

**Modulhandbuch**  
**Studiengang Bachelor of Science Softwaretechnik**  
Prüfungsordnung: 123-2009  
Hauptfach

Sommersemester 2018  
Stand: 09. April 2018

Universität Stuttgart  
Keplerstr. 7  
70174 Stuttgart

## Kontaktpersonen:

---

Studiendekan/in:	Univ.-Prof. Stefan Wagner Institut für Softwaretechnologie Tel.: 0711/685-88455 E-Mail: stefan.wagner@informatik.uni-stuttgart.de
Studiengangsmanager/in:	Katrin Schneider Institut für Maschinelle Sprachverarbeitung Tel.: 685 88520 E-Mail: katrin.schneider@informatik.uni-stuttgart.de
Prüfungsausschussvorsitzende/r:	Univ.-Prof. Lars Grunske Zuverlässige Softwaresysteme E-Mail: lars.grunske@iste.uni-stuttgart.de
Fachstudienberater/in:	Katrin Schneider Institut für Maschinelle Sprachverarbeitung Tel.: 685 88520 E-Mail: katrin.schneider@informatik.uni-stuttgart.de
Stundenplanverantwortliche/r:	Apl. Prof. Ulrich Hertrampf Institut für Formale Methoden der Informatik Tel.: 7816-344 E-Mail: ulrich.hertrampf@f05.uni-stuttgart.de

## Inhaltsverzeichnis

<b>Präambel</b> .....	<b>5</b>
<b>Qualifikationsziele</b> .....	<b>6</b>
<b>100 Basismodule</b> .....	<b>7</b>
10190 Mathematik für Informatiker und Softwaretechniker .....	8
10260 Programmierkurs .....	10
10280 Programmierung und Software-Entwicklung .....	12
10940 Theoretische Grundlagen der Informatik .....	14
12060 Datenstrukturen und Algorithmen .....	16
16520 Software-Qualität .....	18
17210 Einführung in die Softwaretechnik .....	19
<b>200 Kernmodule</b> .....	<b>21</b>
11890 Algorithmen und Berechenbarkeit .....	22
14360 Einführung in die Technische Informatik .....	23
14370 Fachstudie Softwaretechnik .....	25
14390 Programmentwicklung .....	26
14480 Sichere und zuverlässige Softwaresysteme .....	27
16500 Software Engineering .....	29
16510 Software-Praktikum .....	30
<b>210 Kernmodul Studienprojekt</b> .....	<b>31</b>
<b>300 Ergänzungsmodule</b> .....	<b>32</b>
310 Katalog SWT .....	33
10210 Mensch-Computer-Interaktion .....	34
10220 Modellierung .....	36
36530 Rechnerorganisation 1 .....	38
40090 Systemkonzepte und -programmierung .....	40
41590 Einführung in die Numerik und Stochastik für Softwaretechniker .....	42
320 Katalog ISG .....	44
10030 Architektur von Anwendungssystemen .....	45
10060 Computergraphik .....	47
10110 Grundlagen der Künstlichen Intelligenz .....	49
10170 Imaging Science .....	51
10180 Information Retrieval and Text Mining .....	53
10210 Mensch-Computer-Interaktion .....	54
10220 Modellierung .....	56
10330 Systemkonzepte und -programmierung .....	58
36530 Rechnerorganisation 1 .....	60
39040 Rechnernetze .....	62
41590 Einführung in die Numerik und Stochastik für Softwaretechniker .....	64
42410 Grundlagen des Wissenschaftlichen Rechnens .....	66
330 Katalog ISW .....	68
10030 Architektur von Anwendungssystemen .....	69
10060 Computergraphik .....	71
10110 Grundlagen der Künstlichen Intelligenz .....	73
10170 Imaging Science .....	75
10180 Information Retrieval and Text Mining .....	77

10210 Mensch-Computer-Interaktion .....	78
10220 Modellierung .....	80
36530 Rechnerorganisation 1 .....	82
39040 Rechnernetze .....	84
40090 Systemkonzepte und -programmierung .....	86
41590 Einführung in die Numerik und Stochastik für Softwaretechniker .....	88
42410 Grundlagen des Wissenschaftlichen Rechnens .....	90
71740 System and Web Security .....	92
78900 Introduction to Modern Cryptography .....	93
<b>400 Schlüsselqualifikationen fachaffin .....</b>	<b>95</b>
16610 Studienprojekt-Pr .....	96
<b>81110 Bachelorarbeit Softwaretechnik .....</b>	<b>98</b>

## Präambel

Wie andere Ingenieure auch, arbeiten die meisten Informatikerinnen und Informatiker nach ihrer Ausbildung konstruktiv. Sie entwickeln also neue oder verändern, verbessern und erweitern bestehende Software. Daran ist der Studiengang Softwaretechnik ausgerichtet: Er betont die konstruktiven Aspekte der Informatik. Der Lehrstoff, der den Studierenden vermittelt wird, überlappt mit dem des Studiengangs Informatik, aber in der Softwaretechnik ist es wichtig, dass das Gehörte auch angewendet wird. Die Absolventen der Softwaretechnik sind damit besonders gut auf alle Informatikberufe vorbereitet, in denen an Software gearbeitet wird, auch auf die Managementaufgaben, die die meisten früher oder später übernehmen werden.

In jedem Softwareprojekt wird das Ziel verfolgt, Software so zu entwickeln oder zu verändern, dass sie am Ende den Anforderungen der Kunden hinsichtlich Funktion und Qualität entspricht. Dabei sollen die Kosten und die Entwicklungszeit möglichst niedrig sein. Dieses Ziel wird den Studierenden der Softwaretechnik in Vorlesungen und Übungen, vor allem aber in einer Reihe von Projekten, vermittelt bis hin zum zwölfmonatigen Studienprojekt, in dem etwa zehn Studierende gemeinsam und selbstorganisiert eine anspruchsvolle Entwicklungsaufgabe lösen. Neben der Technik werden damit Arbeit im Team, Kommunikation und Präsentation erlernt und geübt.

Auf den Bachelor-Studiengang Softwaretechnik baut der gleichnamige Master-Studiengang auf. Den Softwaretechnik-Absolventen stehen aber auch die anderen Masterstudiengänge der Informatik offen.

## Qualifikationsziele

Das Curriculum des Studienganges sieht im 1. und 2. Semester eine Vertiefung der mathematischen und technischen Grundlagen der Informatik und Softwareentwicklung in Pflichtmodulen vor.

Im 3. bis 6. Semester liegt der Schwerpunkt der Ausbildung auf diversen Grundlagen- und Spezialisierungsfächern verschiedener Bereiche der Informatik und Softwaretechnik. Zusätzliche Inhalte sind Seminare, fachliche Module als Wahlpflichtbereich sowie fachübergreifende Schlüsselqualifikationen. Die praktische Ausbildung wird durch ein Studienprojekt vertieft, in dem ein komplexes Softwareprojekt im Team von ca. 10 Personen innerhalb von 12 Monaten durchgeführt wird. Im 6. Semester liegt ein weiterer Schwerpunkt auf der Bachelorarbeit. Mit der Bachelorarbeit ist die Befähigung zu zeigen, innerhalb einer vorgegebenen Frist eine komplexe Aufgabenstellung aus dem Bereich der Softwaretechnik selbständig nach wissenschaftlichen Methoden zu bearbeiten und die Ergebnisse sachgerecht darzustellen. Die Absolventinnen und Absolventen des Bachelorstudienganges Softwaretechnik

- verfügen über ein vertieftes mathematisches und ingenieurwissenschaftliches Wissen, welches sie befähigt, Probleme und Aufgabenstellungen der Softwareentwicklung zu verstehen, kritisch einzuschätzen und zu lösen.
- sind in der Lage, jenes erlangte Wissen auf Erkenntnisse der Ingenieurwissenschaften anzuwenden.
- verfügen über Fachwissen auf Gebieten der Softwaretechnik und Informatik und können Aufgaben u.a. der Softwareentwicklung wissenschaftlich erkennen, beschreiben und bewerten, analysieren und lösen.
- haben umfassendes Verständnis über Softwareentwicklungsmethoden, ihre Anwendungsmöglichkeiten und verfügen über die Fertigkeit, Konzepte und Lösungen für neue Softwaresysteme zu entwickeln.
- verfügen über eine verantwortliche und selbständige wissenschaftliche Arbeitsweise.

Die Beschäftigungsbereiche der Absolventinnen und Absolventen liegen u. a. in der Datenverarbeitungs- und Entwicklungsindustrie und in industriellen, kommerziellen, verwaltungstechnischen und wissenschaftlichen Anwendungen von Datenverarbeitungssystemen.

## 100 Basismodule

---

Zugeordnete Module:	10190	Mathematik für Informatiker und Softwaretechniker
	10260	Programmierkurs
	10280	Programmierung und Software-Entwicklung
	10940	Theoretische Grundlagen der Informatik
	12060	Datenstrukturen und Algorithmen
	16520	Software-Qualität
	17210	Einführung in die Softwaretechnik

---

## Modul: 10190 Mathematik für Informatiker und Softwaretechniker

2. Modulkürzel:	080300100	5. Moduldauer:	Zweisemestrig
3. Leistungspunkte:	18 LP	6. Turnus:	Wintersemester
4. SWS:	12	7. Sprache:	Deutsch
8. Modulverantwortlicher:	PD Dr. Andreas Markus Kollross		
9. Dozenten:	Wolfgang Rump Andreas Markus Kollross Peter Lesky Wolf-Patrick Düll		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 1. Semester → Basismodule B.Sc. Softwaretechnik, PO 123-2017, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine, die Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Studierenden haben die mathematischen Grundlagen für die Studiengänge Informatik bzw. Softwaretechnik erarbeitet und den selbständigen und kreativen Umgang mit den mathematischen Stoffgebieten gelernt.		
13. Inhalt:	<p>1. Semester:</p> <ul style="list-style-type: none"> <li>• Grundlagen (Aussagenlogik, Mengen, Relationen, Abbildungen, Zahlenmengen, Grundbegriffe der Algebra)</li> <li>• Lineare Algebra (Vektorräume, lineare Abbildungen, Matrizen, Determinanten, lineare Gleichungssysteme, Eigenwerte, Normalformen, Hauptachsentransformation, Skalarprodukte)</li> <li>• Analysis (Konvergenz, Zahlenfolgen und Zahlenreihen, stetige Abbildungen, Folgen und Reihen von Funktionen, spezielle Funktionen)</li> </ul> <p>2. Semester:</p> <ul style="list-style-type: none"> <li>• Differential- und Integralrechnung (Funktionen einer und mehrerer Variablen, Ableitungen, Taylorentwicklungen, Extremwerte, Integration, Anwendungen)</li> <li>• Gewöhnliche Differentialgleichungen (elementar lösbare Differentialgleichungen, Existenz und Eindeutigkeit von Lösungen)</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Anna Sändig, Mathematik, Vorlesungskripte , SS 2007</li> <li>• D. Hachenberger, Mathematik für Informatiker, 2005</li> <li>• M. Brill, Mathematik für Informatiker, 2001</li> <li>• P.Hartmann, Mathematik für Informatiker, 2002</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101901 Vorlesung Mathematik 1 für Informatik und Softwaretechnik</li> <li>• 101902 Übung Mathematik 1 für Informatik und Softwaretechnik</li> <li>• 101903 Vorlesung Mathematik 2 für Informatik und Softwaretechnik</li> <li>• 101904 Übung Mathematik 2 für Informatik und Softwaretechnik</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 126 Stunden Nachbearbeitungszeit: 414 Stunden		
17. Prüfungsnummer/n und -name:	• 10191 Mathematik für Informatiker und Softwaretechniker (PL), Schriftlich, 120 Min., Gewichtung: 1		

- V Vorleistung (USL-V), Schriftlich oder Mündlich  
Ein Übungsschein aus den beiden Veranstaltungen, jeweils im 1. oder 2. Fachsemester zu erwerben
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Geometrie

---

## Modul: 10260 Programmierkurs

2. Modulkürzel:	051520010	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	3 LP	6. Turnus:	Wintersemester
4. SWS:	2	7. Sprache:	Weitere Sprachen
8. Modulverantwortlicher:	Univ.-Prof. Dr. Jonas Kuhn		
9. Dozenten:	Jason Utt		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:			
12. Lernziele:	<p>Selbstständiges Erstellen von Programmen und Lösung von Programmieraufgaben in der Programmiersprache Python, mit einem Schwerpunkt auf Konzepten, die für die maschinelle Sprachverarbeitung und Computerlinguistik wichtig sind.</p> <p>--</p> <p>Independently writing programs and solving programming tasks in the programming language Python, with emphasis on concepts relevant for Natural Language Processing and Computational Linguistics.</p>		
13. Inhalt:	<p>Das Modul Programmierkurs richtet sich primär an Studierende der Maschinellen Sprachverarbeitung (3. Semester), Computerlinguistik und Digital Humanities. Vermittelt werden die wichtigsten Konzepte der Programmiersprache Python und praktische Erfahrung bei der Erstellung von Python-Programmen bei der Verarbeitung von sprachlichen Daten und Ressourcen. Die Modulveranstaltung und die Materialien sind in der Regel überwiegend englischsprachig, es werden jedoch deutschsprachige Hilfestellungen angeboten.</p> <p>--</p> <p>The module primarily targets students in Natural Language Processing (3rd semester), Computational Linguistics and Digital Humanities. It covers the key concepts of the programming language Python and provides practical experience in writing Python programs in the context of processing linguistic data and resources.</p> <p>Typically, the lectures of the module course as well as the materials are in English, however, students not fluent in English in the programming context will receive support in German.</p>		
14. Literatur:	Folien. Slides.		
15. Lehrveranstaltungen und -formen:	• 102601 Übung Programmierkurs		
16. Abschätzung Arbeitsaufwand:	Präsenzzeit: 21 Stunden Nachbearbeitungszeit: 69 Stunden		
17. Prüfungsnummer/n und -name:	10261 Programmierkurs (USL), Sonstige, Gewichtung: 1 Übungsschein - Scheinkriterien werden zu Beginn der Veranstaltung angekündigt. Criteria for credits are announced at the beginning of the course.		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Grundlagen der Computerlinguistik

