr. Stefan Eicker
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 30 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 60 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• besitzen die Fähigkeit zur Auswahl angemessener Methoden und Werkzeuge im Rahmen der Entwicklung betrieblicher Anwendungssysteme</li> <li>• kennen konkrete Konzepte im Bereich der Planung und des Entwurfs von Software (Architekturen, Sichten, etc.)</li> <li>• können traditionelle und neue Architekturansätze beurteilen</li> <li>• können Pattern im Rahmen des Softwareentwurfs anwenden und bewerten</li> <li>• vergleichen gängige Praktiken des Qualitätsmanagements sowohl der Entwicklungsprozesse als auch des Softwareprodukts</li> <li>• können die Konzepte des Software-Konfigurationsmanagements erläutern und anwenden</li> <li>• sind in der Lage, Softwareprojekte zu planen und durchzuführen</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 60-90 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 3 (Wirtschaftsinformatik und E1) &gt;Wirtschaftsinformatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich I: Wirtschaftsinformatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	• Vorlesung mit integrierter Übung: Paradigmen und Konzepte der Softwareentwicklung (6 Credits)
WIWI-M0154 Modul: Paradigmen und Konzepte der Softwareentwicklung	

### Vorlesung mit integrierter Übung: Paradigmen und Konzepte der Softwareentwicklung (6 Credits)

Name im Diploma Supplement	Paradigms and Concepts of Software Engineering		
Anbieter	Lehrstuhl für Wirtschaftsinformatik und Softwaretechnik <a href="http://www.softec.wiwi.uni-due.de/">http://www.softec.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Stefan Eicker		
SWS	4	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen keines			
<b>Lehrinhalte</b> <ul style="list-style-type: none"> <li>• Historische Betrachtung/Veränderung der Softwareentwicklung</li> <li>• Konzepte der Objektorientierung</li> <li>• Unified Modeling Language (UML)</li> <li>• Traditionelle Vorgehensmodelle &amp; Moderne / Agile Softwareentwicklung</li> <li>• Software-Architekturen (Ebenen, Sichten, Architekturstile)</li> <li>• Serviceorientierte Architekturen und REST</li> <li>• Software-Design-Patterns</li> <li>• Software-Qualität und -Metriken</li> <li>• Software-Konfigurationsmanagement</li> <li>• Software-Tests</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Balzert, Helmut: Lehrbuch der Software-Technik</li> <li>• Sommerville, Ian: Software Engineering</li> <li>• Weitere Literaturangaben sind zu den jeweiligen Veranstaltungen themenspezifisch in den Vorlesungsunterlagen zu finden</li> </ul>			
WIWI-C0774 Vorlesung mit integrierter Übung: Paradigmen und Konzepte der Softwareentwicklung im Modul WIWI-M0154: Paradigmen und Konzepte der Softwareentwicklung			

Modul: Programmieren in C/C++ (6 Credits)	
Name im Diploma Supplement	Programming in C/C++
Verantwortlich	Prof. Dr. Pedro José Marrón
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 70 Stunden</li> <li>• Prüfungsvorbereitung: 65 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen und verstehen die grundlegenden Konzepte der objektorientierten Methodik.</li> <li>• können die Unterschiede zwischen Java und C/C++ aufzeigen.</li> <li>• können kleinere Beispiele in C++ selbständig unter Nutzung der vorgestellten Konzept und Methodik programmieren.</li> </ul>
Praxisrelevanz	Das Modul lehrt den Umgang mit der sehr praxisrelevanten, objektorientierten Programmiersprache C/C++. Ein Schwerpunkt dieser Veranstaltung ist die Darstellung von Unterschieden zwischen Java und C++. Das Modul ist durch die weite Verbreitung der Programmiersprache C bzw. C++ in Industrie und Wirtschaft sehr praxisrelevant.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90-120 Minuten) oder mündlichen Prüfung (in der Regel: 30 Minuten); die konkrete Prüfungsform – Klausur versus mündliche Prüfung – wird innerhalb der ersten Wochen der Vorlesungszeit von der zuständigen Dozentin oder dem zuständigen Dozenten festgelegt.  Prüfungsvorleistung: Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme an der Übung (mindestens 50% der Übungspunkte) als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Programmieren in C/C++ (3 Credits)</li> <li>• Übung: Programmieren in C/C++ (3 Credits)</li> </ul>
WIWI-M0610 Modul: Programmieren in C/C++	

Vorlesung: Programmieren in C/C++ (3 Credits)			
Name im Diploma Supplement	Programming in C/C++		
Anbieter	Networked Embedded Systems <a href="http://www.nes.uni-due.de/">http://www.nes.uni-due.de/</a>		
Lehrperson	Prof. Dr. Pedro José Marrón		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Programmierkenntnisse sind empfohlen.			
<b>Lehrinhalte</b> Inhalte im Einzelnen: <ul style="list-style-type: none"> <li>• C++ als Erweiterung von C</li> <li>• Zeigerkonzepte</li> <li>• Klassen, Klassen-Hierarchien, einfache und mehrfache Vererbung, Zugriffsschutzmechanismen, virtuelle Basisklassen, virtuelle Funktionen, statisches und dynamisches Binden, Typisierung und Typkonvertierungen</li> <li>• Funktions- und Operator-Überladen</li> <li>• Exception Handling</li> <li>• Templates</li> <li>• Modularität, Namespaces</li> <li>• Libraries</li> <li>• Streams</li> <li>• Standard Template Library (z.B. Algorithmen, Iteratoren, Container)</li> <li>• Datenstrukturen</li> <li>• kleine Projektbeispiele aus den Anwendungsbereichen der Ingenieurwissenschaften</li> </ul>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• Vorlesungsunterlagen „Programmieren in C/C++“, P. J. Marrón (im Semester erhältlich)</li> <li>• Übungsblätter „Programmieren in C/C++“, P. J. Marrón (im Semester erhältlich)</li> <li>• Stroustrup, Bjarne. The C++ Programming Language: Special Edition. Addison Wesley, New York. Special Edition. 2000. ISBN: 978-0201700732.</li> <li>• Stroustrup, Bjarne. The Design and Evolution of C++. Addison Wesley, New York. 1994. ISBN 978-0201543308.</li> <li>• Robert Sedgewick. Algorithmen in C++. Teil 1-4. Addison-Wesley Longman Verlag. 3. Auflage. 2002. ISBN 978-3827370266.</li> </ul>			
WIWI-C0752 Vorlesung: Programmieren in C/C++ im Modul WIWI-M0610: Programmieren in C/C++			

**Übung: Programmieren in C/C++ (3 Credits)**

Name im Diploma Supplement	Programming in C/C++		
Anbieter	Networked Embedded Systems <a href="http://www.nes.uni-due.de/">http://www.nes.uni-due.de/</a>		
Lehrperson	Prof. Dr. Pedro José Marrón		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt

**empfohlenes Vorwissen**

Programmierkenntnisse sind empfohlen.

**Lehrinhalte**

Entsprechend der in der Vorlesung „Programmieren in C/C++“ vorgestellten Konzepte und Methoden werden in den Übungen anhand von Beispielprogrammen praktisch vermittelt.

Die Inhalte orientieren sich dabei am Inhalt der Vorlesung:

- C++ als Erweiterung von C
- Zeigerkonzepte
- Klassen, Klassen-Hierarchien, einfache und mehrfache Vererbung, Zugriffsschutzmechanismen, virtuelle Basisklassen, virtuelle Funktionen, statisches und dynamisches Binden, Typisierung und Typkonvertierungen
- Funktions- und Operator-Überladen
- Exception Handling
- Templates
- Modularität, Namespaces
- Libraries
- Streams
- Standard Template Library (z.B. Algorithmen, Iteratoren, Container)
- Datenstrukturen
- kleine Projektbeispiele aus den Anwendungsbereichen der Ingenieurwissenschaften

**Literaturangaben**

Siehe Literaturangaben der Vorlesung.

WIWI-C0753 Übung: Programmieren in C/C++ im Modul WIWI-M0610: Programmieren in C/C++

<b>Modul: Requirements Engineering und Management 1 (6 Credits)</b>	
<b>Name im Diploma Supplement</b>	Requirements Engineering and Management 1
<b>Verantwortlich</b>	Prof. Dr. Klaus Pohl
<b>Voraussetzungen</b>	Siehe Prüfungsordnung.
<b>Workload</b>	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
<b>Dauer</b>	Das Modul erstreckt sich über 1 Semester.
<b>Qualifikationsziele</b>	Die Studierenden <ul style="list-style-type: none"> <li>• kennen und verstehen die grundsätzlichen Ziele und Verantwortlichkeiten des Requirements Engineering und Management im Entwicklungsprozess von softwareintensiven Systemen können das Requirements Engineering Rahmenwerk anwenden, um Requirements Engineering Prozesse in der Praxis zu strukturieren</li> <li>• kennen und verstehen die verschiedenen Aktivitäten innerhalb des Requirements Engineering und deren Abhängigkeiten</li> <li>• kennen verschiedene Techniken zur textuellen Dokumentation von Anforderungen und können diese Techniken anwenden, um qualitativ hochwertige textuelle Anforderungen zu formulieren</li> <li>• kennen verschiedenen Techniken zur modellbasierten Dokumentation von Anforderungen und können diese ergänzend zueinander einsetzen, um die Anforderungen eines softwareintensiven Systems durch grafische Modelle zu beschreiben</li> <li>• kennen verbreitete Methoden zur Systemanalyse und zur Gewinnung und Dokumentation von Anforderungen und können Beurteilen, wann welche Methode zweckmäßig eingesetzt wird</li> <li>• besitzen praktische Erfahrungen in der Anwendung von Techniken zur Gewinnung und Dokumentation von Anforderungen</li> <li>• besitzen praktische Erfahrungen in der Anwendung von Techniken zur textuellen Spezifikation von Anforderungen und der Aufdeckung von Qualitätsmängeln in textuell spezifizierten Anforderungen</li> <li>• besitzen praktische Erfahrungen in der Anwendung von Techniken zur modellbasierten Spezifikation von Anforderungen und dem ergänzenden Einsatz verschiedener Diagrammtypen zur vollständigen Spezifikation der Anforderungen durch Modelle</li> <li>• besitzen praktische Erfahrung in Bezug auf den Einsatz professioneller Werkzeugumgebungen zur Dokumentation und Verwaltung von Anforderungen</li> </ul>
<b>Prüfungsmodalitäten</b>	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90 bis 120 Minuten). Die erfolgreiche Teilnahme an der Übung ist als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung.
<b>Verwendung in Studiengängen</b>	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Kernstudium &gt;Pflichtbereich II: Informatik &gt;3.-4. Fachsemester, Pflicht</li> <li>• BWL Bachelor 2006-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Bereich Volkswirtschaftslehre, Rechtswissenschaft, Wirtschaftsinformatik, Informatik &gt;Vertiefungsbereich Informatik &gt;4.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Bachelor 2013&gt;Wahlpflichtbereich &gt;Profil "Software Systems Engineering" &gt;1.-6. Fachsemester, Wahlpflicht</li> <li>• WiInf Bachelor 2010-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Vertiefungsrichtung "Modellierung und Realisierung betrieblicher Informationssysteme" &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• WiInf Bachelor 2010-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Vertiefungsrichtung "Technik und Sicherheit betrieblicher Kommunikationssysteme" &gt;5.-6. Fachsemester, Wahlpflicht</li> </ul>
<b>Bestandteile</b>	<ul style="list-style-type: none"> <li>• Vorlesung: Requirements Engineering und Management 1 (3 Credits)</li> <li>• Übung: Requirements Engineering und Management 1 (3 Credits)</li> </ul>
WIWI-M0120 Modul: Requirements Engineering und Management 1	

Vorlesung: Requirements Engineering und Management 1 (3 Credits)			
Name im Diploma Supplement	Requirements Engineering and Management 1		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen keines			
<p><b>Abstract</b></p> <p>In den meisten Unternehmen sind Anforderungen an Softwaresysteme oft unklar, widersprüchlich, unvollständig und nicht nachvollziehbar dokumentiert. Existierende Anforderungsspezifikationen (z.B. Lasten- und Pflichtenhefte) sind veraltet. Wichtige Anforderungen werden oft zu spät erkannt oder sogar übersehen. Darüber hinaus werden Anforderungen oft unzureichend realisiert. Die Folgen sind oft unzufriedene Kunden, erhebliche Überschreitungen des Budgets und der Terminplanung, Qualitätsmängel, gescheiterte Entwicklungsprojekte und schlecht wartbare Systeme. Aufgabe des Requirements Engineering (RE) ist es, aus oft vagen und teilweise widersprüchlichen Ideen eine möglichst vollständige, korrekte und widerspruchsfreie Anforderungsspezifikation zu erarbeiten, um diesen aufgeführten Problemen frühzeitig entgegenwirken zu können. In der Praxis werden entsprechenden Tätigkeiten mitunter auch unter andern Benennungen zu finden, wie z.B. der Business Analyse, der Systemanalyse oder dem Anforderungsmanagement.</p>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Rahmenwerk des Requirements Engineering: Kontexttheorie, Aktivitäten des Requirements Engineering, Arten von Anforderungsartefakten und deren Beziehungen, die drei Dimensionen des Requirements Engineering.</li> <li>• Textuelle Spezifikation/Anforderungsdokumentation: Probleme der Anforderungsdokumentation in natürlicher Sprache, Kategorisierung von Mehrdeutigkeit; Qualitätsanforderungen für Anforderungsdokumente; standardisierter Aufbau von Anforderungsdokumenten; Normsprache.</li> <li>• Semiotisches Dreieck, Konzeptuelle Modellierung: Theorie der konzeptuellen Modellierung, Sichtenbildung</li> <li>• Modellbasiertes Requirements Engineering: Anforderungsdokumentation durch Modellen; Einsatz formaler Anforderungsmodelle</li> <li>• Verbreitete Modelle zur Datenmodellierung; Funktionsorientierte Modellierung; Verhaltensmodellierung.</li> <li>• Methoden der Systemanalyse und zur Gewinnung und Dokumentation von Anforderungen</li> </ul>			
<p><b>Literaturangaben</b></p> <ul style="list-style-type: none"> <li>• Pflichtliteratur: <ul style="list-style-type: none"> <li>• Klaus Pohl: Requirements Engineering: Grundlagen, Prinzipien, Techniken, dpunkt.verlag, 2. Aufl., 2008</li> </ul> </li> <li>• Vertiefungsliteratur: <ul style="list-style-type: none"> <li>• CHEN, P.: The Entity Relationship Model - Toward a Unified View of Data. In: ACM Transactions on Database Systems, 1 (1976) 1, S.9-36</li> <li>• HAREL, D.: Statecharts: A Visual Formalism for Complex Systems. In: Science of Computer Programming, 8 (1987) 3, S. 231-274</li> </ul> </li> <li>• Ergänzungsliteratur: <ul style="list-style-type: none"> <li>• K. Pohl: Requirements Engineering. In: A. Kent, J. Williams, C. M. Hall (Hrsg.): Encyclopedia of Computer Science and Technology, Vol. 36, M. Dekker, New York, 1997, S. 345-386.</li> <li>• T. Weyer: Kohärenzprüfung von Anforderungsspezifikationen – Ein Ansatz zur Prüfung der Kohärenz von Verhaltensspezifikationen gegen Eigenschaften des operationellen Kontexts. SVH Verlag, Saarbrücken 2011.</li> <li>• S. Robertson, J. Robertson: Mastering the Requirements Process. 2. Aufl., Addison-Wesley, Upper Saddle River, 2006.</li> <li>• A. van Lamsweerde: Goal-Oriented Requirements Engineering: A Guided Tour. In: Proceedings of the 5th IEEE International Symposium on Requirements Engineering (RE'01), IEEE Computer Society Press, Los Alamitos, 2001, S. 249-263.</li> <li>• K. Weidenhaupt, K. Pohl, M. Jarke, P. Haumer: Scenario Usage in System Development – A Report on Current Practice. IEEE Software, Vol. 15, Nr. 2, 1998, S. 34-45.</li> <li>• A. M. Davis: Software Requirements – Objects, Functions, and States. Prentice Hall, Englewood Cliffs, 1993.</li> <li>• T. DeMarco: Structured Analysis and System Specification. Yourdon Press, New York, 1978.</li> <li>• D. Harel: Statecharts – A Visual Formalism for Complex Systems. In: Science of Computer Programming, Vol. 8, 1987, S. 231-274.</li> <li>• D. M. Berry, E. Kamsties, M. M. Krieger: From Contract Drafting to Software Specification – Linguistic Sources of Ambiguity, A Handbook. Waterloo (Canada), Essen (Germany), Los Angeles (U.S.A.), November 2003.</li> <li>• G. Kotonya, I. Sommerville: Requirements Engineering. Processes and Techniques. Wiley, Chichester, 1997.</li> <li>• A. Sutcliffe: User-Centred Requirements Engineering – Theory and Practice. Springer-Verlag, London, 2002.</li> </ul> </li> </ul>			
WIWI-C0347 Vorlesung: Requirements Engineering und Management 1 im Modul WIWI-M0120: Requirements Engineering und Management 1			

Übung: Requirements Engineering und Management 1 (3 Credits)			
Name im Diploma Supplement	Requirements Engineering and Management 1		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> keines			
<b>Lehrinhalte</b> <ul style="list-style-type: none"> <li>• Anwendung des Rahmenwerk zur Strukturierung und Bewertung von Requirements-Engineering-Prozessen in der Praxis.</li> <li>• Anwendung von Techniken zur textuellen Spezifikation von Anforderungen und zur Aufdeckung von Qualitätsmängeln in textuellen Anforderungen.</li> <li>• Anwendung von Techniken zur modellbasierten Spezifikation von Anforderungen in verschiedenen Modellierungsperspektiven (Informationsstruktur, Funktional, Verhalten) und ergänzender Einsatz verschiedener Diagrammtypen.</li> <li>• Anwendung von Methoden zur Systemanalyse und zur Gewinnung und Dokumentation von Anforderungen.</li> </ul>			
<b>Literaturangaben</b> siehe Vorlesung			
WIWI-C0346 Übung: Requirements Engineering und Management 1 im Modul WIWI-M0120: Requirements Engineering und Management 1			

Modul: Requirements Engineering und Management 2 (6 Credits)	
Name im Diploma Supplement	Requirements Engineering and Management 2
Verantwortlich	Prof. Dr. Klaus Pohl
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen Ziele und Szenarien als fortgeschrittenes erweitertes Mittel zur konzeptuellen Modellierung im Requirements Engineering</li> <li>• kennen die Theorie der essentiellen Systemanalyse</li> <li>• haben detaillierte Kenntnisse hinsichtlich der Validierung im Requirements Engineering</li> <li>• haben detaillierte Kenntnisse hinsichtlich des Managements im Requirements Engineering</li> <li>• können Ziele und Szenarien im Requirements Engineering im Rahmen der Gewinnung und Dokumentation von Anforderungen anwenden</li> <li>• können Techniken der essenziellen Systemanalyse im Requirements Engineering einsetzen</li> <li>• beherrschen grundlegende Techniken zum Management und zur Validierung im Requirements Engineering</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90 bis 120 Minuten). Die erfolgreiche Teilnahme an der Übung ist als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Requirements Engineering und Management 2 (3 Credits)</li> <li>• Übung: Requirements Engineering und Management 2 (3 Credits)</li> </ul>
WIWI-M0119 Modul: Requirements Engineering und Management 2	

Vorlesung: Requirements Engineering und Management 2 (3 Credits)			
Name im Diploma Supplement	Requirements Engineering and Management 2		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl Dr. Marian Daun		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b>			
Ziele und Verantwortlichkeiten des Requirements Engineering und Management im Entwicklungsprozess von softwareintensiven Systemen, Requirements Engineering Rahmenwerk, Konzeptuelle Modellierung und Techniken zur Modellierung von Anforderungen in der statisch-strukturellen Perspektive, der Funktionsperspektive und der Verhaltensperspektive			
<b>Lehrinhalte</b>			
<ol style="list-style-type: none"> <li>1. Validierung von Anforderungen: Grundlegende Methodiken zur Anforderungvalidierung, vertiefte Kenntnisse ausgewählter Methodiken, z.B. Checklisten, Prototypen</li> <li>2. Essentielle Anforderungsmodelle: Theorie essentieller Systemanforderungen; Essenz und Inkarnation von Systemen; Vorgehensweise; Vorteile essentieller Modelle</li> <li>3. Zielorientiertes RE: Dokumentation von Zielen durch: Und/Oder Bäume, Featuremodelle; Methodische Ansätze für die zielorientierte Anforderungserhebung: i*; KAOS</li> <li>4. Szenariobasiertes RE: Typen von Szenarien; natürlichsprachlich formulierte Szenarien, Use Cases, Sequenzdiagramme, u.a.; Einsatz von Szenarien; Szenarien als mittlere Abstraktion; Interrelation von Zielen und Szenarien im Requirements Engineering</li> <li>5. Anforderungsmanagement: Unterscheidung zwischen Kunden-, Produkt-, Projekt-Anforderungsmanagement; Versions- und Konfigurationsverwaltung von Anforderungen; Status, Priorisierung von Anforderungen; Risikomanagement; Meta-Modellierung; Change Management; Kategorisierung und Strukturierung von Anforderungen; Nutzen von Spezifikationsstandards</li> </ol>			
<b>Literaturangaben</b>			
<ul style="list-style-type: none"> <li>• J. Carroll: The Scenario Perspective on System Development, Scenario-Based Design – Envisioning Work and Technology in System Development; John Wiley &amp; Sons, 1995</li> <li>• S.M. McMenamin, J.F. Palmer: Strukturierte Systemanalyse; Carl Hanser &amp; Prentice-Hall International, 1984</li> <li>• K. Pohl: Requirements Engineering, Grundlagen, Prinzipien, Techniken; dpunkt.verlag; 2. Auflage, 2008</li> <li>• B. Schienmann: Kontinuierliches Anforderungsmanagement; Addison Wesley, 2002</li> <li>• K.E. Wiegers: Software Requirements; Microsoft Press, 1999</li> </ul>			
WIWI-C0345 Vorlesung: Requirements Engineering und Management 2 im Modul WIWI-M0119: Requirements Engineering und Management 2			