---

## Modul: 10280 Programmierung und Software-Entwicklung

2. Modulkürzel:	051520005	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	9 LP	6. Turnus:	Wintersemester
4. SWS:	6	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Basismodule B.Sc. Softwaretechnik, PO 123-2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine. Teilnahme an einem Vorkurs Java ist hilfreich aber nicht notwendig.		
12. Lernziele:	Die Teilnehmer haben einen Überblick über das Gebiet der Informatik. Sie haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Die Programmiersprache Java und die virtuelle Maschine</li> <li>• Objekte, Klassen, Schnittstellen, Blöcke, Programmstrukturen, Kontrakte</li> <li>• Klassenmodellierung mit der UML</li> <li>• Objekterzeugung und -ausführung</li> <li>• Boolesche Logik</li> <li>• Verzweigungen, Schleifen, Routinen, Abstraktionen, Modularisierung, Variablen, Zuweisungen</li> <li>• Rechner, Hardware</li> <li>• Syntaxdarstellungen</li> <li>• Übersicht über Programmiersprachen und -werkzeuge</li> <li>• Grundlegende Datenstrukturen und Algorithmen</li> <li>• Vererbung, Polymorphe</li> <li>• Semantik</li> <li>• Programmierung graphischer Oberflächen</li> <li>• Übergang zum Software Engineering</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Appelrath, Hans-Jürgen und Ludewig, Jochen, Skriptum Informatik - eine konventionelle Einführung, Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999</li> <li>• Meyer, Bertrand, Touch of Class, Springer-Verlag, 2009</li> <li>• Savitch, Walter, Java. An Introduction to Problem Solving and Programming, Pearson, 6. Auflage, 2012</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 102801 Vorlesung Programmierung und Softwareentwicklung</li> <li>• 102802 Übung Programmierung und Softwareentwicklung</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 63 h Eigenstudiumstunden: 207 h		

Gesamtstunden: 270 h

---

17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10281 Programmierung und Software-Entwicklung (PL), Schriftlich, 90 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich [10281] Programmierung und Software-Entwicklung (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0, [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich, Vorleistung: Übungsschein. Voraussetzungen werden zu Beginn vom Dozenten festgesetzt. Dazu gehören eine bestimmte Anzahl von Vorträgen in den Übungen und ein bestimmter Teil der Übungspunkte.</li></ul>
18. Grundlage für ... :	Datenstrukturen und Algorithmen
19. Medienform:	<ul style="list-style-type: none"><li>• Folien über Beamer</li><li>• Tafelanschrieb</li></ul>
20. Angeboten von:	Architektur von Anwendungssystemen

---

## Modul: 10940 Theoretische Grundlagen der Informatik

2. Modulkürzel:	050420005	5. Moduldauer:	Zweimestrig
3. Leistungspunkte:	12 LP	6. Turnus:	Wintersemester
4. SWS:	8	7. Sprache:	Deutsch
8. Modulverantwortlicher:	apl. Prof. Dr. rer. nat. habil. Ulrich Hertrampf		
9. Dozenten:	Volker Diekert Ulrich Hertrampf		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Keine		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Logik und Diskrete Strukturen: Die Studierenden haben die grundsätzlichen Kenntnisse in Logik und Diskreter Mathematik erworben, wie sie in den weiteren Grundvorlesungen der Informatik in verschiedenen Bereichen benötigt werden.</li> <li>• Automaten und Formale Sprachen: Die Studierenden beherrschen wichtige theoretische Grundlagen der Informatik, insbesondere die Theorie und Algorithmik endlicher Automaten. Hierzu gehört das Kennenlernen, Einordnung und Trennung der Chomskyschen Sprachklassen.</li> </ul>		
13. Inhalt:	<p>Logik und Diskrete Strukturen:</p> <ul style="list-style-type: none"> <li>• Einführung in die Aussagenlogik: Semantik (Wahrheitswerte), Syntax (Axiome und Schlussregeln), Normalformen, Hornformeln, Endlichkeitssatz, aussagenlogische Resolution,</li> <li>• Einführung in die Prädikatenlogik 1. Stufe: Semantik und Syntax, Normalformen, Unifikatoren, Herbrand-Theorie, prädikatenlogische Resolution,</li> <li>• Elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, Euklidischer Algorithmus, Chinesischer Restsatz, Primzahltests, RSA-Verfahren, Wachstumsabschätzungen, Grundbegriffe der Wahrscheinlichkeitsrechnung, Kombinatorik, Graphen.</li> </ul> <p>Automaten und Formale Sprachen:</p> <ul style="list-style-type: none"> <li>• Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, Iterationslemmata für reguläre und kontextfreie Sprachen, Normalformen, Kellerautomaten, Lösen des Wortproblems kontextfreier Sprachen mit dem CYK-Algorithmus, linear beschränkte Automaten, kontextsensitive Grammatiken, Typ 0-Grammatiken und Turingmaschinen.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988.</li> <li>• Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 109401 Vorlesung Logik und Diskrete Strukturen</li> <li>• 109403 Vorlesung Automaten und Formale Sprachen</li> <li>• 109404 Übung Automaten und Formale Sprachen</li> <li>• 109402 Übung Logik und Diskrete Strukturen</li> </ul>		

- 109405 Zusatzkolloquium Theoretische Grundlagen der Informatik für MSV (freiwillig)
- 

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:

- 10941 Theoretische Grundlagen der Informatik (PL), Schriftlich, 120 Min., Gewichtung: 1
  - V Vorleistung (USL-V), Schriftlich oder Mündlich, 30 Min.
- [10941] Theoretische Grundlagen der Informatik (PL), schriftliche Prüfung, 120 Min., Gewicht: 1.0, Prüfungsvorleistung: Übungsschein [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich, 30 Min.
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

Theoretische Informatik

---

## Modul: 12060 Datenstrukturen und Algorithmen

2. Modulkürzel:	051510005	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	9 LP	6. Turnus:	Sommersemester
4. SWS:	6	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 2. Semester → Basismodule B.Sc. Softwaretechnik, PO 123-2017, 2. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	Modul 10280 Programmierung und Software-Entwicklung		
12. Lernziele:	<p>Die Studierenden kennen nach engagierter Mitarbeit in dieser Veranstaltung diverse zentrale Algorithmen auf geeigneten Datenstrukturen, die für eine effiziente Nutzung von Computern unverzichtbar sind. Sie können am Ende zu gängigen Problemen geeignete programmiersprachliche Lösungen angeben und diese in einer konkreten Programmiersprache formulieren.</p> <p>Die Lernziele lassen sich wie folgt zusammenfassen:</p> <ul style="list-style-type: none"> <li>• Kenntnis der Eigenschaften elementarer und häufig benötigter Algorithmen</li> <li>• Verständnis für die Auswirkungen theoretischer und tatsächlicher Komplexität</li> <li>• Erweiterung der Kompetenz im Entwurf und Verstehen von Algorithmen und der zugehörigen Datenstrukturen</li> <li>• Erste Begegnung mit nebenläufigen Algorithmen</li> </ul>		
13. Inhalt:	<p>Es werden die folgenden Themen behandelt:</p> <ul style="list-style-type: none"> <li>• Vorgehensweise bei der Entwicklung und Implementierung von Algorithmen</li> <li>• Komplexität und Effizienz von Algorithmen, O-Notation</li> <li>• Listen (Stack, Queue, doppelt verkettete Listen)</li> <li>• Sortierverfahren (Selection-, Insertion-, Bubble-, Merge-, Quick-Sort)</li> <li>• Bäume (Binär-, AVL-, 2-3-4-, Rot-Schwarz-, B-Bäume, Suchbäume, Traversierung, Heap)</li> <li>• Räumliche Datenstrukturen (uniforme Gitter, Oktal-, BSP-, kD-, CSG-Bäume, Bounding-Volumes)</li> <li>• Graphen (Datenstrukturen, DFS, BFS, topologische Traversierung, Dijkstra-, A*-, Bellman-Ford-Algorithmen, minimale Spannbäume, maximaler Fluss)</li> <li>• Räumliche Graphen (Triangulierung, Voronoi, Delaunay, Graph-Layout)</li> <li>• Textalgorithmen (String-Matching, Knuth-Morris-Pratt, Boyer-Moore, reguläre Ausdrücke, Levenshtein-Distanz)</li> <li>• Hashing (Hashfunktionen, Kollisionen)</li> <li>• Verteilte Algorithmen (Petri-Netze, Programmieren nebenläufiger Abläufe, einige parallele und parallelisierte Algorithmen)</li> </ul>		

- Algorithmenentwurf und -muster (inkrementell, greedy, divide-and-conquer, dynamische Programmierung, Backtracking, randomisierte Algorithmen)
  - Maschinelles Lernen (überwachtes Lernen, Entscheidungsbäume, SVM, neuronale Netze, unüberwachtes Lernen, k-Means)
- 

14. Literatur:

- G. Saake, K. Sattler. *Algorithmen und Datenstrukturen: Eine Einführung mit Java* . 5. Auflage, dpunkt-Verlag, 2013
  - T. Ottmann, P. Widmayer. *Algorithmen und Datenstrukturen* . 5. Auflage, Springer-Verlag, 2012
- 

15. Lehrveranstaltungen und -formen:

- 120601 Vorlesung Datenstrukturen und Algorithmen
  - 120602 Übung Datenstrukturen und Algorithmen
- 

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:

- 12061 Datenstrukturen und Algorithmen (PL), Schriftlich, 120 Min., Gewichtung: 1
  - V Vorleistung (USL-V), Schriftlich oder Mündlich
- Prüfungsvorleistung: Übungsschein. Die genauen Details der Übungsleistungen und Ihrer Anrechnung werden zu Beginn der Lehrveranstaltung bekannt gegeben.
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von:

Intelligente Systeme

---

## Modul: 16520 Software-Qualität

2. Modulkürzel:	051520105	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	3 LP	6. Turnus:	Wintersemester
4. SWS:	2	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Ivan Bogicevic Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 1. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>gleichzeitiger Besuch der Programmierung und Softwareentwicklung</li> </ul>		
12. Lernziele:	Die Teilnehmer kennen und verstehen den Begriff der Software-Qualität. Insbesondere erfahren sie die Schwierigkeiten bei der Evolution großer Systeme. Sie kennen Techniken, deren Anwendung zu einer guten Software-Qualität beiträgt, und können sie anwenden.		
13. Inhalt:	In der Vorlesung wird der Begriff der Software-Qualität vermittelt und am Beispiel anschaulich gemacht. In der Übung wird ein großes Softwaresystem bearbeitet.		
14. Literatur:	<ul style="list-style-type: none"> <li>Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2007</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>165201 Vorlesung Software-Qualität</li> <li>165202 Übung Software-Qualität</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	16521 Software-Qualität (USL), Schriftlich, Gewichtung: 1 Übungsschein, Scheinkriterien werden zu Beginn der Veranstaltung angekündigt.		
18. Grundlage für ... :	Einführung in die Softwaretechnik		
19. Medienform:			
20. Angeboten von:	Software Engineering		

## Modul: 17210 Einführung in die Softwaretechnik

2. Modulkürzel:	051520015	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Stefan Wagner		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 2. Semester → Basismodule B.Sc. Softwaretechnik, PO 123-2017, 2. Semester → Basismodule		
11. Empfohlene Voraussetzungen:	- Modul 10280 Programmierung und Software-Entwicklung - Modul 12060 Datenstrukturen und Algorithmen - sowie entsprechende Programmiererfahrung		
12. Lernziele:	Die Veranstaltung liefert einen ersten Einblick in die Softwaretechnik. Sie ist abgestimmt auf die Software-Qualität im 1. und Programmentwicklung im 3. Semester.  Die Teilnehmer kennen die Grundbegriffe der Softwaretechnik und haben wichtige Techniken des Softwareprojekt-Managements und der Software-Entwicklung erlernt. Sie kennen Scrum als eine konkrete Vorgehensweise zur Softwareentwicklung		
13. Inhalt:	Die Vorlesung behandelt technische und andere Aspekte der Softwarebearbeitung, wie sie in der Praxis stattfindet. Die einzelnen Themen sind: <ul style="list-style-type: none"> <li>• Abgrenzung und Motivation des Software Engineerings</li> <li>• Vorgehensmodelle, agiles Vorgehen, Scrum</li> <li>• Software-Management</li> <li>• Software-Prüfung und Qualitätssicherung</li> <li>• Methoden, Sprachen und Werkzeuge für die einzelnen Phasen: Spezifikation, Grobentwurf, Feinentwurf, Implementierung, Test</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Ludewig, Lichter: Software Engineering. dpunkt-Verlag, Heidelberg. 2. Aufl. 2010</li> <li>• Pfleeger, Atlee: Software Engineering. Pearson, 2010</li> <li>• Rubin: Essential Scrum. Addison-Wesley, 2013</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 172102 Übung Einführung in die Softwaretechnik</li> <li>• 172101 Vorlesung Einführung in die Softwaretechnik</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 17211 Einführung in die Softwaretechnik (PL), Schriftlich, 60 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich, 30 Min. [17211] Einführung in die Softwaretechnik (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0, [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich, Hausaufgaben</li> </ul>		
18. Grundlage für ... :	- Modul Software Engineering - Modul Software-Praktikum		

19. Medienform:

- Folien am Beamer unterstützt durch Tafel und Overhead
- Dokumente, Links und Diskussionsforum in ILIAS