Übung: Requirements Engineering und Management 2 (3 Credits)			
Name im Diploma Supplement	Requirements Engineering and Management 2		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl Dr. Marian Daun wissenschaftliche Mitarbeiter*innen		
SWS	2	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen siehe Vorlesung			
Lehrinhalte Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung sowie praktische Übungen unter Einsatz von kommerziellen Werkzeugumgebungen.			
Literaturangaben siehe Vorlesungsbeschreibung			
WIWI-C0344 Übung: Requirements Engineering und Management 2 im Modul WIWI-M0119: Requirements Engineering und Management 2			



Modul: Secure Software Systems (6 Credits)	
Name im Diploma Supplement	Secure Software Systems
Verantwortlich	Jun.-Prof. Dr. Lucas Davi
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 60 Stunden</li> <li>• Vorbereitung, Nachbereitung: 75 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen die wichtigsten Klassen von Angriffstechniken und Abwehrmethoden im Bereich der Softwaresicherheit von der Applikationsebene bis zum Betriebssystem.</li> <li>• besitzen fundierte Kenntnisse in der Entwicklung von Angriffstechniken auf Softwaresysteme.</li> <li>• sind in der Lage, eigenständig Proof-of-Concept-Angriffe auf Softwaresysteme zu entwickeln.</li> <li>• können konkrete Verfahren zur Härtung von Softwaresystemen gegen fortgeschrittene Softwareangriffe anwenden.</li> <li>• kennen hardware-basierte Verfahren zur Durchsetzung von Softwaresicherheit.</li> <li>• beherrschen die Konzepte von Softwarebasierten Angriffstechniken und Abwehrmethoden auf verschiedenen Rechnerplattformen.</li> <li>• kennen die aktuelle Forschung und Problemstellungen bezüglich der Entwicklung von sicheren Softwaresystemen.</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90-120 Minuten) oder mündlichen Prüfung (in der Regel: 20-40 Minuten); die konkrete Prüfungsform (Klausur oder mündliche Prüfung) wird in der ersten Woche der Vorlesungszeit von dem zuständigen Dozenten festgelegt.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Network Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;weitere Informatik-Module &gt;1.-4. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Secure Software Systems (3 Credits)</li> <li>• Übung: Secure Software Systems (3 Credits)</li> </ul>
WIWI-M0786 Modul: Secure Software Systems	

Vorlesung: Secure Software Systems (3 Credits)			
Name im Diploma Supplement	Secure Software Systems		
Anbieter	Juniorprofessur für Sichere Software Systeme <a href="https://www.syssec.wiwi.uni-due.de/">https://www.syssec.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Lucas Davi		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundlegende Kenntnisse in Programmierung und Software Engineering			
<b>Abstract</b> In dieser Vorlesung erhalten die Studenten einen Überblick über aktuelle Forschung, Angriffstechniken und Abwehrmethoden im Bereich der Software- und Systemsicherheit. Es werden Sicherheitsprobleme und Schutztechnologien auf Applikations- und Betriebssysteme für unterschiedliche Rechnerarchitekturen (Desktop PCs, mobile und eingebettete Systeme) analysiert. Ein besonderer Fokus dieser Vorlesung ist die Verwundbarkeit von Softwaresystemen gegenüber Laufzeitangriffen (Exploits). Ziel der Vorlesung ist sowohl das Verständnis von modernen, praktischen Angriffstechniken gegen Softwaresysteme als auch die Entwicklung und Anwendung von Sicherheitstechnologien für Softwaresysteme.			
<b>Lehrinhalte</b> <ol style="list-style-type: none"> <li>1. Konventionelle und fortgeschrittene Software Exploittechniken (Buffer Overflow, Return-Oriented Programming)</li> <li>2. Entwicklung von Sicherheitstechnologien zur Detektion und Prävention von Software Exploits (Programmfluss-Integrität, Speicherrandomisierung)</li> <li>3. Software Fault Isolation und Application Sandboxing</li> <li>4. Betriebssystemsicherheit und Zugriffsmodelle mit praktischen Beispielen anhand von Sicherheitsarchitekturen in Multics, Android und Windows</li> <li>5. Trusted Computing Konzepte</li> <li>6. Hardware-basierte Konzepte zur Unterstützung von Softwaresicherheit</li> </ol>			
<b>Literaturangaben</b> <ul style="list-style-type: none"> <li>• T. Jaeger: Operating System Security, Morgan &amp; Claypool, 2008</li> <li>• C. Anley, J. Heasman, F. Lindner, G. Richarte: The Shellcoder's Handbook: Discovering and Exploiting Security Holes, Wiley, 2007</li> <li>• L. Davi: Building Secure Defenses Against Code-Reuse Attacks, Springer, 2015</li> <li>• R. Anderson. Security Engineering: A Guide to Building Dependable Distributed Systems, Wiley, 2008</li> <li>• Aktuelle wissenschaftliche Publikationen von einschlägigen Sicherheitstagungen (werden in der Vorlesung bekannt gegeben)</li> </ul>			
WIWI-C1019 Vorlesung: Secure Software Systems im Modul WIWI-M0786: Secure Software Systems			

Übung: Secure Software Systems (3 Credits)			
Name im Diploma Supplement	Secure Software Systems		
Anbieter	Juniorprofessur für Sichere Software Systeme <a href="https://www.syssec.wiwi.uni-due.de/">https://www.syssec.wiwi.uni-due.de/</a>		
Lehrperson	Jun.-Prof. Dr. Lucas Davi		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> siehe Vorlesung			
<b>Abstract</b> Es werden sowohl praktische als auch theoretische Übungen durchgeführt. In den praktischen Übungen werden die Teilnehmer am Beispiel von verwundbaren Softwaresystemen die Anwendung von Exploittechniken kennenlernen. Zum Beispiel werden die Teilnehmer Proof-of-Concept Exploits auf mobilen Android Systemen selbstständig entwickeln und die Anwendung und Konfiguration von Sicherheitstechnologien zur Detektion dieser Angriffe kennenlernen. Die theoretischen Übungen beinhalten vertiefende Aufgaben zum Stoff der Vorlesung und Analysen von aktuellen wissenschaftlichen Publikationen im Bereich der Softwaresicherheit.			
<b>Lehrinhalte</b> siehe Vorlesung			
<b>Literaturangaben</b> siehe Vorlesung			
WIWI-C1020 Übung: Secure Software Systems im Modul WIWI-M0786: Secure Software Systems			

<b>Modul (auslaufend): Software-Engineering für mobile Systeme (6 Credits)</b>	
<b>Wichtige Änderungen im Modul</b>	Modul wird letztmalig im Sommersemester 2020 angeboten. Ein Ersatzangebot ab Sommersemester 2021 im Bereich Machine Learning ist geplant.
<b>Name im Diploma Supplement</b>	Software Engineering for Mobile Systems
<b>Verantwortlich</b>	Prof. Dr. Volker Gruhn
<b>Voraussetzungen</b>	Siehe Prüfungsordnung.
<b>Workload</b>	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
<b>Dauer</b>	Das Modul erstreckt sich über 1 Semester.
<b>Qualifikationsziele</b>	Die Studierenden <ul style="list-style-type: none"> <li>• besitzen Kenntnis der Besonderheiten verteilter und mobiler Anwendungen im Vergleich zum klassischen Personal-Computer</li> <li>• besitzen ein ganzheitliches Verständnis der Aktivitäten, Rollen und Risiken bei Erstellung, - Betrieb, - Wartung mobiler Anwendungen</li> <li>• können selbstständig mobile Anwendungen erstellen, betreiben, vertreiben und warten</li> <li>• analysieren und bewerten die Unterschiede und Besonderheiten der SW-Entwicklung für mobile Systeme</li> <li>• erstellen selbstständig mobile Softwaresysteme zur Lösung von konkreten Problemen auf verschiedenen SW-Plattformen</li> </ul>
<b>Prüfungsmodalitäten</b>	Zum Modul erfolgt eine modulbezogene Prüfung <ul style="list-style-type: none"> <li>• in der Gestalt einer Klausur (in der Regel: 60-90 Minuten) oder</li> <li>• in der Gestalt einer mündlichen Prüfung (in der Regel: 20-40 Minuten).</li> </ul> <p>Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Bearbeitung von Projektaufgaben als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Die bzw. der Dozierende nimmt zu Beginn der Lehrveranstaltung die Präzisierung von Art und Umfang der Prüfungsvorleistungen vor. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen des Semesters, in dem die Prüfungsvorleistung erworben wurde, sowie für das darauffolgende Sommer- und Wintersemester.</p>
<b>Verwendung in Studiengängen</b>	<ul style="list-style-type: none"> <li>• AI-SE Bachelor 2017&gt;Vertiefungsstudium &gt;Wahlpflichtbereich I: Informatik &gt;5.-6. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• WiInf Bachelor 2010-V2013&gt;Vertiefungsstudium &gt;Wahlpflichtbereich &gt;Vertiefungsrichtung "Modellierung und Realisierung betrieblicher Informationssysteme" &gt;5.-6. Fachsemester, Wahlpflicht</li> </ul>
<b>Bestandteile</b>	<ul style="list-style-type: none"> <li>• Vorlesung mit integrierter Übung: Software-Engineering für mobile Systeme (6 Credits)</li> </ul>
<b>WIWI-M0410 Modul: Software-Engineering für mobile Systeme</b>	

**Vorlesung mit integrierter Übung: Software-Engineering für mobile Systeme (6 Credits)**

Name im Diploma Supplement	Software Engineering for Mobile Systems		
Anbieter	Lehrstuhl für Software-Engineering, insb. mobile Anwendungen <a href="http://www.se.wiwi.uni-due.de/">http://www.se.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Volker Gruhn		
SWS	4	Sprache	deutsch
Turnus	Sommersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Programmierung in Java, Software-Entwicklungswerkzeuge Für dieses Modul werden Programmierungskennnisse vorausgesetzt.			
<b>Lehrinhalte</b> Die Vorlesung vermittelt sowohl einen allgemeinen Überblick über das Thema Software-Engineering für mobile Systeme betrachtet dabei insbesondere die Grundlagen der Entwicklung mobiler Systeme, Grundlagen der Programmierung mobiler Benutzerschnittstellen („Touch“), Grundlagen plattformspezifischer Konzepte, Backend-basierte Anwendungen / Kommunikation mit Services, plattformspezifische Vertiefung der Programmierung und Cross-Plattform Strategien und Technologien. Neben diesen grundlegenden Konzepten und Prinzipien, werden ausgewählte mobile Plattformen vorgestellt. Die folgenden Themen werden in der Vorlesung unter anderem behandelt:			
<ul style="list-style-type: none"> <li>• Mobile Software Development &amp; Konzepte Programmiersprachen: Vermittlung von Grundlagen des Mobile Software Development und Einführung in plattformspezifische Programmiersprachen &amp; Paradigmen.</li> <li>• Konzepte Anwendungsentwicklung &amp; User Interface: Einführung grundlegender Plattformkonzepte auf Ebene der Anwendungsarchitektur und User Interaction sowie Erstellung und Anbindung des User Interface.</li> <li>• Kommunikation &amp; Backend-Anbindung: Einführung und Diskussion von Strategien zur Anbindung an Server und Webservices: Client/Server Kommunikation, herunterladen von Daten, Interpretation von Daten.</li> <li>• Integration von Hardware (intern, extern) &amp; Bibliotheken: Zugriff auf geräteinterne sowie externe Hardware (GPS, Kompass, RFID, Barcode-Scanner), Standard-APIs und deren Verwendung und Umgang mit Fehlersituationen.</li> <li>• Deployment &amp; Vertriebskanäle: Platzierung von Endkundenanwendungen in App-Stores.</li> <li>• Plattformspezifische Vertiefung: Besonderheiten einzelner Plattformen werden detailliert Diskutiert und praxisnah demonstriert.</li> <li>• Cross-Plattform Development: Einführung in die State-of-the-Art des Cross-Plattform-Development: Technologien und Trends</li> </ul>			
<b>Literaturangaben</b>			
<ul style="list-style-type: none"> <li>• <a href="https://developer.apple.com/library/ios/">developer.apple.com/library/ios/</a></li> <li>• <a href="https://developer.android.com/guide/">developer.android.com/guide/</a></li> <li>• <a href="https://msdn.microsoft.com/en-us/library/ff402535%28v=VS.92%29.aspx">msdn.microsoft.com/en-us/library/ff402535%28v=VS.92%29.aspx</a></li> </ul>			
WIWI-C0591 Vorlesung mit integrierter Übung: Software-Engineering für mobile Systeme im Modul WIWI-M0410: Software-Engineering für mobile Systeme			

<b>Modul: Software-Qualitätssicherung (6 Credits)</b>	
<b>Name im Diploma Supplement</b>	Software Quality Assurance
<b>Verantwortlich</b>	Prof. Dr. Klaus Pohl
<b>Voraussetzungen</b>	Siehe Prüfungsordnung.
<b>Workload</b>	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
<b>Dauer</b>	Das Modul erstreckt sich über 1 Semester.
<b>Qualifikationsziele</b>	Die Studierenden <ul style="list-style-type: none"> <li>• kennen die wichtigsten Klassen von Verfahren zur Software-Qualitätssicherung und zum Software-Qualitätsmanagement</li> <li>• kennen die grundlegenden Konzepte in den Themengebieten Softwaretest und Inspektionen</li> <li>• kennen generelle Ansätze zur Software-Messung und Software-Maßen</li> <li>• kennen konkrete Techniken für den Softwaretest, inklusive spezifikationsbasiertem, quellcodebasiertem, modellbasiertem, objektorientiertem und risikobasiertem Test</li> <li>• können konkrete Techniken zur Software-Qualitätssicherung (insbesondere für den Software-Test und für Inspektionen) praktisch anwenden</li> <li>• sind in der Lage, eigenständig eine Inspektion vorzunehmen</li> <li>• sind in der Lage, eine begründete Auswahl von Qualitätssicherungstechniken vorzunehmen (z.B. quellcodebasiertes Testen vs. Spezifikationsbasiertes Testen)</li> <li>• können Techniken zur Software-Messung anwenden und Software-Maße zielgerichtet auswählen</li> </ul>
<b>Prüfungsmodalitäten</b>	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer Klausur (in der Regel: 90 bis 120 Minuten). Vom Dozierenden wird zu Beginn der Veranstaltung festgelegt, ob die erfolgreiche Teilnahme an der Übung als Prüfungsvorleistung Zulassungsvoraussetzung zur Modulprüfung ist. Bestandene Prüfungsvorleistungen haben nur Gültigkeit für die Prüfungen, die zu der Veranstaltung im jeweiligen Semester gehören.
<b>Verwendung in Studiengängen</b>	<ul style="list-style-type: none"> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Network Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• AI-SE Master 2010&gt;Bereich 1 (Profilbereich) &gt;Software Systems Engineering &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• Mathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• SNE Master 2016&gt;Wahlpflichtbereich &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• TechMathe Master 2013&gt;Anwendungsfach "Informatik" &gt;Profil "Software Systems Engineering" &gt;1.-3. Fachsemester, Wahlpflicht</li> <li>• WiInf Master 2010&gt;Wahlpflichtbereich &gt;Wahlpflichtbereich II: Informatik, BWL, VWL &gt;Wahlpflichtmodule der Informatik &gt;1.-3. Fachsemester, Wahlpflicht</li> </ul>
<b>Bestandteile</b>	<ul style="list-style-type: none"> <li>• Vorlesung: Software-Qualitätssicherung (3 Credits)</li> <li>• Übung: Software-Qualitätssicherung (3 Credits)</li> </ul>
WIWI-M0068 Modul: Software-Qualitätssicherung	

Vorlesung: Software-Qualitätssicherung (3 Credits)			
Name im Diploma Supplement	Software Quality Assurance		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl Dr. Andreas Metzger		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> Grundkenntnisse in der (objektorientierten) Programmierung und der Modellierung von Software-Systemen			
<b>Lehrinhalte</b>			
<ol style="list-style-type: none"> <li>1. Einführung: Motivation, Begriff der Software-Qualität, Definitionen, konstruktive und analytische Qualitätssicherung, Übersicht über Verfahren (statisch, dynamisch, formale Techniken)</li> <li>2. Standards für die Qualitätssicherung in verschiedenen Bereichen (z.B. IEEE 829-1998, DO-178B)</li> <li>3. Überblick über Verfahren zum dynamischen Test, Diskussion der Vor- und Nachteile und Bewertung der praktischen Relevanz</li> <li>4. Fortgeschrittene Techniken für den dynamischen Test, z.B. zustandsbasierter Test, Ursache-Wirkungs-Analyse, datenflussorientierter Test, Regressionstest, diversifizierender Test</li> <li>5. Objektorientiertes Testen: Klassentest, Integrationstest, Use-Case-basiertes Testen</li> <li>6. Test service-basierter Systeme: Grundlagen SOA, Testen vs. Monitoring</li> <li>7. Messen und Bewerten von Softwareentwicklungsprozessen: Motivation und Einführung, Messtheorie (u.a. Skalentypen), Vorstellung ausgewählter Maße für Größe, Struktur und Qualität (z.B. McCabe, und neuere Maße für die objektorientierte SW-Entwicklung), zielorientiertes Messen mit der Goal-Question-Metric Methode (GQM)</li> <li>8. Frühzeitige Qualitätssicherung durch statische Verfahren: Vorgehensweisen (Audits, Walkthroughs, Inspektionen und Reviews), Vorstellung des Inspektionsprozesses, Lesetechniken für verschiedene Arten von Dokumenten (Anforderungen, Design, Kode)</li> </ol>			
<b>Literaturangaben</b>			
<ul style="list-style-type: none"> <li>• P. Liggesmeyer: Software-Qualität – Testen, Analysieren und Verifizieren von Software; Spektrum Verlag, 2002</li> <li>• H.M. Sneed, M. Winter: Testen objektorientierter Software; Hanser, 2002</li> <li>• R.V. Binder: Testing Object-oriented Systems; Addison-Wesley, 1999</li> <li>• L. Baresi, E. Di Nitto: Test and Analysis of Web Services, Springer, 2007</li> <li>• N.E. Fenton, S.L. Pfleeger: Software Metrics – A Rigorous Practical Approach, International Thomson Computer Press, 1996</li> <li>• R. van Solingen, E. Berghout: The Goal/Question/Metric Method – A Practical Guide for Quality Improvement of Software Development; McGraw-Hill, 1999</li> </ul>			
WIWI-C0336 <b>Vorlesung: Software-Qualitätssicherung</b> im Modul WIWI-M0068: Software-Qualitätssicherung			

Übung: Software-Qualitätssicherung (3 Credits)			
Name im Diploma Supplement	Software Quality Assurance		
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>		
Lehrperson	Prof. Dr. Klaus Pohl Dr. Andreas Metzger		
SWS	2	Sprache	deutsch
Turnus	Wintersemester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> siehe Vorlesung			
<b>Lehrinhalte</b> Vertiefende Aufgaben und Beispiele zum Stoff der Vorlesung.			
<b>Literaturangaben</b> siehe Vorlesungsbeschreibung			
WIWI-C0335 <b>Übung: Software-Qualitätssicherung</b> im Modul WIWI-M0068: Software-Qualitätssicherung			

## **Pflichtbereich Didaktik der Informatik - 1. Fachsemester, Pflicht**

<b>Modul: Didaktik der Informatik II (6 Credits)</b>	
Name im Diploma Supplement	Teaching Informatics II
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	180 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen ausgewählte Forschungsergebnisse zu den Schwerpunkten der Lehrveranstaltung und können diese begründet auf dem Prozess der Planung und Analyse schulbezogener Lehr-Lern-Situationen und -einheiten in der Informatik beziehen,</li> <li>• kennen ausgewählte Forschungskonzeptionen und -methoden im Bereich der Didaktik der Informatik und können diese im Kleinen auf die Gestaltung forschungsbezogener Erkundungen im Praxisfeld transferieren.</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> </ul>
Praxisrelevanz	Im Rahmen eigener lebensbegleitender Lernprozesse müssen sich Lehrkräfte der Informatik immer wieder mit neuen Konzepten und Erkenntnissen aus der Fachwissenschaft und Fachdidaktik auseinandersetzen, um zu bewerten, ob und wie ihr Unterricht dadurch weiterentwickelt werden kann. Durch die aktive Auseinandersetzung mit solchen Konzepten und deren Bezug auf den Unterricht wird dafür eine wesentliche Grundlage gelegt.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer mündlichen Prüfung (in der Regel 20-40 Minuten).
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Pflichtbereich Didaktik der Informatik &gt;1. Fachsemester, Pflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Vorlesung: Didaktik der Informatik II (3 Credits)</li> <li>• Übung: Didaktik der Informatik II (3 Credits)</li> </ul>
WIWI-M0701 Modul: Didaktik der Informatik II	