---

20. Angeboten von: Software Engineering

---

## 200 Kernmodule

---

Zugeordnete Module:	11890	Algorithmen und Berechenbarkeit
	14360	Einführung in die Technische Informatik
	14370	Fachstudie Softwaretechnik
	14390	Programmentwicklung
	14480	Sichere und zuverlässige Softwaresysteme
	16500	Software Engineering
	16510	Software-Praktikum

---

## Modul: 11890 Algorithmen und Berechenbarkeit

2. Modulkürzel:	050420020	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Funke		
9. Dozenten:	Ulrich Hertrampf Volker Diekert Stefan Funke		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	Vorlesungen aus dem 1. und 2. Semester		
12. Lernziele:	Die Studierenden kennen die Klassifizierung von Algorithmen in effizient berechenbar, NP-vollständig, PSPACE-Algorithmen und prinzipielle Unberechenbarkeit. Sie haben wichtige Entwurfsstrategien und Analysemethoden kennengelernt.		
13. Inhalt:	Berechenbarkeit vs. Unberechenbarkeit, Church'sche These, NP-Vollständigkeit, PSPACE-vollständige Algorithmen (QBF). Entwurfsstrategien: Teile und Herrsche, gierig (greedy), Dynamisches Programmieren, Randomisierte Algorithmen		
14. Literatur:	<ul style="list-style-type: none"> <li>• John Hopcroft, Jeffrey Ullman: Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988</li> <li>• Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Introduction to Algorithms (Second Edition), 2001</li> <li>• Volker Diekert: Entwurf und Analyse effizienter Algorithmen (Vorlesungsskript), 2006</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 118901 Vorlesung Algorithmen und Berechenbarkeit</li> <li>• 118902 Übung Algorithmen und Berechenbarkeit</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 11891 Algorithmen und Berechenbarkeit (PL), Schriftlich, 120 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich, 30 Min. Prüfungsvorleistung: Übungsschein</li> </ul>		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Algorithmik		

## Modul: 14360 Einführung in die Technische Informatik

2. Modulkürzel:	051400105	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Univ.-Prof. Dr. Otto Eggenberger	
9. Dozenten:		Otto Eggenberger	
10. Zuordnung zum Curriculum in diesem Studiengang:		B.Sc. Softwaretechnik, PO 123-2017, → Pflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Kernmodule	
11. Empfohlene Voraussetzungen:		Keine	
12. Lernziele:		Der/die Studierende kennt die grundlegende Funktionsweise eines Computers, versteht die elektrotechnischen Grundlagen und Technologien und kann einfache digitale Schaltungen analysieren, entwerfen und optimieren.	
13. Inhalt:		Grundlegende Funktionsweise eines Computers <ul style="list-style-type: none"> <li>• Informationsdarstellung</li> <li>• Zahlendarstellung und Codes</li> <li>• Digitale Grundbausteine</li> <li>• Logische Funktionen, Speicherelemente</li> <li>• Befehlsausführung, Programmablauf</li> </ul> Elektrotechnische Grundlagen <ul style="list-style-type: none"> <li>• Physikalische Grundbegriffe</li> <li>• Elektrische Spannung, elektrischer Strom</li> <li>• Elektrische Netzwerke</li> <li>• Halbleiterbauelemente</li> <li>• Digitale Grundsaltungen</li> </ul> Digitale Schaltungen <ul style="list-style-type: none"> <li>• Schaltnetzwerke</li> <li>• Boolesche Algebra und Schaltalgebra</li> <li>• Darstellung und Minimierung von Schaltfunktionen</li> <li>• Rückkopplung, Zustandsbegriff</li> <li>• Automaten und sequentielle Netzwerke</li> <li>• Digitale Standardschaltungen</li> <li>• Entwurfsmethodik</li> </ul>	
14. Literatur:		<ul style="list-style-type: none"> <li>• Dirk W. Hoffman: Grundlagen der technischen Informatik, Hanser, 2007</li> <li>• Bernd Becker, Rolf Drechsler, Paul Molitor: Technische Informatik, Pearson Studium, 2005</li> <li>• Jörg Keller, Wolfgang J. Paul: Hardware Design, Teubner, 3. Aufl. 2005</li> </ul>	
15. Lehrveranstaltungen und -formen:		<ul style="list-style-type: none"> <li>• 143602 Gruppenübungen Einführung in die Technische Informatik</li> <li>• 143601 Vorlesung Einführung in die Technische Informatik</li> </ul>	
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:		14361 Einführung in die Technische Informatik (PL), Schriftlich, 60 Min., Gewichtung: 1	

[14361] Einführung in die Technische Informatik (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0

---

18. Grundlage für ... : Rechnerorganisation 1

---

19. Medienform:

---

20. Angeboten von: Grundlagen der Informatik

---

## Modul: 14370 Fachstudie Softwaretechnik

2. Modulkürzel:	051520185	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester/ Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	- Bestandene Prüfung "Software Engineering"		
12. Lernziele:	Die Teilnehmer sind in der Lage, eine konkrete praktische Frage der Softwaretechnik, beispielsweise über die anzuwendende Methode oder das geeignete Werkzeug, zu analysieren und zu entscheiden und ihre Entscheidung angemessen zu präsentieren. Die Arbeit erfolgt in Dreiergruppen.		
13. Inhalt:	Eine Gruppe analysiert eine (im Allgemeinen aus der Praxis kommende) Frage auf der Basis der Literatur und eigener Untersuchungen, auch Befragungen, und präsentiert ihre Empfehlung mündlich und in Form eines Berichts.		
14. Literatur:	wird vom Dozenten bekanntgegeben		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 143701 Praktikum Fachstudie Softwaretechnik</li> <li>• 143702 Teamarbeit an den beteiligten Instituten mit örtlicher fachlicher Betreuung</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h		
17. Prüfungsnummer/n und -name:	14371 Fachstudie Softwaretechnik (USL), Sonstige, Gewichtung: 1 [14371] Fachstudie Softwaretechnik (USL), schriftlich und mündlich		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Visualisierung und Interaktive Systeme		

## Modul: 14390 Programmentwicklung

2. Modulkürzel:	051520120	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Stefan Wagner Jan-Peter Ostberg		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Programmierung und Softwareentwicklung</li> <li>• Einführung in die Softwaretechnik</li> </ul>		
12. Lernziele:	Die Teilnehmer kennen die Prinzipien der objektorientierten Programmierung und sind in der Lage, Programme in UML zu beschreiben und in Java zu implementieren.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Grundlagen der objektorientierten Programmierung</li> <li>• Spezifikation und Entwurf objektorientierter Programme mit UML</li> <li>• Vertiefte Programmierung in Java</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Rumbaugh, Jacobson, Booch, The unified modeling language reference manual, 2nd ed., 2004</li> <li>• Rupp, Queins, Zengler, UML 2 glasklar: Praxiswissen für die UML-Modellierung, 3. Aufl. 2007</li> <li>• Ullenboom: Java ist auch eine Insel. Galileo Computing, 8. Aufl. 2009</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 143901 Vorlesung Programmentwicklung</li> <li>• 143902 Übung Programmentwicklung</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	14391 Programmentwicklung (PL), Schriftlich, 60 Min., Gewichtung: 1 Klausur 60 min, keine Vorleistungen.		
18. Grundlage für ... :			
19. Medienform:	<ul style="list-style-type: none"> <li>• Folien am Beamer unterstützt durch Tafel und Overhead</li> <li>• Dokumente, Links und Diskussionsforen in ILIAS</li> </ul>		
20. Angeboten von:	Software Engineering		

## Modul: 14480 Sichere und zuverlässige Softwaresysteme

2. Modulkürzel:	051520115	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Steffen Becker		
9. Dozenten:	Steffen Becker André van Hoorn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Vorgezogene Master-Module B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	Erfolgreiche Teilnahme an der Lehrveranstaltung Software Engineering (für Softwaretechniker)		
12. Lernziele:	Nach der erfolgreichen Teilnahme sind die Teilnehmer in der Lage, die behandelten Methoden zur Theorie, Modellierung und Bewertung der Vertrauenswürdigkeit von Softwaresystemen zu benennen und zu erklären sowie basierend auf einer praktischen Fragestellung auszuwählen und anzuwenden.		
13. Inhalt:	In der Veranstaltung werden Grundlagen, Notationen und Verfahren zur Modellierung und Bewertung der Vertrauenswürdigkeit von Softwaresystemen sowie deren Einsatz in der ingenieurmäßigen Softwareentwicklung behandelt. Der Schwerpunkt liegt auf den Qualitätsattributen Performance, Zuverlässigkeit und Sicherheit (Safety, Security).		
14. Literatur:	<ul style="list-style-type: none"> <li>• K. S. Trivedi, A. Bobbio. Reliability and Availability Engineering Modeling, Analysis, and Applications. Wiley, 2017</li> <li>• A. Birolini. Reliability Engineering: Theory and Practice. Springer, 2010</li> <li>• A. Bondi. Foundations of Software and System Performance Engineering: Process, Performance Modeling, Requirements, Testing, Scalability, and Practice. Addison-Wesley. 2014</li> <li>• N. G. Leveson. Safeware: System Safety and Computers. Addison-Wesley, 1995</li> </ul> <p>Weitere Literatur wird in der Vorlesung bereitgestellt.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 144801 Vorlesung Sichere und zuverlässige Softwaresysteme</li> <li>• 144802 Übung Sichere und zuverlässige Softwaresysteme</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	14481 Sichere und zuverlässige Softwaresysteme (PL), Schriftlich oder Mündlich, 90 Min., Gewichtung: 1 Prüfungsleistung(PL), Schriftlich oder Mündlich, 90Min.		
18. Grundlage für ... :			
19. Medienform:			

20. Angeboten von:

Zuverlässige Software-Systeme

---

## Modul: 16500 Software Engineering

2. Modulkürzel:	051520110	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Steffen Becker		
9. Dozenten:	Steffen Becker André van Hoorn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Kernmodule B.Sc. Softwaretechnik, PO 123-2017, 4. Semester → Pflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2017, 4. Semester → Vorgezogene Master-Module		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Einführung in die Softwaretechnik</li> <li>• Programmierung und Softwareentwicklung</li> </ul>		
12. Lernziele:	Die Teilnehmer haben tiefe und umfassende Kenntnisse auf dem Gebiet des Software Engineerings.		
13. Inhalt:	<p>Ergänzend zur Einführung in die Softwaretechnik und daran anknüpfend, behandelt diese Lehrveranstaltung folgende Themen:</p> <ul style="list-style-type: none"> <li>• Organisationsaspekte des Software Engineering</li> <li>• Softwareentwicklungsprozesse, Prozessbewertung und -verbesserung</li> <li>• Anforderungsanalyse</li> <li>• Softwarearchitektur</li> <li>• Realisierung und Debugging</li> <li>• Softwarequalitätssicherung</li> <li>• Softwarewartung</li> <li>• Model-Driven Software Development</li> <li>• Weitere ausgewählte Kapitel des Software Engineerings</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Summerville, Software Engineering, AW</li> <li>• Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken, 2. Aufl. 2010</li> <li>• Liggesmeyer P., Software-Qualität. Testen, Analysieren und Verifizieren von Software. Spektrum Akademischer Verlag, 2002.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 165001 Vorlesung Software Engineering</li> <li>• 165002 Übung Software Engineering</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Vorlesungen und Zentralübungen		
17. Prüfungsnummer/n und -name:	16501 Software Engineering (PL), Schriftlich, 90 Min., Gewichtung: 1 90min Klausur		
18. Grundlage für ... :	Requirements Engineering and Software Architecture Model-Driven Software Development Software Qualität und -Wartung		
19. Medienform:	Folien, Videoaufzeichnung, Übungsblätter, elektronische Umfragen und -Tests		
20. Angeboten von:	Zuverlässige Software-Systeme		

## Modul: 16510 Software-Praktikum

2. Modulkürzel:	051520180	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Ivan Bogicevic Stefan Wagner Jan-Peter Ostberg Jasmin Ramadani		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Pflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Einführung in die Softwaretechnik</li> <li>• Gleichzeitiger Besuch der Programmentwicklung</li> </ul>		
12. Lernziele:	Die Teilnehmer können eine Software-Entwicklung von der Spezifikation bis zur Auslieferung durchführen.		
13. Inhalt:	Die Teilnehmer bearbeiten in Dreiergruppen eine zentral gestellte Aufgabe. Sie erheben dazu die notwendigen Informationen, erstellen die notwendigen Dokumente und implementieren und prüfen ein Programm, das die Aufgabe löst.		
14. Literatur:	<ul style="list-style-type: none"> <li>• Ludewig J., Lichter, H., Software Engineering - Grundlagen, Menschen, Prozesse, Techniken. dpunkt-Verlag, Heidelberg, 2. Aufl. 2010</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 165101 Praktikum Software-Praktikum</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	16511 Software-Praktikum (USL), Sonstige, Gewichtung: 1		
18. Grundlage für ... :	Studienprojekt-Pr Studienprojekt-Th		
19. Medienform:	Die meisten Dokumente erarbeiten die Studierenden selbst und stellen sie auch vor. Zusatzinformationen und Diskussionsforen werden in ILIAS zur Verfügung gestellt.		
20. Angeboten von:	Software Engineering		

## 210 Kernmodul Studienprojekt

---

---

## 300 Ergänzungsmodule

---

Zugeordnete Module:	310	Katalog SWT
	320	Katalog ISG
	330	Katalog ISW

---

## 310 Katalog SWT

---

Zugeordnete Module:    10210 Mensch-Computer-Interaktion  
                              10220 Modellierung  
                              36530 Rechnerorganisation 1  
                              40090 Systemkonzepte und -programmierung  
                              41590 Einführung in die Numerik und Stochastik für Softwaretechniker