Vorlesung: Didaktik der Informatik II (3 Credits)			
Name im Diploma Supplement	Teaching informatics II		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	2	Sprache	deutsch
Turnus	jedes Semester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul>			
<b>Abstract</b> Im Rahmen der Veranstaltung wird der in den Grundlagenveranstaltungen angelegte Prozess der Planung schulbezogener Lehr-Lern-Situationen und -Einheiten in der Informatik mit theoretisch fundierten Konzepten, Methoden und Ergebnissen zur Vermittlung von Informatikkonzepten, lernerbezogenen Aspekten und vertiefenden unterrichtsmethodischen Konzeptionen des Informatikunterrichts untersetzt. Der Schwerpunkt liegt hierbei auf informatikdidaktischen Forschungsarbeiten, so dass deren Konzeption, Strukturierung und Ergebnisse auch Gegenstand sind mit dem Ziel, Studierenden verstärkt eine Forschungsperspektive auf die Informatikdidaktik zu öffnen.			
<b>Qualifikationsziele</b> Die Studierenden <ul style="list-style-type: none"> <li>• kennen ausgewählte Forschungsergebnisse zu den Schwerpunkten der Lehrveranstaltung und können diese begründet auf dem Prozess der Planung und Analyse schulbezogener Lehr-Lern-Situationen und -einheiten in der Informatik beziehen,</li> <li>• kennen ausgewählte Forschungskonzeptionen und –methoden im Bereich der Didaktik der Informatik und können diese im Kleinen auf die Gestaltung forschungsbezogener Erkundungen im Praxisfeld transferieren.</li> </ul>			
<b>Lehrinhalte</b> Gegenstand der Lehrveranstaltung ist eine semesterweise Auswahl aus den u.g. Themenfeldern: Konzepte, Methoden und Ergebnisse zur Vermittlung von Informatikkonzepten, u. a. <ul style="list-style-type: none"> <li>• Programmieren lehren und lernen</li> <li>• Rekursion vermitteln</li> <li>• Objects first vs. Objects later</li> <li>• Vermittlung von Informatikkonzepten durch die Entwicklung von Spielen</li> <li>• Netzwerk- und Hardwarekonzepte</li> </ul> Lernerbezogene Aspekte des Informatikunterrichts <ul style="list-style-type: none"> <li>• Schülervorstellungen in der Informatik</li> <li>• Kompetenzmodellierung im Bereich der Informatik</li> <li>• Inklusion in der informatischen Bildung</li> </ul> Methodische Gestaltung des Informatikunterrichts <ul style="list-style-type: none"> <li>• Kreativität im Informatikunterricht</li> <li>• Gestaltung von Unterrichts- und Prüfungsaufgaben im Bereich der Informatik</li> </ul>			
<b>Literaturangaben</b> Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.			
<b>didaktisches Konzept</b> Die Veranstaltung folgt zu großen Teilen einem Inverse-Classroom-Konzept. Es steht ein umfangreiches Folienskript zur Verfügung, anhand dessen sich die Studierenden auf die jeweiligen Präsenztermine vorbereiten. Bei diesen erfolgt jeweils eine Zusammenfassung bzw. weiterführende Erläuterung durch den Dozenten. Anschließend wird in Einzel-, Gruppen- oder Plenumsarbeit anhand von vorbereiteten Aufgaben der Lehrveranstaltungsstoff angewandt und vertieft, weiterhin verteilt erarbeitete Ergebnisse im Plenum präsentiert und diskutiert. Die Bearbeitung der Aufgaben bezieht als Gegenstand und Medium verschiedene unterrichtsgeeignete Hard- und Software mit ein, so dass die Studierenden mit der LV auch ein Modell für eigengestaltete Lehr-Lern-Szenarien erhalten.			
WIWI-C0862 Vorlesung: Didaktik der Informatik II im Modul WIWI-M0701: Didaktik der Informatik II			

**Übung: Didaktik der Informatik II (3 Credits)**

Name im Diploma Supplement	Teaching informatics II		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Wissenschaftliche/r Mitarbeiter/in oder Lehrbeauftragte/r des Lehrstuhls für Didaktik der Informatik		
SWS	2	Sprache	deutsch
Turnus	jedes Semester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b> siehe Vorlesung			
<b>Lehrinhalte</b> Die Gliederung der Übung orientiert sich an der Gliederung der zugeordneten Vorlesung.			
<b>Literaturangaben</b> Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.			
WIWI-C0863 <b>Übung: Didaktik der Informatik II</b> im Modul WIWI-M0701: Didaktik der Informatik II			

## Wahlpflichtbereich Didaktik der Informatik - 2.-3. Fachsemester, Pflicht

Praxissemesterteilnehmende können eine Prüfungsleistung im Umfang von 2 Credits erbringen, der Regelfall sind 3 Credits. Anerkennungsfähig sind ebenfalls die Lehrveranstaltungen aus dem Bereich E-Learning ([Prof. Dr. H. U. Hoppe, Duisburg](#)). Für weitere Informationen kontaktieren Sie bitte die jeweiligen Dozenten.

Modul: Aktuelle Beiträge zur Didaktik der Informatik (3 Credits)	
Name im Diploma Supplement	Current Contributions to Informatics Education Research
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 23 Stunden</li> <li>• Vorbereitung, Nachbereitung: 22 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• diskutieren ausgewählte Forschungsansätze und -ergebnisse im Bereich der Didaktik der Informatik im Hinblick auf die Schulpraxis</li> <li>• analysieren und bewerten Forschungsmethoden im Bereich Didaktik der Informatik</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> <li>• analysieren Fachkonzepte im Hinblick auf ihren sprachlichen Anforderungen</li> <li>• kennen Konzepte zu Verbindung von sprachlichem und fachlichem Lernen im Informatikunterricht</li> <li>• entwickeln eigene Ansätze für einen inklusiven Informatikunterricht in sprachlich heterogenen Klassen</li> </ul>
Praxisrelevanz	Es gehört zu den Aufgaben einer Informatik-Lehrkraft, wissenschaftliche Entwicklungen im Bereich der Fachwissenschaft, der Fachdidaktik und der Bildungswissenschaften zu beobachten, deren unterrichtliche Relevanz zu bewerten und zum Theorie-Praxis-Transfer beizutragen.
Prüfungsmodalitäten	Zum Modul "Aktuelle Beiträge zur Didaktik der Informatik" erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt: <ul style="list-style-type: none"> <li>• Entweder Seminararbeit (in der Regel 10-15 DIN A4-Seiten) oder Portfolio mit Lerneinheit(en) für den Informatikunterricht; die konkrete Prüfungsform (Seminararbeit oder Portfolio) wird zu Beginn der Veranstaltung festgelegt (60% der Gesamtnote)</li> <li>• Seminarvortrag (in der Regel: 30 Minuten) und anschl. Diskussion im LV-Plenum (40% der Gesamtnote)</li> </ul> Die Prüfung in diesem Modul darf nicht abgelegt werden, wenn "Aktuelle Forschungsthemen der Didaktik der Informatik" bereits bestanden ist.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Didaktik der Informatik &gt;2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Aktuelle Beiträge zur Didaktik der Informatik (3 Credits)</li> </ul>
WIWI-M0692 Modul: Aktuelle Beiträge zur Didaktik der Informatik	

Seminar: Aktuelle Beiträge zur Didaktik der Informatik (3 Credits)			
Name im Diploma Supplement	Current Contributions to Informatics Education Research		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörerschaft	unbeschränkt
<b>Erläuterung zum unregelmäßigen Turnus</b> Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.			
<b>empfohlenes Vorwissen</b> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul>			
<b>Lehrinhalte</b> Die Lehrinhalte und Seminarthemen werden semesteraktuell basierend auf aktuellen Forschungspublikationen im Bereich der informatischen Bildung an Schulen und Hochschulen gewählt und vergeben, Schwerpunkte können z. B. sein: <ul style="list-style-type: none"> <li>• Internationale Bildungskonzeptionen im Bereich der Informatik</li> <li>• Vermittlungsmethoden im Bereich der Informatik</li> <li>• Medien und Werkzeuge für die informatische Bildung</li> <li>• Kompetenzmodellierung im Bereich der Informatik</li> <li>• Empirische Untersuchungen im Bereich der informatischen Bildung</li> <li>• Inklusion in der informatischen Bildung</li> <li>• Einführung in die Grundlagen der Sprachbildung im Fach Informatik, u.a. Erarbeitung von Leitlinien für sprachaufmerksamen Fachunterricht</li> <li>• Sach- und zielgruppengerechte didaktische Aufbereitung von informatischen Inhalten unter besonderer Berücksichtigung der Sprachbildung</li> <li>• Analyse der gesprochenen und geschriebenen Sprache im Informatikunterricht anhand des Genre-Cycles</li> </ul>			
<b>Literaturangaben</b> Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.			
WIWI-C0853 Seminar: Aktuelle Beiträge zur Didaktik der Informatik im Modul WIWI-M0692: Aktuelle Beiträge zur Didaktik der Informatik			

Modul: Informatik in der Sekundarstufe I (3 Credits)	
Name im Diploma Supplement	Teaching Informatics in Lower Secondary Schools
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 23 Stunden</li> <li>• Vorbereitung, Nachbereitung: 22 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen didaktische Ansätze und Unterrichtskonzeptionen für den Informatikunterricht in der Sek. I</li> <li>• verwenden Unterrichtshilfen für den Informatikunterricht in der Sek. I sachgerecht</li> <li>• gestalten kompetenzorientierte informatische Lehr-Lern-Situationen für die Sek. I unter Verwendung adäquater Unterrichtsmethoden</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> </ul>
Praxisrelevanz	Das Modul wählt einen praxisorientierten Zugang zu Unterrichtsansätzen und -mitteln für die Sekundarstufe I
Prüfungsmodalitäten	Zum Modul „Informatikunterricht in der Sekundarstufe I“ erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt: <ul style="list-style-type: none"> <li>• Seminararbeit im Umfang von 10-15 DIN A4-Seiten (60% der Gesamtnote)</li> <li>• Seminarvortrag (in der Regel: 30 Minuten) und anschl. Diskussion im LV-Plenum (40% der Gesamtnote)</li> </ul>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Didaktik der Informatik &gt;2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Informatik in der Sekundarstufe I (3 Credits)</li> </ul>
WIWI-M0690 Modul: Informatik in der Sekundarstufe I	

Seminar: Informatik in der Sekundarstufe I (3 Credits)			
Name im Diploma Supplement	Teaching Informatics in Lower Secondary Schools		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<p><b>Erläuterung zum unregelmäßigen Turnus</b> Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.</p>			
<p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Detaillierte Planung eines Unterrichtsszenarios unter Einbezug allgemein- und fachdidaktischer Theorien sowie bildungspolitischer Legitimation</li> <li>• Sach- und zielgruppengerechte didaktische Aufbereitung von informatischen Inhalten für eine Lerngruppe</li> <li>• Erstellung von zielführendem und differenzierenden Lernmaterial</li> <li>• Erprobung eines Unterrichtsausschnittes mit den Seminarteilnehmenden</li> </ul>			
<p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
WIWI-C0851 Seminar: Informatik in der Sekundarstufe I im Modul WIWI-M0690: Informatik in der Sekundarstufe I			

Modul: Informatiksysteme aus fachdidaktischer Sicht (3 Credits)	
Name im Diploma Supplement	Taking a Pedagogical Perspective on Informatics Systems
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 23 Stunden</li> <li>• Vorbereitung, Nachbereitung: 22 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen unterrichtsrelevante Informatiksysteme</li> <li>• wissen, wie sie unterrichtsrelevante Informatiksysteme in die Konzeption von Unterricht sachgerecht einbeziehen</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> </ul>
Praxisrelevanz	Softwarebasierte Werkzeuge und Lernhilfen gehören zum beruflichen Handwerkzeug einer jeden Informatik-Lehrkraft.
Prüfungsmodalitäten	Zum Modul „Informatiksysteme aus fachdidaktischer Sicht“ erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt: <ul style="list-style-type: none"> <li>• Seminararbeit im Umfang von 10-15 DIN A4-Seiten (60% der Gesamtnote)</li> <li>• Seminarvortrag (in der Regel: 30 Minuten) und anschl. Diskussion im LV-Plenum (40% der Gesamtnote)</li> </ul>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014 &gt; Wahlpflichtbereich Didaktik der Informatik &gt; 2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Informatiksysteme aus fachdidaktischer Sicht (3 Credits)</li> </ul>
WIWI-M0714 Modul: Informatiksysteme aus fachdidaktischer Sicht	