---

## Modul: 10210 Mensch-Computer-Interaktion

2. Modulkürzel:	051900001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Jun.-Prof. Dr. Niels Henze		
9. Dozenten:	Niels Henze Tonja Machulla		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	Modul 10280 Programmierung und Software-Entwicklung		
12. Lernziele:	Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.		
13. Inhalt:	<p>Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet. Die folgenden Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> <li>• Einführung in die Grundlagen der Mensch-Computer Interaktion, historische Entwicklung</li> <li>• Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme</li> <li>• Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers</li> <li>• Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides</li> <li>• Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme</li> <li>• Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen</li> <li>• Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge</li> <li>• Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten</li> <li>• Akzeptanz, Evaluationsmethoden und Qualitätssicherung</li> </ul>		

14. Literatur:	<ul style="list-style-type: none"><li>• Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin, 2. Auflage. 2010</li><li>• Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004</li><li>• Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102101 Vorlesung Mensch-Computer-Interaktion</li><li>• 102102 Übung Mensch-Computer-Interaktion</li></ul>
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10211 Mensch-Computer-Interaktion (PL), Schriftlich, 90 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li></ul> Prüfungsvorleistung: Übungsschein
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Soziokognitive Systeme

---

## Modul: 10220 Modellierung

2. Modulkürzel:	052010001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Frank Leymann		
9. Dozenten:	Bernhard Mitschang Frank Leymann Uwe Breitenbücher		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10280 Programmierung und Software-Entwicklung</li> <li>• Modul 12060 Datenstrukturen und Algorithmen</li> <li>• Modul 40090 Systemkonzepte und -programmierung</li> </ul>		
12. Lernziele:	<p>Am Ende des Moduls sind die Studierenden in der Lage, wesentliche Artefakte eines IT Systems zu modellieren. Der Zusammenhang und das Zusammenspiel solcher Artefakte ist verstanden. Die Rolle von Metamodellen und deren Erstellung ist klar.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Entity-Relationship Modell und komplexe Objekte</li> <li>• Relationenmodell und Relationenalgebra , Überblick SQL - Transformationen von ER nach Relationen, Normalisierung</li> <li>• XML, DTD, XML-Schema, Info-Set, Namensräume</li> <li>• Metamodelle und Repository - RDF, RDF-S und Ontologien</li> <li>• UML</li> <li>• Petri Netze, Workflownetze</li> <li>• BPMN</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002.</li> <li>• R. Eckstein, S. Eckstein, XML und Datenmodellierung , dpunkt.verlag 2004.</li> <li>• M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work</li> <li>• Objektorientierte Modellierung mit UML2, 2005.</li> <li>• P. Hitzler, M. Krötzsch, S. Rudolph, Y. Sure, Semantic Web, 2008.</li> <li>• T.J. Teorey, Database Modeling und Design, 2nd Edition, 1994.</li> <li>• H.J. Habermann, F. Leymann, Repository , Oldenbourg 1993.</li> <li>• W. Reisig, Petri-Netze , Vieweg und Teubner 2010.</li> </ul>		

- B. Silver, BPMN Method und Style ,Cody-Cassidy Press 2009.
- 

15. Lehrveranstaltungen und -formen:
- 102201 Vorlesung Modellierung
  - 102202 Übung Modellierung
- 

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:
- 10221 Modellierung (PL), Schriftlich, 90 Min., Gewichtung: 1
  - V Vorleistung (USL-V), Schriftlich oder Mündlich
- [10221] Modellierung (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0, Prüfungsvorleistung: Übungsschein [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich
- 

18. Grundlage für ... :
- Architektur von Anwendungssystemen Datenbanken und Informationssysteme
- 

19. Medienform:

---

20. Angeboten von:
- Architektur von Anwendungssystemen
-

## Modul: 36530 Rechnerorganisation 1

2. Modulkürzel:	051700006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	Einführung in die Technische Informatik (14360) oder Technische Grundlagen der Informatik (78650)		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung</li> <li>• Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen</li> </ul>		
13. Inhalt:	<p>Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Labor-Übungen vertieft. Im Einzelnen werden behandelt:</p> <ul style="list-style-type: none"> <li>• Grundstrukturen: Logik Multiplexer, Speicher, Addierer, Multiplizierer, ALU</li> <li>• Zahlensysteme; Typ Integer, Typ Float, Gleitkommazahlen</li> <li>• Informationsdarstellung in Hardware: Daten- und Befehlsformate, Befehlssätze</li> <li>• Instruction Set Architecture</li> <li>• Assemblerprogrammierung</li> <li>• Grundzüge einer Hardware-Beschreibungssprache</li> <li>• RISC und CISC Architekturen</li> <li>• Steuerwerksentwurf und Mikroprogrammierung</li> <li>• Pipelining und Hazards</li> <li>• Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung</li> <li>• Multi-Core Prozessoren</li> <li>• GPU-Architekturen und Programmierung</li> <li>• FPGA-Architekturen</li> </ul>		
14. Literatur:	Literatur, siehe Veranstaltungshinweise		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 365301 Vorlesung Rechnerorganisation 1</li> <li>• 365302 Praktische und theoretische Übungen Rechnerorganisation 1</li> </ul>		
16. Abschätzung Arbeitsaufwand:			



## Modul: 40090 Systemkonzepte und -programmierung

2. Modulkürzel:	051200005	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel Frank Dürr		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 3. Semester → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10280 Programmierung und Software-Entwicklung</li> <li>• Modul 12060 Datenstrukturen und Algorithmen</li> </ul>		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen</li> <li>• Verstehen systemnaher Konzepte und Mechanismen</li> <li>• Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.</li> <li>• Kann systemnahe Software entwerfen und implementieren.</li> <li>• Kann nebenläufige Programme entwickeln</li> <li>• Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.</li> </ul>		
13. Inhalt:	<p>Grundlegende Systemstrukturen - und organisationen</p> <ul style="list-style-type: none"> <li>• Multitaskingsystem</li> <li>• Multiprozessorsystem</li> <li>• Verteiltes System Modellierung und Analyse nebenläufiger Programme</li> <li>• Abstraktionen: Atomare Befehle, Prozesse, nebenläufiges Programm</li> <li>• Korrektheit- und Leitungskriterien Betriebssystemkonzepte</li> <li>• Organisation von Betriebssystemen</li> <li>• Prozesse und Threads</li> <li>• Eingabe/Ausgabe</li> <li>• Scheduling Konzepte zur Synchronisation über gemeinsamen Speicher</li> <li>• Synchronisationsprobleme und -lösungen</li> <li>• Synchronisationswerkzeuge: Semaphore, Monitor Konzepte zur Kommunikation und Synchronisation mittels Nachrichtentransfer</li> <li>• Taxonomie: Kommunikation und Synchronisation</li> <li>• Nachrichten als Kommunikationskonzept</li> <li>• Höhere Kommunikationskonzepte Basisalgorithmen für Verteilte Systeme</li> <li>• Erkennung globaler Eigenschaften</li> </ul>		

- Schnappschussproblem
  - Konsistenter globaler Zustand
  - Verteilte Terminierung Praktische nebenläufige Programmierung in Java
  - Threads und Synchronisation
  - Socketschnittstelle
  - RMI Programmierung
- 

14. Literatur: Literatur, siehe Webseite zur Veranstaltung

---

15. Lehrveranstaltungen und -formen: 

- 400901 Vorlesung Systemkonzepte und -programmierung
- 400902 Übung Systemkonzepte und -programmierung

---

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name: 

- 40091 Systemkonzepte und -programmierung (PL), Schriftlich, 120 Min., Gewichtung: 1
- V Vorleistung (USL-V), Schriftlich oder Mündlich

[40091] Systemkonzepte und -programmierung (PL), schriftliche Prüfung, 120 Min., Gewicht: 1.0 [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Verteilte Systeme

---

## Modul: 41590 Einführung in die Numerik und Stochastik für Softwaretechniker

2. Modulkürzel:	051240006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Dirk Pflüger		
9. Dozenten:	Thomas Ertl Daniel Weiskopf Miriam Mehl Stefan Zimmer Dirk Pflüger		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Zusatzmodule		
11. Empfohlene Voraussetzungen:	Mathematik für Informatiker und Softwaretechniker (Modulnummer 10190)		
12. Lernziele:	Beherrschung grundlegender Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der erlernten Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.		
13. Inhalt:	Methoden der angewandten Mathematik, insbesondere der Numerik und Stochastik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse: <ul style="list-style-type: none"> <li>• numerische Algorithmik</li> <li>• Gleitpunktzahlen und Gleitpunktarithmetik</li> <li>• Interpolation und Approximation</li> <li>• Integration</li> <li>• lineare Gleichungssysteme</li> <li>• Iterative Lösung linearer und nichtlinearer Gleichungen</li> <li>• gewöhnliche Differentialgleichungen</li> <li>• Stochastik</li> <li>• Zufall und Unsicherheit</li> <li>• diskrete und kontinuierliche Wahrscheinlichkeitsräume</li> <li>• Asymptotik</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Dahmen, Reusken: Numerik für Ingenieure</li> <li>• Schwarz, Köckler: Numerische Mathematik</li> <li>• Huckle, Schneider: Numerik für Informatiker</li> <li>• Henze: Stochastik für Einsteiger</li> <li>• Schickinger, Steger: Diskrete Strukturen, Band 2</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 415901 Vorlesung Einführung in die Numerik und Stochastik für Softwaretechniker</li> <li>• 415902 Übung Einführung in die Numerik und Stochastik für Softwaretechniker</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h		

Eigenstudiumstunden: 138 h

Gesamtstunden: 180 h

---

17. Prüfungsnummer/n und -name: 41591 Einführung in die Numerik und Stochastik für  
Softwaretechniker (PL), Schriftlich, 90 Min., Gewichtung: 1  
[41591] Einführung in die Numerik und Stochastik für  
Softwaretechniker (PL), schriftlich, eventuell mündlich, 90 Min.,  
Gewicht: 1.0

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Simulation Software Engineering

---

## 320 Katalog ISG

---

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10060	Computergraphik
	10110	Grundlagen der Künstlichen Intelligenz
	10170	Imaging Science
	10180	Information Retrieval and Text Mining
	10210	Mensch-Computer-Interaktion
	10220	Modellierung
	10330	Systemkonzepte und -programmierung
	36530	Rechnerorganisation 1
	39040	Rechnernetze
	41590	Einführung in die Numerik und Stochastik für Softwaretechniker
	42410	Grundlagen des Wissenschaftlichen Rechnens

---

## Modul: 10030 Architektur von Anwendungssystemen

2. Modulkürzel:	052010002	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	Die Vorlesung erläutert den Begriff der Architektur von Anwendungssystemen und die Rolle des Architekten solcher Systeme. Die wesentlichen Bestandteile von Anwendungsarchitektur wie etwa Datenbanksysteme, Anwendungsserver, Messaging Systeme, Workflowsysteme und TP-Monitore werden diskutiert. Die wesentlichen Mustern zur Erstellung von Anwendungssystemen sind verstanden.		
13. Inhalt:	Architekturelle Stile wie etwa N-stufige Aufbauten oder Service-Orientierung werden vorgestellt. Architekturmuster werden detailliert. Fundamentale Konzepte wie Transaktionen und Queuing werden eingeführt. Darauf aufbauend wird Direct TP vs Queues TP diskutiert. Grundlegende Qualitätseigenschaften wie Verfügbarkeit und Skalierbarkeit werden erläutert und Mechanismen zu deren Erzielen eingeführt. Die Rolle von Komponenten und Programmierung im Großen wird herausgearbeitet und Modell-getriebene Architektur vorgestellt.		
14. Literatur:	<ul style="list-style-type: none"> <li>- A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002.</li> <li>- B. Neubauer, T. Ritter, F. Stoinnski, CORBA Komponenten, 2004.</li> <li>- F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, Pattern-orientierte Software Architektur</li> <li>- Ein Patternsystem, 1998.</li> <li>- F. Leymann, D. Roller, Production Workflow, 2000.</li> <li>- L. Hohmann, Beyond Software Architecture, 2003.</li> <li>- M. Fowler, Patterns of Enterprise Application Architecture, 2003.</li> <li>- P. Bernstein, E. Newcomer, Principles of Transaction Processing, 1997.</li> <li>- S. Conrad, W. Hasselbring, A. Koschel, R. Tritsch, Enterprise Application Integration, 2006.</li> <li>- S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005.</li> <li>- W. Emmerich, Konstruktion von verteilten Objekten, 2003.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 100301 Vorlesung Grundlagen der Architektur von Anwendungssystemen</li> <li>• 100302 Übung Grundlagen der Architektur von Anwendungssystemen</li> </ul>		

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name: 10031 Architektur von Anwendungssystemen (PL), Schriftlich, 90 Min., Gewichtung: 1  
[10031] Architektur von Anwendungssystemen (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0

---

18. Grundlage für ... : - Modul Loose Coupling and Message Based Applications - Modul Service Computing - Modul Business Process Management

---

19. Medienform: Vorlesungen mit begleitenden Übungen

---

20. Angeboten von: Architektur von Anwendungssystemen

---

## Modul: 10060 Computergraphik

2. Modulkürzel:	051900002	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Thomas Ertl		
9. Dozenten:	Thomas Ertl Daniel Weiskopf Michael Krone Guido Reina		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --> Wahlpflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10210 Mensch-Computer-Interaktion</li> <li>• Modul 41590 Einführung in die Numerik und Stochastik</li> </ul>		
12. Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.		
13. Inhalt:	Folgende Themen werden in der Vorlesung behandelt: <ul style="list-style-type: none"> <li>• Überblick über den Prozess der Bildsynthese</li> <li>• Graphische Geräte, visuelle Wahrnehmung, Farbsysteme</li> <li>• Grundlegende Rastergraphik und Bildverarbeitung</li> <li>• Raytracing und Beleuchtungsmodelle</li> <li>• 2D und 3D Geometrietransformationen, 3D Projektion</li> <li>• Graphikprogrammierung in OpenGL 3</li> <li>• Texturen</li> <li>• Polygonale und hierarchische Modelle</li> <li>• Rasterisierung und Verdeckungsrechnung</li> <li>• Grundlagen der geometrischen Modellierung (Kurven, Flächen)</li> <li>• Räumliche Datenstrukturen</li> </ul> Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.		
14. Literatur:	<ul style="list-style-type: none"> <li>• J. Encarnacao, W. Strasser, R. Klein: Graphische Datenverarbeitung (Band1 und 2), 1997</li> <li>• J. Foley, A. van Dam, S. Feiner, J. Hughes: Computer Graphics: Principle and Practice, 1990</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 100601 Vorlesung Computergraphik</li> <li>• 100602 Übung Computergraphik</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10061 Computergraphik (PL), Schriftlich, 60 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li> </ul> Prüfungsvorleistung: Übungsschein.		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Praktische Informatik (Dialogsysteme)

---

## Modul: 10110 Grundlagen der Künstlichen Intelligenz

2. Modulkürzel:	051900205	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Marc Toussaint		
9. Dozenten:	Daniel Hennes Marc Toussaint Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Modul 10190 Mathematik für Informatiker und Softwaretechniker		
12. Lernziele:	Der Student / die Studentin beherrscht die Grundlagen der Künstlichen Intelligenz, kann Probleme der KI selbständig einordnen und mit den erlernten Methoden und Algorithmen bearbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Intelligenz</li> <li>• Agentenbegriff</li> <li>• Problemlösen durch Suchen, Suchverfahren</li> <li>• Probleme mit Rand- und Nebenbedingungen</li> <li>• Spiele</li> <li>• Aussagen- und Prädikatenlogik</li> <li>• Logikbasierte Agenten, Wissensrepräsentation</li> <li>• Inferenz</li> <li>• Planen</li> <li>• Unsicherheit, probabilistisches Schließen</li> <li>• Probabilistisches Schließen über die Zeit</li> <li>• Entscheidungstheorie</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• S. Russell, P. Norvig, Künstliche Intelligenz: Ein Moderner Ansatz, 3. Aufl., 2012</li> <li>• S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, 2009</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101101 Vorlesung Grundlagen der Künstlichen Intelligenz</li> <li>• 101102 Übung Grundlagen der Künstlichen Intelligenz</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10111 Grundlagen der Künstlichen Intelligenz (PL), Schriftlich, 90 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li> </ul> <p>[10111] Grundlagen der Künstlichen Intelligenz (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0 Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich</p>		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Autonome Systeme

---

## Modul: 10170 Imaging Science

2. Modulkürzel:	051900210	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --> Wahlpflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Modul 10190 Mathematik für Informatiker und Softwaretechniker		
12. Lernziele:	Der Student / die Studentin beherrscht die Grundlagen der Repräsentation und Verarbeitung digitaler Bilder, kann Probleme aus dem Fachgebiet einordnen und selbständig mit den erlernten Algorithmen und Verfahren lösen.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Grundlagen aus der Optik:Lochkamera, Linsengleichung</li> <li>• Bildaufnahme:Kameras, Objektive, Beleuchtung, Aufnahmeprozess</li> <li>• Bildrepräsentation:Diskretisierung, Farbräume</li> <li>• Elementare Bildbearbeitung:Punktoperationen (z.B. Kontrastverstärkung, Binarisierung)</li> <li>• Lineare und nichtlineare Filter:Faltung, morphologische Operatoren</li> <li>• Fouriertransformation, Bilddarstellung und -bearbeitung im Fourierraum, Abtasttheorem</li> <li>• Orthogonale Transformationen:Cosinus, Wavelets</li> <li>• Kompression:Generische Verfahren (RLE, Entropie), spezielle Bildverfahren (z.B. jpeg)</li> <li>• Video:Formate, Kompression (z.B. MPEG)</li> <li>• Bildverbesserung und Restauration</li> <li>• Elementare Segmentierungsverfahren</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Bässmann, Henning, Kreys, Jutta, Bildverarbeitung Ad Oculos, 2004.</li> <li>• Forsyth, David and Ponce, Jean, Computer Vision. A Modern Approach, 2003.</li> <li>• Gonzalez, Rafael C., Woods, Richard E., Eddins, Steven L., Digital Image Processing, 2004.</li> <li>• Bigun, J.: Vision with Direction, 2006.</li> <li>• Klaus D. Tönnies, Grundlagen der Bildverarbeitung, 2005.</li> <li>• L. G. Shapiro, G. C. Stockman, Computer Vision, 2001.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101701 Vorlesung Imaging Science</li> <li>• 101702 Übung Imaging Science</li> </ul>		
16. Abschätzung Arbeitsaufwand:			

17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10171 Imaging Science (PL), Schriftlich oder Mündlich, 120 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich [10171] Imaging Science (PL), schriftlich, eventuell mündlich, 120 Min., Gewicht: 1.0, Prüfungsvorleistung (USL-V): Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben, schriftlich, eventuell mündlich</li></ul>
18. Grundlage für ... :	Modul Computer Vision Modul Correspondence Problems in Computer Vision
19. Medienform:	
20. Angeboten von:	Intelligente Systeme

## Modul: 10180 Information Retrieval and Text Mining

2. Modulkürzel:	052401010	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dr. Roman Klinger		
9. Dozenten:	Sebastian Pado Roman Klinger		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	Erfahrung mit Programmierung und Unix, erster Kontakt mit Verfahren des Maschinellen Lernens		
12. Lernziele:	Die Studierenden haben ein grundlegendes Verständnis der Konzepte und Algorithmen des Information Retrieval und Text Mining entwickelt.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Textpräprozessierung</li> <li>• invertierte Indexe</li> <li>• IR-Modelle (z.B. Vektorraum-basiertes IR)</li> <li>• Linkanalyse</li> <li>• Clustering</li> <li>• Frage-Antwort-Systeme</li> <li>• korpusbasierter Erwerb von lexikalischem und Weltwissen</li> </ul>		
14. Literatur:	- Chris Manning, Prabhakar Raghavan, Hinrich Schütze, Introduction to Information Retrieval, 2008 Cambridge University Press.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101801 Vorlesung Information Retrieval and Text Mining</li> <li>• 101802 Übung Information Retrieval and Text Mining</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10181 Information Retrieval and Text Mining (PL), Schriftlich, Gewichtung: 1</li> <li>• 10182 Information Retrieval und Text Mining - Hausübungen (USL), Sonstige, Gewichtung: 1</li> </ul> <p>[10181] Information Retrieval und Text Mining (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0 [10182] Information Retrieval und Text Mining - Hausübungen (USL), Sonstiges</p>		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Theoretische Computerlinguistik		

## Modul: 10210 Mensch-Computer-Interaktion

2. Modulkürzel:	051900001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Jun.-Prof. Dr. Niels Henze		
9. Dozenten:	Niels Henze Tonja Machulla		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	Modul 10280 Programmierung und Software-Entwicklung		
12. Lernziele:	Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.		
13. Inhalt:	<p>Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet. Die folgenden Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> <li>• Einführung in die Grundlagen der Mensch-Computer Interaktion, historische Entwicklung</li> <li>• Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme</li> <li>• Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers</li> <li>• Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides</li> <li>• Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme</li> <li>• Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen</li> <li>• Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge</li> <li>• Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten</li> <li>• Akzeptanz, Evaluationsmethoden und Qualitätssicherung</li> </ul>		

14. Literatur:	<ul style="list-style-type: none"><li>• Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin, 2. Auflage. 2010</li><li>• Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004</li><li>• Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102101 Vorlesung Mensch-Computer-Interaktion</li><li>• 102102 Übung Mensch-Computer-Interaktion</li></ul>
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10211 Mensch-Computer-Interaktion (PL), Schriftlich, 90 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li></ul> Prüfungsvorleistung: Übungsschein
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Soziokognitive Systeme

---

## Modul: 10220 Modellierung

2. Modulkürzel:	052010001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Frank Leymann		
9. Dozenten:	Bernhard Mitschang Frank Leymann Uwe Breitenbücher		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10280 Programmierung und Software-Entwicklung</li> <li>• Modul 12060 Datenstrukturen und Algorithmen</li> <li>• Modul 40090 Systemkonzepte und -programmierung</li> </ul>		
12. Lernziele:	<p>Am Ende des Moduls sind die Studierenden in der Lage, wesentliche Artefakte eines IT Systems zu modellieren. Der Zusammenhang und das Zusammenspiel solcher Artefakte ist verstanden. Die Rolle von Metamodellen und deren Erstellung ist klar.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Entity-Relationship Modell und komplexe Objekte</li> <li>• Relationenmodell und Relationenalgebra , Überblick SQL - Transformationen von ER nach Relationen, Normalisierung</li> <li>• XML, DTD, XML-Schema, Info-Set, Namensräume</li> <li>• Metamodelle und Repository - RDF, RDF-S und Ontologien</li> <li>• UML</li> <li>• Petri Netze, Workflownetze</li> <li>• BPMN</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002.</li> <li>• R. Eckstein, S. Eckstein, XML und Datenmodellierung , dpunkt.verlag 2004.</li> <li>• M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work</li> <li>• Objektorientierte Modellierung mit UML2, 2005.</li> <li>• P. Hitzler, M. Kröttsch, S. Rudolph, Y. Sure, Semantic Web, 2008.</li> <li>• T.J. Teorey, Database Modeling und Design, 2nd Edition, 1994.</li> <li>• H.J. Habermann, F. Leymann, Repository , Oldenbourg 1993.</li> <li>• W. Reisig, Petri-Netze , Vieweg und Teubner 2010.</li> </ul>		

- B. Silver, BPMN Method und Style ,Cody-Cassidy Press 2009.
- 

15. Lehrveranstaltungen und -formen:

- 102201 Vorlesung Modellierung
- 102202 Übung Modellierung

---

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:

- 10221 Modellierung (PL), Schriftlich, 90 Min., Gewichtung: 1
- V Vorleistung (USL-V), Schriftlich oder Mündlich

[10221] Modellierung (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0, Prüfungsvorleistung: Übungsschein [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich

---

18. Grundlage für ... :

Architektur von Anwendungssystemen Datenbanken und Informationssysteme

---

19. Medienform:

---

20. Angeboten von:

Architektur von Anwendungssystemen

---

## Modul: 10330 Systemkonzepte und -programmierung

2. Modulkürzel:	051200005	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel Frank Dürr		
10. Zuordnung zum Curriculum in diesem Studiengang:			
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 051520005 Programmierung und Software-Entwicklung</li> <li>• Modul 051510005 Datenstrukturen und Algorithmen</li> </ul>		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen</li> <li>• Verstehen systemnaher Konzepte und Mechanismen</li> <li>• Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.</li> <li>• Kann systemnahe Software entwerfen und implementieren.</li> <li>• Kann nebenläufige Programme entwickeln</li> <li>• Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.</li> </ul>		
13. Inhalt:	<p>Grundlegende Systemstrukturen - und organisationen            Multitaskingsystem            Multiprozessorsystem            Verteiltes System            Modellierung und Analyse nebenläufiger Programme            Abstraktionen: Atomare Befehle, Prozesse, nebenläufiges Programm            Korrektheit- und Leitungskriterien            Betriebssystemkonzepte            Organisation von Betriebssystemen            Prozesse und Threads            Eingabe/Ausgabe            Scheduling            Konzepte zur Synchronisation über gemeinsamen Speicher            Synchronisationsprobleme und -lösungen            Synchronisationswerkzeuge: Semaphor, Monitor            Konzepte zur Kommunikation und Synchronisation mittels Nachrichtentransfer            Taxonomie: Kommunikation und Synchronisation            Nachrichten als Kommunikationskonzept            Höhere Kommunikationskonzepte            Basisalgorithmen für Verteilte Systeme            Erkennung globaler Eigenschaften            Schnapsschussproblem            Konsistenter globaler Zustand            Verteilte Terminierung            Praktische nebenläufige Programmierung in Java            Threads und Synchronisation            Socketschnittstelle            RMI Programmierung</p>		

14. Literatur:	<ul style="list-style-type: none"><li>• Literatur, siehe Webseite zur Veranstaltung</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 103301 Vorlesung Systemkonzepte und -programmierung</li><li>• 103302 Übung Systemkonzepte und -programmierung</li></ul>
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10331 Systemkonzepte und -programmierung (PL), Schriftlich, 120 Min., Gewichtung: 7</li><li>• 10332 Systemkonzepte und -programmierung - Übungsschein (LBP), Schriftlich, 30 Min., Gewichtung: 3</li></ul>
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Verteilte Systeme

---

## Modul: 36530 Rechnerorganisation 1

2. Modulkürzel:	051700006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	Einführung in die Technische Informatik (14360) oder Technische Grundlagen der Informatik (78650)		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung</li> <li>• Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen</li> </ul>		
13. Inhalt:	<p>Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Labor-Übungen vertieft. Im Einzelnen werden behandelt:</p> <ul style="list-style-type: none"> <li>• Grundstrukturen: Logik Multiplexer, Speicher, Addierer, Multiplizierer, ALU</li> <li>• Zahlensysteme; Typ Integer, Typ Float, Gleitkommazahlen</li> <li>• Informationsdarstellung in Hardware: Daten- und Befehlsformate, Befehlssätze</li> <li>• Instruction Set Architecture</li> <li>• Assemblerprogrammierung</li> <li>• Grundzüge einer Hardware-Beschreibungssprache</li> <li>• RISC und CISC Architekturen</li> <li>• Steuerwerksentwurf und Mikroprogrammierung</li> <li>• Pipelining und Hazards</li> <li>• Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung</li> <li>• Multi-Core Prozessoren</li> <li>• GPU-Architekturen und Programmierung</li> <li>• FPGA-Architekturen</li> </ul>		
14. Literatur:	Literatur, siehe Veranstaltungshinweise		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 365301 Vorlesung Rechnerorganisation 1</li> <li>• 365302 Praktische und theoretische Übungen Rechnerorganisation 1</li> </ul>		
16. Abschätzung Arbeitsaufwand:			