Seminar: Informatiksysteme aus fachdidaktischer Sicht (3 Credits)			
Name im Diploma Supplement	Taking a Pedagogical Perspective on Informatics Systems		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda Wissenschaftliche/r Mitarbeiter/in oder Lehrbeauftragte/r des Lehrstuhls für Didaktik der Informatik		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<p><b>Erläuterung zum unregelmäßigen Turnus</b> Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.</p>			
<p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und –architektur, Software Engineering</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul>			
<p><b>Abstract</b></p> <p>Ein wichtiger Teil der Tätigkeit einer Informatiklehrperson ist der Umgang mit verschiedensten Informatiksystemen. Dabei werden Informatiklehrkräfte häufig auch in Entscheidungs- und Beschaffungsprozesse solcher unterrichtsrelevanter Systeme mit einbezogen, so dass ein grundlegendes Wissen über verschiedene Systeme und Arten von Systemen von elementarer Bedeutung ist. Aus diesem Grund sollen die Studierenden in diesem Seminar grundlegende Konzepte didaktischer Informatiksysteme kennenlernen und ausgewählte Systeme auch selbst im unterrichtlichen Einsatz erproben.</p>			
<p><b>Qualifikationsziele</b></p> <p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• kennen unterrichtsrelevante Informatiksysteme</li> <li>• wissen, wie sie unterrichtsrelevante Informatiksysteme in die Konzeption von Unterricht sachgerecht mit einbeziehen</li> </ul>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Lehren und Lernen in der Informatik mit digitalen Medien</li> <li>• Gestaltung von Blended Learning Szenarien</li> </ul>			
<p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
WIWI-C0924 Seminar: Informatiksysteme aus fachdidaktischer Sicht im Modul WIWI-M0714: Informatiksysteme aus fachdidaktischer Sicht			

Modul: Methodeneinsatz im Informatikunterricht (3 Credits)	
Name im Diploma Supplement	Teaching Methods in Informatics Education
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 23 Stunden</li> <li>• Vorbereitung, Nachbereitung: 22 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen für den Informatikunterricht geeignete Unterrichtsmethoden</li> <li>• können für ein Vermittlungsziel geeignete Unterrichtsmethoden auswählen</li> <li>• gestalten informatische Lehr-Lern-Situationen unter Verwendung adäquater Unterrichtsmethoden</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> </ul>
Praxisrelevanz	Unterrichtsmethoden gehören zum beruflichen Handwerkzeug einer jeden Lehrkraft.
Prüfungsmodalitäten	Zum Modul „Methodeneinsatz im Informatikunterricht“ erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt: <ul style="list-style-type: none"> <li>• Seminararbeit im Umfang von 10-15 DIN A4-Seiten (60% der Gesamtnote)</li> <li>• Seminarvortrag (in der Regel: 30 Minuten) und anschl. Diskussion im LV-Plenum (40% der Gesamtnote)</li> </ul>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Didaktik der Informatik &gt;2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Methodeneinsatz im Informatikunterricht (3 Credits)</li> </ul>
WIWI-M0715 Modul: Methodeneinsatz im Informatikunterricht	

Seminar: Methodeneinsatz im Informatikunterricht (3 Credits)			
Name im Diploma Supplement	Teaching Methods in Informatics Education		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda Wissenschaftliche/r Mitarbeiter/in oder Lehrbeauftragte/r des Lehrstuhls für Didaktik der Informatik		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<p><b>Erläuterung zum unregelmäßigen Turnus</b>Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.</p> <p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und –architektur, Software Engineering</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul> <p><b>Abstract</b></p> <p>Von ausgebildeten Lehrkräften wird umfassendes methodisches Wissen verlangt, um damit einen differenzierenden Unterricht in heterogenen Lerngruppen zu gestalten. Dies setzt voraus, dass sich Studierende im Rahmen ihrer Ausbildung umfassend mit den für Informatikunterricht sinnvoll einsetzbaren Methoden auseinandersetzen. Aus diesem Grund liegt der Schwerpunkt dieser Veranstaltung auf der Erarbeitung, Erprobung und vergleichenden Bewertung unterschiedlicher für Informatikunterricht geeigneter Unterrichtsmethoden. Übergeordnetes Ziel soll es sein, die Studierenden in die Lage zu versetzen, die erworbenen Kenntnisse über Unterrichtsmethodiken sachgerecht auszuwählen und einzusetzen.</p> <p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Kennenlernen und Bewerten unterschiedlicher Unterrichtsmethoden für den Informatikunterricht</li> <li>• Einarbeitung in exemplarische Unterrichtsmethoden</li> <li>• Verknüpfung der untersuchten Methode(n) mit fachwissenschaftlichen Inhalten</li> </ul> <p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
WIWI-C0925 Seminar: Methodeneinsatz im Informatikunterricht im Modul WIWI-M0715: Methodeneinsatz im Informatikunterricht			

Modul: Schülerlabor Informatik (3 Credits)	
Name im Diploma Supplement	Informatics Teaching Lab
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 23 Stunden</li> <li>• Vorbereitung, Nachbereitung: 22 Stunden</li> <li>• Prüfungsvorbereitung: 45 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• entwickeln in Teams zielgruppengerechte Informatik-Lehr-Lern-Einheiten inkl. des erforderlichen Begleitmaterials für schulische und außerschulische Lernorte</li> <li>• führen die geplante Sequenz selbstständig durch und evaluieren sie</li> <li>• haben Kenntnisse in inklusionsorientierten Fragestellungen</li> </ul>
Praxisrelevanz	Die Gestaltung, Durchführung und Evaluation informatischer Lehr-Lern-Einheiten ist für zukünftige Informatiklehrkräfte von hoher praktischer Relevanz.
Prüfungsmodalitäten	Das Seminar ist abgeschlossen, wenn <ul style="list-style-type: none"> <li>• das vollständige Versuchsportfolio einschließlich der Materialien rechtzeitig (Inhalte und Termin werden semesteraktuell bekannt gegeben) abgegeben wurde und</li> <li>• der Versuch mindestens einmal durchgeführt und evaluiert wurde. Dies beinhaltet auch eine Reflexion über den Ablauf des durchgeführten Versuchs.</li> </ul>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Didaktik der Informatik &gt;2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Schülerlabor Informatik (3 Credits)</li> </ul>
WIWI-M0691 Modul: Schülerlabor Informatik	

Seminar: Schülerlabor Informatik (3 Credits)			
Name im Diploma Supplement	Informatics Teaching Lab		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<p><b>Erläuterung zum unregelmäßigen Turnus</b>Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.</p>			
<p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> </ul>			
<p><b>Abstract</b></p> <p>Eine Aufgabe von Informatiklehrkräften ist es, Informatikunterricht vielseitig und zielgruppenangemessen zu gestalten. Dabei finden sich informatische Lehr-Lern-Situationen nicht nur im Schulunterricht, sondern auch an außerschulischen Lernorten, wie einem universitären Lehr-Lern-Labor, z. B. im Rahmen von universitären Angeboten an Schulen. Die Studierenden entwickeln daher für den schulischen oder außerschulischen Einsatz unter didaktischen und fachwissenschaftlichen Blickwinkeln verschiedene motivierende Lehr-Lern-Sequenzen zu selbstgewählten Themen, um sie danach in der Praxis zu evaluieren und bei Bedarf weiterzuentwickeln.</p>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Entwickeln von zielgruppengerechten Informatiksequenzen für schulische und außerschulische Lernorte</li> <li>• Einbettung in die Fachwissenschaft und Fachdidaktik Informatik</li> <li>• Sach- und zielgruppengerechtes Entwickeln von Unterrichts- und Seminarmaterialien</li> <li>• Selbstständige Durchführung und Evaluation der geplanten Sequenz</li> </ul>			
<p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
WIWI-C0852 Seminar: Schülerlabor Informatik im Modul WIWI-M0691: Schülerlabor Informatik			



<b>Modul: Sprachbildung im Informatikunterricht (3 Credits)</b>	
Name im Diploma Supplement	Discipline-Specific Language Learning in Informatics
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 30 Stunden</li> <li>• Vorbereitung, Nachbereitung: 20 Stunden</li> <li>• Prüfungsvorbereitung: 40 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• analysieren Fachkonzepte im Hinblick auf ihre sprachlichen Anforderungen</li> <li>• können Konzepte zu Verbindung von sprachlichem und fachlichem Lernen im Informatikunterricht benennen und erläutern</li> <li>• können informatische Lehr-Lern-Situationen unter Berücksichtigung der Sprachbildung gestalten</li> <li>• entwickeln eigene Ansätze für einen inklusiven Informatikunterricht in sprachlich heterogenen Klassen</li> </ul>
Praxisrelevanz	Unterrichtsmethoden gehören zum beruflichen Handwerkzeug einer jeden Lehrkraft.
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt: <ul style="list-style-type: none"> <li>• Semesterbegleitendes Portfolio mit folgenden Bestandteilen: Dokumentation des eigenen Lernprozesses, Erstellung von Materialien, Bearbeitung von Lernaufgaben und Aufträgen, sowie ein ausgewählter Schwerpunkt, der die fachlichen und sprachlichen Anforderungen unter Anwendung der vermittelten Konzepte und Methoden der Sprachbildung berücksichtigt (80% der Note)</li> <li>• Seminarvortrag (in der Regel 30 Minuten): Präsentation des Portfolio-Schwerpunktes (20% der Note)</li> </ul>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Wahlpflichtbereich Didaktik der Informatik &gt;2.-3. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Sprachbildung im Informatikunterricht (3 Credits)</li> </ul>
WIWI-M0903 Modul: Sprachbildung im Informatikunterricht	