## Modul: 39040 Rechnernetze

2. Modulkürzel:	051200010	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel Frank Dürr		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Programmierung und Software-Entwicklung - Datenstrukturen und Algorithmen - Grundkenntnisse in Java		
12. Lernziele:	- Versteht grundsätzliche Eigenschaften, Konzepte und Methoden von Rechnernetzen, insbesondere dem Internet. - Versteht Schichten und deren Zusammenwirken in einem Protokollstapel - Kann Rechnernetze aufbauen, verwalten und analysieren. - Kann Protokolle entwickeln und in Schichtenarchitektur einbetten. - Kann höhere Kommunikationsdienste zur Entwicklung von netzgestützten Systemen anwenden. - Kann sich mit Experten anderer Domänen über Methoden der Rechnernetze verständigen.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Einführung in die Rechnernetze, ISO Referenzmodell,</li> <li>• Bitübertragungsschicht: Übertragungsmedien, analoge und digitale Informationskodierung und -übertragung, Vermittlungsarten,</li> <li>• Sicherungsschicht: Betriebsarten, Fehlererkennung und -behandlung, Flusskontrolle,</li> <li>• Lokale Netze: CSMA/CD, Token Ring, Token Bus, FDDI, Kopplung,</li> <li>• Vermittlungsschicht: Verbindungsorientierter und verbindungsloser Dienst, Leitwegbestimmung, Überlastkontrolle,</li> <li>• Internetworking,</li> <li>• Internet-Protokoll,</li> <li>• Transportschicht: ausgewählte Realisierungsprobleme und Internet-Protokolle,</li> <li>• Echtzeitkommunikation: IntServ, DiffServ, Sicherheit: Verfahren, IPsec, SSL, TLS.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A.S. Tanenbaum, Computer Networks, 4th Edition, 2003 - D.E. Comer, Computernetzwerke und Internets, 2000</li> <li>• D.E. Comer, Internetworking with TCP/IP Volume I: Principles, Protocols, and Architecture, 1995</li> <li>• J. F. Kurose, K. W. Ross, Computer Networks: a top-down approach featuring the Internet, 2001</li> </ul>		

- L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach, 1999
- 

15. Lehrveranstaltungen und -formen:
- 390401 VL Rechnernetze
  - 390402 ÜB Rechnernetze
- 

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:
- 39041 Rechnernetze (PL), Schriftlich oder Mündlich, 90 Min., Gewichtung: 1
  - V Vorleistung (USL-V), Schriftlich oder Mündlich  
Prüfungsdauer: 90 min schriftlich oder 30 min mündlich
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Verteilte Systeme

---

## Modul: 41590 Einführung in die Numerik und Stochastik für Softwaretechniker

2. Modulkürzel:	051240006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Dirk Pflüger		
9. Dozenten:	Thomas Ertl Daniel Weiskopf Miriam Mehl Stefan Zimmer Dirk Pflüger		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester          → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester          → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester          → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester          → Zusatzmodule</p>		
11. Empfohlene Voraussetzungen:	Mathematik für Informatiker und Softwaretechniker (Modulnummer 10190)		
12. Lernziele:	Beherrschung grundlegender Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der erlernten Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.		
13. Inhalt:	<p>Methoden der angewandten Mathematik, insbesondere der Numerik und Stochastik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse:</p> <ul style="list-style-type: none"> <li>• numerische Algorithmik</li> <li>• Gleitpunktzahlen und Gleitpunktarithmetik</li> <li>• Interpolation und Approximation</li> <li>• Integration</li> <li>• lineare Gleichungssysteme</li> <li>• Iterative Lösung linearer und nichtlinearer Gleichungen</li> <li>• gewöhnliche Differentialgleichungen</li> <li>• Stochastik</li> <li>• Zufall und Unsicherheit</li> <li>• diskrete und kontinuierliche Wahrscheinlichkeitsräume</li> <li>• Asymptotik</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Dahmen, Reusken: Numerik für Ingenieure</li> <li>• Schwarz, Köckler: Numerische Mathematik</li> <li>• Huckle, Schneider: Numerik für Informatiker</li> <li>• Henze: Stochastik für Einsteiger</li> <li>• Schickinger, Steger: Diskrete Strukturen, Band 2</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 415901 Vorlesung Einführung in die Numerik und Stochastik für Softwaretechniker</li> <li>• 415902 Übung Einführung in die Numerik und Stochastik für Softwaretechniker</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h		

Eigenstudiumstunden: 138 h

Gesamtstunden: 180 h

---

17. Prüfungsnummer/n und -name: 41591 Einführung in die Numerik und Stochastik für  
Softwaretechniker (PL), Schriftlich, 90 Min., Gewichtung: 1  
[41591] Einführung in die Numerik und Stochastik für  
Softwaretechniker (PL), schriftlich, eventuell mündlich, 90 Min.,  
Gewicht: 1.0

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Simulation Software Engineering

---

## Modul: 42410 Grundlagen des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240020	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Dirk Pflüger		
9. Dozenten:	Miriam Mehl Stefan Zimmer Dirk Pflüger		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>- Modul 10190 Mathematik für Informatiker und Softwaretechniker und</li> <li>- Modul 10240 Numerische und Stochastische Grundlagen der Informatik bzw.</li> <li>- Modul 41590 Einführung in die Numerik und Stochastik für Softwaretechniker</li> </ul>		
12. Lernziele:	<p>Kenntnis grundlegender Konzepte, Algorithmen und Methoden des Wissenschaftlichen Rechnens. Fähigkeit, mit den erlernten Kenntnissen selbständig Methoden zu entwickeln, zu analysieren und umzusetzen, mit denen anwendungsorientierte Probleme effizient und genau gelöst werden können.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>- Überblick über die Simulationspipeline und die Wechselwirkungen zwischen den einzelnen Schritten</li> <li>- Skalenabhängige Modellierung</li> <li>- Diskretisierung (Gitter, Finite Elemente, Zeitschrittverfahren)</li> <li>- Algorithmen (Gittergenerierung, Adaptivität, Lineare Löser, Linked-Cell, Fast Multipole)</li> <li>- Parallelisierung (Gitterpartitionierung, Lastbalancierung)</li> <li>- Kurzer Überblick über die Visualisierung</li> </ul>		
14. Literatur:	<p>Martin Hanke-Bourgeois: Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens, Vieweg+Teubner Verlag 2009.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 424102 Übung Grundlagen des Wissenschaftlichen Rechnens</li> <li>• 424101 Vorlesung Grundlagen des Wissenschaftlichen Rechnens</li> </ul>		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h</p>		
17. Prüfungsnummer/n und -name:	<p>42411 Grundlagen des Wissenschaftlichen Rechnens (PL), Schriftlich oder Mündlich, 90 Min., Gewichtung: 1 [42411] Grundlagen des Wissenschaftlichen Rechnens (PL), schriftlich oder mündlich, 90 Min., Gewicht: 1.0</p>		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Simulation Software Engineering

---

## 330 Katalog ISW

---

Zugeordnete Module:	10030	Architektur von Anwendungssystemen
	10060	Computergraphik
	10110	Grundlagen der Künstlichen Intelligenz
	10170	Imaging Science
	10180	Information Retrieval and Text Mining
	10210	Mensch-Computer-Interaktion
	10220	Modellierung
	36530	Rechnerorganisation 1
	39040	Rechnernetze
	40090	Systemkonzepte und -programmierung
	41590	Einführung in die Numerik und Stochastik für Softwaretechniker
	42410	Grundlagen des Wissenschaftlichen Rechnens
	71740	System and Web Security
	78900	Introduction to Modern Cryptography

---

## Modul: 10030 Architektur von Anwendungssystemen

2. Modulkürzel:	052010002	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Frank Leymann		
9. Dozenten:	Frank Leymann		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	Vorlesungen des Grundstudiums.		
12. Lernziele:	Die Vorlesung erläutert den Begriff der Architektur von Anwendungssystemen und die Rolle des Architekten solcher Systeme. Die wesentlichen Bestandteile von Anwendungsarchitektur wie etwa Datenbanksysteme, Anwendungsserver, Messaging Systeme, Workflowsysteme und TP-Monitore werden diskutiert. Die wesentlichen Mustern zur Erstellung von Anwendungssystemen sind verstanden.		
13. Inhalt:	Architekturelle Stile wie etwa N-stufige Aufbauten oder Service-Orientierung werden vorgestellt. Architekturmuster werden detailliert. Fundamentale Konzepte wie Transaktionen und Queuing werden eingeführt. Darauf aufbauend wird Direct TP vs Queues TP diskutiert. Grundlegende Qualitätseigenschaften wie Verfügbarkeit und Skalierbarkeit werden erläutert und Mechanismen zu deren Erzielen eingeführt. Die Rolle von Komponenten und Programmierung im Großen wird herausgearbeitet und Modell-getriebene Architektur vorgestellt.		
14. Literatur:	<ul style="list-style-type: none"> <li>- A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002.</li> <li>- B. Neubauer, T. Ritter, F. Stoinnski, CORBA Komponenten, 2004.</li> <li>- F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, Pattern-orientierte Software Architektur</li> <li>- Ein Patternsystem, 1998.</li> <li>- F. Leymann, D. Roller, Production Workflow, 2000.</li> <li>- L. Hohmann, Beyond Software Architecture, 2003.</li> <li>- M. Fowler, Patterns of Enterprise Application Architecture, 2003.</li> <li>- P. Bernstein, E. Newcomer, Principles of Transaction Processing, 1997.</li> <li>- S. Conrad, W. Hasselbring, A. Koschel, R. Tritsch, Enterprise Application Integration, 2006.</li> <li>- S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D. Ferguson, Web Services Platform Architecture, 2005.</li> <li>- W. Emmerich, Konstruktion von verteilten Objekten, 2003.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 100301 Vorlesung Grundlagen der Architektur von Anwendungssystemen</li> <li>• 100302 Übung Grundlagen der Architektur von Anwendungssystemen</li> </ul>		

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name: 10031 Architektur von Anwendungssystemen (PL), Schriftlich, 90 Min., Gewichtung: 1  
[10031] Architektur von Anwendungssystemen (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0

---

18. Grundlage für ... : - Modul Loose Coupling and Message Based Applications - Modul Service Computing - Modul Business Process Management

---

19. Medienform: Vorlesungen mit begleitenden Übungen

---

20. Angeboten von: Architektur von Anwendungssystemen

---

## Modul: 10060 Computergraphik

2. Modulkürzel:	051900002	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Thomas Ertl		
9. Dozenten:	Thomas Ertl Daniel Weiskopf Michael Krone Guido Reina		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --> Wahlpflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10210 Mensch-Computer-Interaktion</li> <li>• Modul 41590 Einführung in die Numerik und Stochastik</li> </ul>		
12. Lernziele:	Die Studierenden haben Wissen über die Grundlagen der Computergraphik sowie praktische Fähigkeiten in der Graphikprogrammierung erworben.		
13. Inhalt:	Folgende Themen werden in der Vorlesung behandelt: <ul style="list-style-type: none"> <li>• Überblick über den Prozess der Bildsynthese</li> <li>• Graphische Geräte, visuelle Wahrnehmung, Farbsysteme</li> <li>• Grundlegende Rastergraphik und Bildverarbeitung</li> <li>• Raytracing und Beleuchtungsmodelle</li> <li>• 2D und 3D Geometrietransformationen, 3D Projektion</li> <li>• Graphikprogrammierung in OpenGL 3</li> <li>• Texturen</li> <li>• Polygonale und hierarchische Modelle</li> <li>• Rasterisierung und Verdeckungsrechnung</li> <li>• Grundlagen der geometrischen Modellierung (Kurven, Flächen)</li> <li>• Räumliche Datenstrukturen</li> </ul> Die Veranstaltung besteht aus Vorlesung mit Übungen. Die Übungen umfassen praktische Programmierübungen, theoretische Themen und Programmierprojekte.		
14. Literatur:	<ul style="list-style-type: none"> <li>• J. Encarnacao, W. Strasser, R. Klein: Graphische Datenverarbeitung (Band1 und 2), 1997</li> <li>• J. Foley, A. van Dam, S. Feiner, J. Hughes: Computer Graphics: Principle and Practice, 1990</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 100601 Vorlesung Computergraphik</li> <li>• 100602 Übung Computergraphik</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10061 Computergraphik (PL), Schriftlich, 60 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li> </ul> Prüfungsvorleistung: Übungsschein.		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Praktische Informatik (Dialogsysteme)

---

## Modul: 10110 Grundlagen der Künstlichen Intelligenz

2. Modulkürzel:	051900205	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Marc Toussaint		
9. Dozenten:	Daniel Hennes Marc Toussaint Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Modul 10190 Mathematik für Informatiker und Softwaretechniker		
12. Lernziele:	Der Student / die Studentin beherrscht die Grundlagen der Künstlichen Intelligenz, kann Probleme der KI selbständig einordnen und mit den erlernten Methoden und Algorithmen bearbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Intelligenz</li> <li>• Agentenbegriff</li> <li>• Problemlösen durch Suchen, Suchverfahren</li> <li>• Probleme mit Rand- und Nebenbedingungen</li> <li>• Spiele</li> <li>• Aussagen- und Prädikatenlogik</li> <li>• Logikbasierte Agenten, Wissensrepräsentation</li> <li>• Inferenz</li> <li>• Planen</li> <li>• Unsicherheit, probabilistisches Schließen</li> <li>• Probabilistisches Schließen über die Zeit</li> <li>• Entscheidungstheorie</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• S. Russell, P. Norvig, Künstliche Intelligenz: Ein Moderner Ansatz, 3. Aufl., 2012</li> <li>• S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, 2009</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101101 Vorlesung Grundlagen der Künstlichen Intelligenz</li> <li>• 101102 Übung Grundlagen der Künstlichen Intelligenz</li> </ul>		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10111 Grundlagen der Künstlichen Intelligenz (PL), Schriftlich, 90 Min., Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li> </ul> <p>[10111] Grundlagen der Künstlichen Intelligenz (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0 Prüfungsvorleistung: Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich</p>		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Autonome Systeme

---

## Modul: 10170 Imaging Science

2. Modulkürzel:	051900210	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Andrés Bruhn		
9. Dozenten:	Andrés Bruhn		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --> Wahlpflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Modul 10190 Mathematik für Informatiker und Softwaretechniker		
12. Lernziele:	Der Student / die Studentin beherrscht die Grundlagen der Repräsentation und Verarbeitung digitaler Bilder, kann Probleme aus dem Fachgebiet einordnen und selbständig mit den erlernten Algorithmen und Verfahren lösen.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Grundlagen aus der Optik:Lochkamera, Linsengleichung</li> <li>• Bildaufnahme:Kameras, Objektive, Beleuchtung, Aufnahmeprozess</li> <li>• Bildrepräsentation:Diskretisierung, Farbräume</li> <li>• Elementare Bildbearbeitung:Punktoperationen (z.B. Kontrastverstärkung, Binarisierung)</li> <li>• Lineare und nichtlineare Filter:Faltung, morphologische Operatoren</li> <li>• Fouriertransformation, Bilddarstellung und -bearbeitung im Fourierraum, Abtasttheorem</li> <li>• Orthogonale Transformationen:Cosinus, Wavelets</li> <li>• Kompression:Generische Verfahren (RLE, Entropie), spezielle Bildverfahren (z.B. jpeg)</li> <li>• Video:Formate, Kompression (z.B. MPEG)</li> <li>• Bildverbesserung und Restauration</li> <li>• Elementare Segmentierungsverfahren</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• Bässmann, Henning, Kreys, Jutta, Bildverarbeitung Ad Oculos, 2004.</li> <li>• Forsyth, David and Ponce, Jean, Computer Vision. A Modern Approach, 2003.</li> <li>• Gonzalez, Rafael C., Woods, Richard E., Eddins, Steven L., Digital Image Processing, 2004.</li> <li>• Bigun, J.: Vision with Direction, 2006.</li> <li>• Klaus D. Tönnies, Grundlagen der Bildverarbeitung, 2005.</li> <li>• L. G. Shapiro, G. C. Stockman, Computer Vision, 2001.</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101701 Vorlesung Imaging Science</li> <li>• 101702 Übung Imaging Science</li> </ul>		
16. Abschätzung Arbeitsaufwand:			

17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10171 Imaging Science (PL), Schriftlich oder Mündlich, 120 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich [10171] Imaging Science (PL), schriftlich, eventuell mündlich, 120 Min., Gewicht: 1.0, Prüfungsvorleistung (USL-V): Übungsschein, Kriterien werden in der ersten Vorlesung bekannt gegeben, schriftlich, eventuell mündlich</li></ul>
18. Grundlage für ... :	Modul Computer Vision Modul Correspondence Problems in Computer Vision
19. Medienform:	
20. Angeboten von:	Intelligente Systeme

---

## Modul: 10180 Information Retrieval and Text Mining

2. Modulkürzel:	052401010	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Dr. Roman Klinger		
9. Dozenten:	Sebastian Pado Roman Klinger		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	Erfahrung mit Programmierung und Unix, erster Kontakt mit Verfahren des Maschinellen Lernens		
12. Lernziele:	Die Studierenden haben ein grundlegendes Verständnis der Konzepte und Algorithmen des Information Retrieval und Text Mining entwickelt.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Textpräprozessierung</li> <li>• invertierte Indexe</li> <li>• IR-Modelle (z.B. Vektorraum-basiertes IR)</li> <li>• Linkanalyse</li> <li>• Clustering</li> <li>• Frage-Antwort-Systeme</li> <li>• korpusbasierter Erwerb von lexikalischem und Weltwissen</li> </ul>		
14. Literatur:	- Chris Manning, Prabhakar Raghavan, Hinrich Schütze, Introduction to Information Retrieval, 2008 Cambridge University Press.		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 101801 Vorlesung Information Retrieval and Text Mining</li> <li>• 101802 Übung Information Retrieval and Text Mining</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 10181 Information Retrieval and Text Mining (PL), Schriftlich, Gewichtung: 1</li> <li>• 10182 Information Retrieval und Text Mining - Hausübungen (USL), Sonstige, Gewichtung: 1</li> </ul> [10181] Information Retrieval und Text Mining (PL), schriftliche Prüfung, 60 Min., Gewicht: 1.0 [10182] Information Retrieval und Text Mining - Hausübungen (USL), Sonstiges		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Theoretische Computerlinguistik		

## Modul: 10210 Mensch-Computer-Interaktion

2. Modulkürzel:	051900001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Jun.-Prof. Dr. Niels Henze		
9. Dozenten:	Niels Henze Tonja Machulla		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	Modul 10280 Programmierung und Software-Entwicklung		
12. Lernziele:	Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile.		
13. Inhalt:	<p>Die Vorlesung vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet. Die folgenden Themen werden in der Vorlesung behandelt:</p> <ul style="list-style-type: none"> <li>• Einführung in die Grundlagen der Mensch-Computer Interaktion, historische Entwicklung</li> <li>• Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme</li> <li>• Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers</li> <li>• Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides</li> <li>• Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme</li> <li>• Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen</li> <li>• Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge</li> <li>• Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten</li> <li>• Akzeptanz, Evaluationsmethoden und Qualitätssicherung</li> </ul>		

14. Literatur:	<ul style="list-style-type: none"><li>• Bernhard Preim, Raimund Dachsel. Interaktive Systeme 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer, Berlin, 2. Auflage. 2010</li><li>• Alan Dix, Janet Finley, Gregory Abowd, Russell Beale, Human-Computer Interaction, 2004</li><li>• Ben Shneiderman, Catherine Plaisant, Designing the User Interfaces, 2005</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102101 Vorlesung Mensch-Computer-Interaktion</li><li>• 102102 Übung Mensch-Computer-Interaktion</li></ul>
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10211 Mensch-Computer-Interaktion (PL), Schriftlich, 90 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li></ul> Prüfungsvorleistung: Übungsschein
18. Grundlage für ... :	
19. Medienform:	
20. Angeboten von:	Soziokognitive Systeme

---

## Modul: 10220 Modellierung

2. Modulkürzel:	052010001	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Frank Leymann		
9. Dozenten:	Bernhard Mitschang Frank Leymann Uwe Breitenbücher		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2012, 4. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Praktische Informatik --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10280 Programmierung und Software-Entwicklung</li> <li>• Modul 12060 Datenstrukturen und Algorithmen</li> <li>• Modul 40090 Systemkonzepte und -programmierung</li> </ul>		
12. Lernziele:	<p>Am Ende des Moduls sind die Studierenden in der Lage, wesentliche Artefakte eines IT Systems zu modellieren. Der Zusammenhang und das Zusammenspiel solcher Artefakte ist verstanden. Die Rolle von Metamodellen und deren Erstellung ist klar.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Entity-Relationship Modell und komplexe Objekte</li> <li>• Relationenmodell und Relationenalgebra , Überblick SQL - Transformationen von ER nach Relationen, Normalisierung</li> <li>• XML, DTD, XML-Schema, Info-Set, Namensräume</li> <li>• Metamodelle und Repository - RDF, RDF-S und Ontologien</li> <li>• UML</li> <li>• Petri Netze, Workflownetze</li> <li>• BPMN</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, 2002.</li> <li>• R. Eckstein, S. Eckstein, XML und Datenmodellierung , dpunkt.verlag 2004.</li> <li>• M. Hitz, G. Kappel, E. Kapsammer, W. Retschitzegger, UML @ Work</li> <li>• Objektorientierte Modellierung mit UML2, 2005.</li> <li>• P. Hitzler, M. Kröttsch, S. Rudolph, Y. Sure, Semantic Web, 2008.</li> <li>• T.J. Teorey, Database Modeling und Design, 2nd Edition, 1994.</li> <li>• H.J. Habermann, F. Leymann, Repository , Oldenbourg 1993.</li> <li>• W. Reisig, Petri-Netze , Vieweg und Teubner 2010.</li> </ul>		

	<ul style="list-style-type: none"><li>• B. Silver, BPMN Method und Style ,Cody-Cassidy Press 2009.</li></ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"><li>• 102201 Vorlesung Modellierung</li><li>• 102202 Übung Modellierung</li></ul>
16. Abschätzung Arbeitsaufwand:	
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"><li>• 10221 Modellierung (PL), Schriftlich, 90 Min., Gewichtung: 1</li><li>• V Vorleistung (USL-V), Schriftlich oder Mündlich</li></ul> [10221] Modellierung (PL), schriftliche Prüfung, 90 Min., Gewicht: 1.0, Prüfungsvorleistung: Übungsschein [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich
18. Grundlage für ... :	Architektur von Anwendungssystemen Datenbanken und Informationssysteme
19. Medienform:	
20. Angeboten von:	Architektur von Anwendungssystemen

---

## Modul: 36530 Rechnerorganisation 1

2. Modulkürzel:	051700006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr.-Ing. Sven Simon		
9. Dozenten:	Sven Simon		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Zusatzmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	Einführung in die Technische Informatik (14360) oder Technische Grundlagen der Informatik (78650)		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Kenntnisse über die Grundlagen der Organisation von Rechnern und deren maschinennahe Programmierung</li> <li>• Grundzüge über die Beschreibung und den Entwurf von Hardwaresystemen</li> </ul>		
13. Inhalt:	<p>Aufbau, Entwurf und maschinennahe Programmierung von Mikroprozessoren werden erläutert. In den Übungen wird das Wissen durch schriftliche Aufgaben sowie Labor-Übungen vertieft. Im Einzelnen werden behandelt:</p> <ul style="list-style-type: none"> <li>• Grundstrukturen: Logik Multiplexer, Speicher, Addierer, Multiplizierer, ALU</li> <li>• Zahlensysteme; Typ Integer, Typ Float, Gleitkommazahlen</li> <li>• Informationsdarstellung in Hardware: Daten- und Befehlsformate, Befehlssätze</li> <li>• Instruction Set Architecture</li> <li>• Assemblerprogrammierung</li> <li>• Grundzüge einer Hardware-Beschreibungssprache</li> <li>• RISC und CISC Architekturen</li> <li>• Steuerwerksentwurf und Mikroprogrammierung</li> <li>• Pipelining und Hazards</li> <li>• Speicherorganisation: Cachestrukturen und virtueller Speicher, Seitenverwaltung</li> <li>• Multi-Core Prozessoren</li> <li>• GPU-Architekturen und Programmierung</li> <li>• FPGA-Architekturen</li> </ul>		
14. Literatur:	Literatur, siehe Veranstaltungshinweise		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 365301 Vorlesung Rechnerorganisation 1</li> <li>• 365302 Praktische und theoretische Übungen Rechnerorganisation 1</li> </ul>		
16. Abschätzung Arbeitsaufwand:			



## Modul: 39040 Rechnernetze

2. Modulkürzel:	051200010	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel Frank Dürr		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 6. Semester → Wahlmodule --> Ergänzungsmodule		
11. Empfohlene Voraussetzungen:	- Programmierung und Software-Entwicklung - Datenstrukturen und Algorithmen - Grundkenntnisse in Java		
12. Lernziele:	- Versteht grundsätzliche Eigenschaften, Konzepte und Methoden von Rechnernetzen, insbesondere dem Internet. - Versteht Schichten und deren Zusammenwirken in einem Protokollstapel - Kann Rechnernetze aufbauen, verwalten und analysieren. - Kann Protokolle entwickeln und in Schichtenarchitektur einbetten. - Kann höhere Kommunikationsdienste zur Entwicklung von netzgestützten Systemen anwenden. - Kann sich mit Experten anderer Domänen über Methoden der Rechnernetze verständigen.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Einführung in die Rechnernetze, ISO Referenzmodell,</li> <li>• Bitübertragungsschicht: Übertragungsmedien, analoge und digitale Informationskodierung und -übertragung, Vermittlungsarten,</li> <li>• Sicherungsschicht: Betriebsarten, Fehlererkennung und -behandlung, Flusskontrolle,</li> <li>• Lokale Netze: CSMA/CD, Token Ring, Token Bus, FDDI, Kopplung,</li> <li>• Vermittlungsschicht: Verbindungsorientierter und verbindungsloser Dienst, Leitwegbestimmung, Überlastkontrolle,</li> <li>• Internetworking,</li> <li>• Internet-Protokoll,</li> <li>• Transportschicht: ausgewählte Realisierungsprobleme und Internet-Protokolle,</li> <li>• Echtzeitkommunikation: IntServ, DiffServ, Sicherheit: Verfahren, IPsec, SSL, TLS.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• A.S. Tanenbaum, Computer Networks, 4th Edition, 2003 - D.E. Comer, Computernetzwerke und Internets, 2000</li> <li>• D.E. Comer, Internetworking with TCP/IP Volume I: Principles, Protocols, and Architecture, 1995</li> <li>• J. F. Kurose, K. W. Ross, Computer Networks: a top-down approach featuring the Internet, 2001</li> </ul>		