Seminar: Sprachbildung im Informatikunterricht (3 Credits)			
Name im Diploma Supplement	Discipline-Specific Language Learning in Informatics		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda Wissenschaftliche/r Mitarbeiter/in oder Lehrbeauftragte/r des Lehrstuhls für Didaktik der Informatik		
SWS	2	Sprache	deutsch
Turnus	unregelmäßig	maximale Hörschaft	unbeschränkt
<p><b>Erläuterung zum unregelmäßigen Turnus</b> Der Lehrstuhl für Didaktik der Informatik bietet jedes Semester ca. 2 informatikdidaktische Seminare in Rotation an, so dass jedes einzelne Seminar ca. alle 2-3 Semester angeboten wird.</p>			
<p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und –architektur, Software Engineering</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und -Einheiten in der Informatik</li> </ul>			
<p><b>Abstract</b></p> <p>Schwerpunkt der Veranstaltung ist die Thematisierung von Sprachbildung im Fachunterricht Informatik. Am Beispiel verschiedener Informatikthemen werden unterschiedliche Aspekte informatischer Lernumgebungen in sprachbewusster Perspektive thematisiert. Die Studierenden beschäftigen sich mit grundlegenden Spracherwerbsprozessen und machen ihr Wissen um solche Prozesse für das fachliche Lernen nutzbar. Weiterhin lernen sie Konzepte und Strategien für den sprachbildenden Unterricht in Informatik kennen und probieren diese an kleineren Sequenzen aus. Dazu gehört sowohl die Entwicklung von Aufgaben zur Vertiefung prozessbezogener Kompetenzen (z. B. Modellieren, Analysieren oder Beschreiben) als auch die Gestaltung unterschiedlicher Phasen im Informatikunterricht (Einstiege, Systematisierung und Übung) unter besonderer Berücksichtigung der Sprachbildung.</p>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Einführung in die Grundlagen der Sprachbildung im Fach Informatik, u. a. Erarbeitung von Leitlinien für sprachaufmerksamen Fachunterricht</li> <li>• Sach- und zielgruppengerechte didaktische Aufbereitung von informatischen Inhalten unter besonderer Berücksichtigung der Sprachbildung</li> <li>• Analyse der gesprochenen und geschriebenen Sprache im Informatikunterricht anhand des Genre-Cycles</li> </ul>			
<p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
<p><b>didaktisches Konzept</b></p> <p>Das Seminar ist folgendermaßen aufgebaut:</p> <ul style="list-style-type: none"> <li>• Einführung: Was macht das Textverständnis so schwierig?</li> <li>• Analyse von Schulbuchtexten auf sprachliche Besonderheiten</li> <li>• Analyse gesprochener Sprache unter Durchführung eines Informatik-Spiels</li> <li>• Analyse der Operatoren und den dazugehörigen Textsorten im Informatikunterricht</li> <li>• Konzepte zur Textproduktion im Informatikunterricht und Einbindung in den Unterricht</li> <li>• Sprachstandsdiagnose: Vergleich verschiedener Modelle</li> <li>• Leseverständnis: Texte verstehen</li> </ul>			
WIWI-C1156 Seminar: Sprachbildung im Informatikunterricht im Modul WIWI-M0903: Sprachbildung im Informatikunterricht			

## **Begleitmodul zum Praxissemester "Schule und Unterricht forschend verstehen" - 2. Fachsemester, Pflicht**

Während des Praxissemesters sind insgesamt zwei Studienprojekte zu absolvieren. Für das Praxissemester werden in diesem Modulhandbuch nur die Credits und die Modulverantwortlichen für den Bereich der Informatik ausgewiesen.

- Wird das Studienprojekt im Fach Informatik durchgeführt, so werden 5 Credits durch eine Modulteilprüfung erworben.
- Wird im Fach Informatik kein Studienprojekt durchgeführt, so werden durch den Besuch der Begleitveranstaltung 2 Credits erworben.

—

Studienprojekt wird im Fach Informatik durchgeführt:

Modul: Praxissemester (5 Credits)	
Name im Diploma Supplement	Obligatory Long-term School Placement (duration: five months)
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	150 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 90 Stunden</li> <li>• Prüfungsvorbereitung: 15 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• identifizieren praxisbezogene Entwicklungsaufgaben schulformspezifisch</li> <li>• planen kleinere Studien-, Unterrichts- und/oder Forschungsprojekte (auch unter Berücksichtigung der Interessen der Praktikumsschulen)</li> <li>• können dabei wissenschaftliche Inhalte der Bildungswissenschaften und der Unterrichtsfächer auf Situationen und Prozesse schulischer Praxis beziehen</li> <li>• kennen Ziele und Phasen empirischer Forschung und wenden ausgewählte Methoden exemplarisch in den schul- und unterrichtsbezogenen Projekten an</li> <li>• sind befähigt, Lehr-Lernprozesse unter Berücksichtigung individueller, institutioneller und gesellschaftlicher Rahmenbedingungen zu gestalten</li> <li>• reflektieren theoriegeleitet Beobachtungen und Erfahrungen in Schule und Unterricht</li> </ul> <p><i>davon Schlüsselqualifikationen:</i></p> <ul style="list-style-type: none"> <li>• Organisationsfähigkeit, realistische Zeit- und Arbeitsplanung</li> <li>• Planungs-, Projekt- und Innovationsmanagement</li> <li>• Kooperationsfähigkeit</li> <li>• Erschließung, kritische Sichtung und Präsentation von Forschungsergebnissen</li> <li>• Anwendung wissenschaftlicher Methoden und Auswertungsstrategien</li> <li>• konstruktive Wertschätzung von Diversity</li> </ul>
Prüfungsmodalitäten	<p>Benotetes Kolloquium basierend auf den obligatorischen Anteilen des prozessbegleitenden Portfolios ("Portfolio Praxiselemente").</p> <p>Zur Lehrveranstaltung „Begleitung des Praxissemesters“ erfolgt eine modulbezogene Prüfung, die sich auf folgende Prüfungsformen erstreckt:</p> <ul style="list-style-type: none"> <li>• Ausarbeitung zum durchgeführten Studienprojekt im Umfang von 10-15 DIN A4-Seiten</li> <li>• Vortrag (in der Regel: 30 Minuten) und anschl. Diskussion im Plenum</li> </ul> <p><b>Hinweis:</b> Insgesamt sind Begleitveranstaltungen in den Bildungswissenschaften sowie den beiden Fächern/Fachdidaktiken abzuschließen. In diesem Modulhandbuch ist ausschließlich der sich auf das Fach Informatik beziehende Anteil ausgewiesen. Es sind zwei Modulteilprüfungen in den beiden Fächern mit Studienprojekt zu absolvieren. Die beiden Modulteilprüfungen gehen zu gleichen Teilen in die Modulabschlussnote ein.</p>
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Begleitmodul zum Praxissemester "Schule und Unterricht forschend verstehen" &gt;2. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Begleitung des Praxissemesters (5 Credits)</li> </ul>
WIWI-M0146 Modul: Praxissemester	

Seminar: Begleitung des Praxissemesters (5 Credits)			
Name im Diploma Supplement	Teacher Development - Reflective Practice		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	3	Sprache	deutsch
Turnus	jedes Semester	maximale Hörschaft	unbeschränkt
<p><b>empfohlenes Vorwissen</b></p> <ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> <li>• Ausgewählte Forschungskonzeptionen, –methoden und –ergebnisse im Bereich der Didaktik der Informatik</li> </ul>			
<p><b>Lehrinhalte</b></p> <ul style="list-style-type: none"> <li>• Methoden forschenden Lernens</li> <li>• Konzeption, Durchführung und Evaluation von Studien- und Unterrichtsprojekten zu ausgewählten Themen der informatischen Bildung</li> <li>• Beobachtung und Bewertung von Lehr-Lern-Prozessen</li> <li>• Lernerfolgskontrollen</li> <li>• Planung, Durchführung und Evaluation von Informatikunterrichtsausschnitten unter Begleitung</li> </ul>			
<p><b>Literaturangaben</b></p> <p>Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.</p>			
WIWI-C0308 Seminar: Begleitung des Praxissemesters im Modul WIWI-M0146: Praxissemester			

Studienprojekt wird **nicht** im Fach Informatik durchgeführt:

Modul: Praxissemester (2 Credits)	
Name im Diploma Supplement	Obligatory Long-term School Placement (duration: five months)
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	60 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 45 Stunden</li> <li>• Vorbereitung, Nachbereitung: 15 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• identifizieren praxisbezogene Entwicklungsaufgaben schulformspezifisch</li> <li>• planen kleinere Studien-, Unterrichts- und/oder Forschungsprojekte (auch unter Berücksichtigung der Interessen der Praktikumsschulen)</li> <li>• können dabei wissenschaftliche Inhalte der Bildungswissenschaften und der Unterrichtsfächer auf Situationen und Prozesse schulischer Praxis beziehen</li> <li>• kennen Ziele und Phasen empirischer Forschung und wenden ausgewählte Methoden exemplarisch in den schul- und unterrichtsbezogenen Projekten an</li> <li>• sind befähigt, Lehr-Lernprozesse unter Berücksichtigung individueller, institutioneller und gesellschaftlicher Rahmenbedingungen zu gestalten</li> <li>• reflektieren theoriegeleitet Beobachtungen und Erfahrungen in Schule und Unterricht</li> </ul> <p><i>davon Schlüsselqualifikationen:</i></p> <ul style="list-style-type: none"> <li>• Organisationsfähigkeit, realistische Zeit- und Arbeitsplanung</li> <li>• Planungs-, Projekt- und Innovationsmanagement</li> <li>• Kooperationsfähigkeit</li> <li>• Erschließung, kritische Sichtung und Präsentation von Forschungsergebnissen</li> <li>• Anwendung wissenschaftlicher Methoden und Auswertungsstrategien</li> <li>• konstruktive Wertschätzung von Diversity</li> </ul>
Prüfungsmodalitäten	Keine <b>Hinweis:</b> Insgesamt sind Begleitveranstaltungen in den Bildungswissenschaften sowie den beiden Fächern/Fachdidaktiken abzuschließen. In diesem Modulhandbuch ist ausschließlich der sich auf das Fach Informatik beziehende Anteil ausgewiesen. Es sind zwei Modulteilprüfungen in den beiden Fächern mit Studienprojekt zu absolvieren. Die beiden Modulteilprüfungen gehen zu gleichen Teilen in die Modulabschlussnote ein.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014&gt;Begleitmodul zum Praxissemester "Schule und Unterricht forschend verstehen" &gt;2. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Seminar: Begleitung des Praxissemesters (2 Credits)</li> </ul>
WIWI-M0885 Modul: Praxissemester	

Seminar: Begleitung des Praxissemesters (2 Credits)			
Name im Diploma Supplement	Teacher Development - Reflective Practice		
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>		
Lehrperson	Prof. Dr. Torsten Brinda		
SWS	3	Sprache	deutsch
Turnus	jedes Semester	maximale Hörschaft	unbeschränkt
<b>empfohlenes Vorwissen</b>			
<ul style="list-style-type: none"> <li>• Grundlagen aus den Bereichen Programmierung, Algorithmen und Datenstrukturen, informatische Modellierung, Datenbanksysteme, Rechnernetze und -architektur</li> <li>• Grundkonzepte der Allgemeinen Didaktik</li> <li>• Grundkonzepte der Didaktik der Informatik</li> <li>• Prozess der Planung schulbezogener Lehr-Lern-Situationen und –Einheiten in der Informatik</li> <li>• Ausgewählte Forschungskonzeptionen, –methoden und –ergebnisse im Bereich der Didaktik der Informatik</li> </ul>			
<b>Lehrinhalte</b>			
<ul style="list-style-type: none"> <li>• Methoden forschenden Lernens</li> <li>• Konzeption, Durchführung und Evaluation von Studien- und Unterrichtsprojekten zu ausgewählten Themen der informatischen Bildung</li> <li>• Beobachtung und Bewertung von Lehr-Lern-Prozessen</li> <li>• Lernerfolgskontrollen</li> <li>• Planung, Durchführung und Evaluation von Informatikunterrichtsausschnitten unter Begleitung</li> </ul>			
<b>Literaturangaben</b>			
Literaturhinweise werden semesteraktuell zu Beginn der Veranstaltung bekannt gegeben.			
WIWI-C0308 Seminar: Begleitung des Praxissemesters im Modul WIWI-M0885: Praxissemester			

## **Pflichtbereich Masterarbeit einschließlich Begleitmodul - 4. Fachsemester, Pflicht**

Für das 'Begleitmodul zur Masterarbeit' werden in diesem Modulhandbuch nur die Credits und die Modulverantwortlichen für den Bereich der Informatik ausgewiesen.

Modul: Professionelles Handeln wissenschaftsbasiert weiterentwickeln (3 Credits)	
Name im Diploma Supplement	A Science-based Approach to Teacher Development
Verantwortlich	Dozenten aus der Informatik und der Didaktik der Informatik
Voraussetzungen	Siehe Prüfungsordnung.
Workload	90 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 20 Stunden</li> <li>• Vorbereitung, Nachbereitung: 55 Stunden</li> <li>• Prüfungsvorbereitung: 15 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• kennen Forschungsmethoden sowie deren methodologische Begründungszusammenhänge und können auf dieser Grundlage Forschungsergebnisse rezipieren</li> <li>• haben vertiefte Kenntnisse über den Aufbau und Ablauf von Forschungsprojekten mit anwendungsbezogenen, schulrelevanten Themen</li> <li>• können ihre bildungswissenschaftlichen, fachlichen, fachdidaktischen und methodischen Kompetenzen im Hinblick auf konkrete Theorie-Praxis-Fragen integrieren und anwenden</li> <li>• stellen die Masterarbeitsthema-relevanten Gebiete der Informatik bzw. der Didaktik der Informatik in ihren wissenschaftlichen Zusammenhang, sofern die Masterarbeit in der Informatik oder der Didaktik der Informatik angefertigt wird</li> <li>• durchdringen die Informatik-relevanten Aspekte des Masterarbeits-Themas, sofern die Masterarbeit nicht in der Informatik oder der Didaktik der Informatik angefertigt wird</li> </ul> <i>davon Schlüsselqualifikationen:</i> <ul style="list-style-type: none"> <li>• interdisziplinäres Verstehen, Fähigkeit verschiedene Sichtweisen einzunehmen und anzuwenden</li> <li>• Organisationsfähigkeit, realistische Zeit- und Arbeitsplanung</li> <li>• Erschließung, kritische Sichtung und Präsentation von Forschungsergebnissen</li> <li>• Professionelles Selbstverständnis des Berufes als ständige Lernaufgabe</li> </ul>
Prüfungsmodalitäten	Studienleistung: Präsentation im Umfang von 20 Minuten zu einem vereinbarten Thema. Das Modul ist unbenotet.
Verwendung in Studiengängen	• LA Info GyGe Master 2014>Pflichtbereich Masterarbeit einschließlich Begleitmodul >4. Fachsemester, Pflicht
Bestandteile	• Seminar: Professionelles Handeln wissenschaftsbasiert weiterentwickeln aus der Perspektive der Informatik und der Didaktik der Informatik (3 Credits)
WIWI-M0140 Modul: Professionelles Handeln wissenschaftsbasiert weiterentwickeln	

Seminar: Professionelles Handeln wissenschaftsbasiert weiterentwickeln aus der Perspektive der Informatik und der Didaktik der Informatik (3 Credits)			
Name im Diploma Supplement			
Anbieter	Lehrstühle der Informatik und der Didaktik der Informatik		
Lehrperson	Dozenten aus der Informatik und der Didaktik der Informatik		
SWS	2	Sprache	deutsch
Turnus	jedes Semester	maximale Hörschaft	unbeschränkt
empfohlenes Vorwissen	Keines		
Lehrinhalte	Die Lernziele werden durch eine individuell betreute Arbeit vermittelt, die thematisch so ausgerichtet ist, wie unter den Lernzielen angegeben.		
Literaturangaben			
WIWI-C0010 Seminar: Professionelles Handeln wissenschaftsbasiert weiterentwickeln aus der Perspektive der Informatik und der Didaktik der Informatik im Modul WIWI-M0140: Professionelles Handeln wissenschaftsbasiert weiterentwickeln			

Das Thema der Abschlussarbeit wird i.d.R. von einer Hochschullehrerin oder einem Hochschullehrer, einer Hochschuldozentin oder einem Hochschuldozenten bzw. einer Privatdozentin oder einem Privatdozenten der Fakultät für Wirtschaftswissenschaften gestellt und betreut, die oder der im jeweiligen Studiengang Lehrveranstaltungen durchführt. Potentielle Betreuerinnen und Betreuer einer Abschlussarbeit sind, vorbehaltlich der Bestellung weiterer Betreuerinnen oder Betreuer durch den Prüfungsausschuss, nachfolgend mit Verweisen zu den jeweiligen Voraussetzungen und Bewerbungsmodalitäten aufgeführt. Im übrigen gelten die Bestimmungen der Prüfungsordnung.



Modul: Masterarbeit (Master LA Info GyGe) (20 Credits)	
Name im Diploma Supplement	Master Thesis
Verantwortlich	Prof. Dr. Torsten Brinda
Voraussetzungen	Siehe Prüfungsordnung.
Workload	600 Stunden studentischer Workload gesamt, davon: <ul style="list-style-type: none"> <li>• Präsenzzeit: 30 Stunden</li> </ul>
Dauer	Das Modul erstreckt sich über 1 Semester.
Qualifikationsziele	Die Studierenden <ul style="list-style-type: none"> <li>• sind in der Lage, innerhalb einer vorgegebenen Frist ein anspruchsvolles Problem der Informatik oder der Didaktik der Informatik selbstständig und unter Anwendung wissenschaftlicher Methoden zu lösen und darzustellen</li> <li>• sind befähigt zu selbstständiger Literaturrecherche und Eingrenzung eines Themas</li> <li>• verfügen über ein planvolles und rationales Zeitmanagement für einen längeren Zeitraum</li> <li>• wenden Techniken wissenschaftlichen Arbeitens an</li> <li>• wenden nach eigenständiger Prüfung fachwissenschaftliche Theorien, Modelle und domänenspezifische Forschungsmethoden auf eine neue Frage- bzw. Problemstellung an</li> <li>• sind in der Lage, den aktuellen Stand wissenschaftlicher Erkenntnis zu dem zu bearbeitenden Thema aufzubereiten</li> <li>• erarbeiten Lösungsansätze für die bearbeitete Frage- bzw. Problemstellung auf aktuellem wissenschaftlichen Niveau</li> <li>• identifizieren weiteren Forschungsbedarf</li> </ul>
Prüfungsmodalitäten	Zum Modul erfolgt eine modulbezogene Prüfung in der Gestalt einer schriftlichen Arbeit (in der Regel: 50 Seiten). Die Bearbeitungszeit für die Masterarbeit beträgt 15 Wochen. Nähere Modalitäten sind in der Prüfungsordnung geregelt.
Verwendung in Studiengängen	<ul style="list-style-type: none"> <li>• LA Info GyGe Master 2014 &gt; Pflichtbereich Masterarbeit einschließlich Begleitmodul &gt; 4. Fachsemester, Wahlpflicht</li> </ul>
Bestandteile	<ul style="list-style-type: none"> <li>• Abschlussarbeit: Didaktik der Informatik</li> <li>• Abschlussarbeit: Networked Embedded Systems</li> <li>• Abschlussarbeit: Software Systems Engineering</li> <li>• Abschlussarbeit: Software-Engineering, insb. mobile Anwendungen</li> <li>• Abschlussarbeit: Spezifikation von Softwaresystemen</li> <li>• Abschlussarbeit: Technik der Rechnernetze</li> <li>• Abschlussarbeit: Mensch-Computer Interaktion</li> <li>• Abschlussarbeit: Sichere Software Systeme</li> <li>• Abschlussarbeit: Visualisierung</li> </ul>
WIWI-M0205 Modul: Masterarbeit (Master LA Info GyGe)	

Abschlussarbeit: Didaktik der Informatik (20 Credits)	
Anbieter	Lehrstuhl für Didaktik der Informatik <a href="http://www.ddi.wiwi.uni-due.de/">http://www.ddi.wiwi.uni-due.de/</a>
Gutachter	Prof. Dr. Torsten Brinda
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0023 <b>Abschlussarbeit: Didaktik der Informatik</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

Abschlussarbeit: Networked Embedded Systems (20 Credits)	
Anbieter	Networked Embedded Systems <a href="http://www.nes.uni-due.de/">http://www.nes.uni-due.de/</a>
Gutachter	Prof. Dr. Pedro José Marrón
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0024 <b>Abschlussarbeit: Networked Embedded Systems</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

Abschlussarbeit: Software Systems Engineering (20 Credits)	
Anbieter	Lehrstuhl für Software Systems Engineering <a href="http://www.sse.uni-due.de/">http://www.sse.uni-due.de/</a>
Gutachter	Prof. Dr. Klaus Pohl
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der Homepage des Lehrstuhls: <a href="#">Bachelor</a> bzw. <a href="#">Master</a>	
WIWI-F0026 <b>Abschlussarbeit: Software Systems Engineering</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Software-Engineering, insb. mobile Anwendungen (20 Credits)**

Anbieter	Lehrstuhl für Software-Engineering, insb. mobile Anwendungen <a href="http://www.se.wiwi.uni-due.de/">http://www.se.wiwi.uni-due.de/</a>
Gutachter	Prof. Dr. Volker Gruhn
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0027 <b>Abschlussarbeit: Software-Engineering, insb. mobile Anwendungen</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Spezifikation von Softwaresystemen (20 Credits)**

Anbieter	Lehrstuhl für Spezifikation von Softwaresystemen <a href="http://www.s3.uni-due.de/">http://www.s3.uni-due.de/</a>
Gutachter	Prof. Dr. Michael Goedicke
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0025 <b>Abschlussarbeit: Spezifikation von Softwaresystemen</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Technik der Rechnernetze (20 Credits)**

Anbieter	Lehrstuhl für Technik der Rechnernetze <a href="http://www.tdr.wiwi.uni-due.de/">http://www.tdr.wiwi.uni-due.de/</a>
Gutachter	Prof. Dr.-Ing. Erwin P. Rathgeb
Sprache	deutsch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0028 <b>Abschlussarbeit: Technik der Rechnernetze</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Mensch-Computer Interaktion (20 Credits)**

Anbieter	Juniorprofessur für Mensch-Computer Interaktion <a href="https://www.hci.wiwi.uni-due.de/">https://www.hci.wiwi.uni-due.de/</a>
Gutachter	Jun.-Prof. Dr. Stefan Schneegaß
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0019 <b>Abschlussarbeit: Mensch-Computer Interaktion</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Sichere Software Systeme (20 Credits)**

Anbieter	Juniorprofessur für Sichere Software Systeme <a href="https://www.syssec.wiwi.uni-due.de/">https://www.syssec.wiwi.uni-due.de/</a>
Gutachter	Jun.-Prof. Dr. Lucas Davi
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0020 <b>Abschlussarbeit: Sichere Software Systeme</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	

**Abschlussarbeit: Visualisierung (20 Credits)**

Anbieter	Juniorprofessur für Visualisierung <a href="https://www.vis.wiwi.uni-due.de/">https://www.vis.wiwi.uni-due.de/</a>
Gutachter	Jun.-Prof. Dr. Fabian Beck
Sprache	deutsch/englisch
<b>Beschreibung</b> Informationen zu den Voraussetzungen und zur Bewerbung finden Sie auf der <a href="#">Homepage des Lehrstuhls</a> .	
WIWI-F0021 <b>Abschlussarbeit: Visualisierung</b> im Modul WIWI-M0205: Masterarbeit (Master LA Info GyGe)	