- L.L. Peterson, B.S. Davie, Computer Networks: A Systems Approach, 1999
- 

15. Lehrveranstaltungen und -formen:
- 390401 VL Rechnernetze
  - 390402 ÜB Rechnernetze
- 

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name:
- 39041 Rechnernetze (PL), Schriftlich oder Mündlich, 90 Min., Gewichtung: 1
  - V Vorleistung (USL-V), Schriftlich oder Mündlich  
Prüfungsdauer: 90 min schriftlich oder 30 min mündlich
- 

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Verteilte Systeme

---

## Modul: 40090 Systemkonzepte und -programmierung

2. Modulkürzel:	051200005	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Kurt Rothermel		
9. Dozenten:	Kurt Rothermel Frank Dürr		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodule --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 3. Semester → Praktische Informatik --> Wahlpflichtmodule --> Kernmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlpflichtmodule allgemein --> Wahlpflichtmodule --> Kernmodule		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>• Modul 10280 Programmierung und Software-Entwicklung</li> <li>• Modul 12060 Datenstrukturen und Algorithmen</li> </ul>		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Verstehen grundlegender Architekturen und Organisationsformen von Software-Systemen</li> <li>• Verstehen systemnaher Konzepte und Mechanismen</li> <li>• Kann existierende Systemplattformen und Betriebssysteme hinsichtlich ihrer Eigenschaften analysieren und anwenden.</li> <li>• Kann systemnahe Software entwerfen und implementieren.</li> <li>• Kann nebenläufige Programme entwickeln</li> <li>• Kann mit Experten anderer Fachgebiete die Anwendung von Systemfunktionen abstimmen.</li> </ul>		
13. Inhalt:	Grundlegende Systemstrukturen - und organisationen <ul style="list-style-type: none"> <li>• Multitaskingsystem</li> <li>• Multiprozessorsystem</li> <li>• Verteiltes System Modellierung und Analyse nebenläufiger Programme</li> <li>• Abstraktionen: Atomare Befehle, Prozesse, nebenläufiges Programm</li> <li>• Korrektheit- und Leitungskriterien Betriebssystemkonzepte</li> <li>• Organisation von Betriebssystemen</li> <li>• Prozesse und Threads</li> <li>• Eingabe/Ausgabe</li> <li>• Scheduling Konzepte zur Synchronisation über gemeinsamen Speicher</li> <li>• Synchronisationsprobleme und -lösungen</li> <li>• Synchronisationswerkzeuge: Semaphore, Monitor Konzepte zur Kommunikation und Synchronisation mittels Nachrichtentransfer</li> <li>• Taxonomie: Kommunikation und Synchronisation</li> <li>• Nachrichten als Kommunikationskonzept</li> <li>• Höhere Kommunikationskonzepte Basisalgorithmen für Verteilte Systeme</li> <li>• Erkennung globaler Eigenschaften</li> </ul>		

- Schnappschussproblem
  - Konsistenter globaler Zustand
  - Verteilte Terminierung Praktische nebenläufige Programmierung in Java
  - Threads und Synchronisation
  - Socketschnittstelle
  - RMI Programmierung
- 

14. Literatur: Literatur, siehe Webseite zur Veranstaltung

---

15. Lehrveranstaltungen und -formen: 

- 400901 Vorlesung Systemkonzepte und -programmierung
- 400902 Übung Systemkonzepte und -programmierung

---

16. Abschätzung Arbeitsaufwand:

---

17. Prüfungsnummer/n und -name: 

- 40091 Systemkonzepte und -programmierung (PL), Schriftlich, 120 Min., Gewichtung: 1
- V Vorleistung (USL-V), Schriftlich oder Mündlich

[40091] Systemkonzepte und -programmierung (PL), schriftliche Prüfung, 120 Min., Gewicht: 1.0 [Prüfungsvorleistung] Vorleistung (USL-V), schriftlich, eventuell mündlich

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Verteilte Systeme

---

## Modul: 41590 Einführung in die Numerik und Stochastik für Softwaretechniker

2. Modulkürzel:	051240006	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:		Univ.-Prof. Dr. Dirk Pflüger	
9. Dozenten:		Thomas Ertl Daniel Weiskopf Miriam Mehl Stefan Zimmer Dirk Pflüger	
10. Zuordnung zum Curriculum in diesem Studiengang:		B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 3. Semester → Katalog SWT --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Zusatzmodule	
11. Empfohlene Voraussetzungen:		Mathematik für Informatiker und Softwaretechniker (Modulnummer 10190)	
12. Lernziele:		Beherrschung grundlegender Begriffe und Methoden der Numerik und Stochastik, Kenntnis der Anwendungsbereiche und Gültigkeitsgrenzen der erlernten Methoden, insbesondere Kenntnis der Auswirkungen von Näherungen, Beherrschung der Modellierung einfacher Probleme mit stochastischen Methoden.	
13. Inhalt:		Methoden der angewandten Mathematik, insbesondere der Numerik und Stochastik, sind für viele Bereiche der Informatik wie Simulation, Grafik oder Bildverarbeitung von zentraler Bedeutung. In Ergänzung der Mathematik-Grundausbildung vermittelt diese Vorlesung folgende Grundkenntnisse: <ul style="list-style-type: none"> <li>• numerische Algorithmik</li> <li>• Gleitpunktzahlen und Gleitpunktarithmetik</li> <li>• Interpolation und Approximation</li> <li>• Integration</li> <li>• lineare Gleichungssysteme</li> <li>• Iterative Lösung linearer und nichtlinearer Gleichungen</li> <li>• gewöhnliche Differentialgleichungen</li> <li>• Stochastik</li> <li>• Zufall und Unsicherheit</li> <li>• diskrete und kontinuierliche Wahrscheinlichkeitsräume</li> <li>• Asymptotik</li> </ul>	
14. Literatur:		<ul style="list-style-type: none"> <li>• Dahmen, Reusken: Numerik für Ingenieure</li> <li>• Schwarz, Köckler: Numerische Mathematik</li> <li>• Huckle, Schneider: Numerik für Informatiker</li> <li>• Henze: Stochastik für Einsteiger</li> <li>• Schickinger, Steger: Diskrete Strukturen, Band 2</li> </ul>	
15. Lehrveranstaltungen und -formen:		<ul style="list-style-type: none"> <li>• 415901 Vorlesung Einführung in die Numerik und Stochastik für Softwaretechniker</li> <li>• 415902 Übung Einführung in die Numerik und Stochastik für Softwaretechniker</li> </ul>	
16. Abschätzung Arbeitsaufwand:		Präsenzstunden: 42 h	

Eigenstudiumstunden: 138 h

Gesamtstunden: 180 h

---

17. Prüfungsnummer/n und -name: 41591 Einführung in die Numerik und Stochastik für Softwaretechniker (PL), Schriftlich, 90 Min., Gewichtung: 1  
[41591] Einführung in die Numerik und Stochastik für Softwaretechniker (PL), schriftlich, eventuell mündlich, 90 Min., Gewicht: 1.0

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Simulation Software Engineering

---

## Modul: 42410 Grundlagen des Wissenschaftlichen Rechnens

2. Modulkürzel:	051240020	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Dirk Pflüger		
9. Dozenten:	Miriam Mehl Stefan Zimmer Dirk Pflüger		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlpflichtmodule allgemein --&gt; Wahlpflichtmodule --&gt; Kernmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISG --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, 5. Semester → Katalog ISW --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2017, 5. Semester → Wahlmodule --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	<ul style="list-style-type: none"> <li>- Modul 10190 Mathematik für Informatiker und Softwaretechniker und</li> <li>- Modul 10240 Numerische und Stochastische Grundlagen der Informatik bzw.</li> <li>- Modul 41590 Einführung in die Numerik und Stochastik für Softwaretechniker</li> </ul>		
12. Lernziele:	<p>Kenntnis grundlegender Konzepte, Algorithmen und Methoden des Wissenschaftlichen Rechnens. Fähigkeit, mit den erlernten Kenntnissen selbständig Methoden zu entwickeln, zu analysieren und umzusetzen, mit denen anwendungsorientierte Probleme effizient und genau gelöst werden können.</p>		
13. Inhalt:	<ul style="list-style-type: none"> <li>- Überblick über die Simulationspipeline und die Wechselwirkungen zwischen den einzelnen Schritten</li> <li>- Skalenabhängige Modellierung</li> <li>- Diskretisierung (Gitter, Finite Elemente, Zeitschrittverfahren)</li> <li>- Algorithmen (Gittergenerierung, Adaptivität, Lineare Löser, Linked-Cell, Fast Multipole)</li> <li>- Parallelisierung (Gitterpartitionierung, Lastbalancierung)</li> <li>- Kurzer Überblick über die Visualisierung</li> </ul>		
14. Literatur:	<p>Martin Hanke-Bourgeois: Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens, Vieweg+Teubner Verlag 2009.</p>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 424102 Übung Grundlagen des Wissenschaftlichen Rechnens</li> <li>• 424101 Vorlesung Grundlagen des Wissenschaftlichen Rechnens</li> </ul>		
16. Abschätzung Arbeitsaufwand:	<p>Präsenzstunden: 42 h Eigenstudiumstunden: 138 h Gesamtstunden: 180 h</p>		
17. Prüfungsnummer/n und -name:	<p>42411 Grundlagen des Wissenschaftlichen Rechnens (PL), Schriftlich oder Mündlich, 90 Min., Gewichtung: 1 [42411] Grundlagen des Wissenschaftlichen Rechnens (PL), schriftlich oder mündlich, 90 Min., Gewicht: 1.0</p>		

18. Grundlage für ... :

---

19. Medienform:

---

20. Angeboten von: Simulation Software Engineering

---

## Modul: 71740 System and Web Security

2. Modulkürzel:	052900002	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Sommersemester
4. SWS:	4	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Dr. rer. nat. Ralf Küsters		
9. Dozenten:	Ralf Küsters		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, → Wahlmodule aus Master SWT --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2012, → Vorgezogene Master-Module B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodul aus Master --> Ergänzungsmodule B.Sc. Softwaretechnik, PO 123-2017, → Vorgezogene Master-Module		
11. Empfohlene Voraussetzungen:	Solide Kenntnisse in mindestens einer Programmiersprache.		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Students are sensitized for common security vulnerabilities and attack vectors in computer systems and the web,</li> <li>• Students are familiar with concrete attacks on computer systems and the web, and understand the underlying principles,</li> <li>• Students are familiar with common defense mechanisms.</li> </ul>		
13. Inhalt:	<p>IT-systems are constantly under attack, by various kinds of attackers with diverse interests: criminal organizations with monetary interests, intelligence agencies, industrial espionage by states and companies.</p> <p>The course covers the most common attack vectors on computer systems, including mobile devices, and the web, including, for example, stack and heap overflows, format string vulnerabilities, integer overflows, return-oriented-programming, Cross-Site-Scripting (CSS/XSS), SQL Injections, and Cross-Site-Request-Forgery (XSRF), etc.</p> <p>The course also discusses common defense mechanisms, including, for example, access control mechanisms, address space layout randomization (ASLR), static code analysis, security monitoring, input/output sanitization, prepared statements, etc.</p>		
14. Literatur:	Will be announced in class		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 717401 Vorlesung System and Web Security</li> <li>• 717402 Übung System and Web Security</li> </ul>		
16. Abschätzung Arbeitsaufwand:	Vorlesung und Übung System- und Websicherheit		
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 71741 System and Web Security (PL), Schriftlich, Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Unbenotete Studienleistung als Vorleistung (USL-V); ausreichende Punktzahl in den Übungen</li> </ul> Prüfungsleistung (PL): Klausur (90 Minuten) zur Vorlesung und Übung System- und Websicherheit		
18. Grundlage für ... :			
19. Medienform:	Projektor, Tafel		
20. Angeboten von:	Informationssicherheit		

## Modul: 78900 Introduction to Modern Cryptography

2. Modulkürzel:	052900003	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	6 LP	6. Turnus:	Wintersemester
4. SWS:	4	7. Sprache:	Englisch
8. Modulverantwortlicher:	Univ.-Prof. Dr. rer. nat. Ralf Küsters		
9. Dozenten:	Ralf Küsters		
10. Zuordnung zum Curriculum in diesem Studiengang:	<p>B.Sc. Softwaretechnik, PO 123-2017, → Vorgezogene Master-Module</p> <p>B.Sc. Softwaretechnik, PO 123-2017, → Wahlmodul aus Master --&gt; Ergänzungsmodule</p> <p>B.Sc. Softwaretechnik, PO 123-2012, → Vorgezogene Master-Module</p> <p>B.Sc. Softwaretechnik, PO 123-2012, → Wahlmodule aus Master SWT --&gt; Ergänzungsmodule</p>		
11. Empfohlene Voraussetzungen:	<p>Die Veranstaltung verlangt solide Kenntnisse in den Grundlagen der Mathematik wie sie in den ersten drei oder vier Semestern eines Bachelorstudiengangs in Informatik/Mathematik vermittelt werden.</p> <p>Kenntnisse der Inhalte der Vorlesung <i>Grundlagen der Informationssicherheit</i> sind nützlich, aber keine zwingende Voraussetzung.</p>		
12. Lernziele:	<p>Students will acquire an in-depth understanding of cryptography. They will be able to judge and assess the security of cryptographic constructions used in practice (encryption schemes, digital signatures, messages authentication codes, etc.) and will be able to read scientific papers on cryptography.</p>		
13. Inhalt:	<p>p { margin-bottom: 0.1in; direction: ltr; color: rgb(0, 0, 10); line-height: 120%; text-align: left; }p.western { font-family: "Calibri", serif; font-size: 11pt; }p.cjk { font-family: "Times New Roman"; font-size: 11pt; }p.cjl { font-family: "Times New Roman"; font-size: 11pt; }a:link { color: rgb(0, 0, 255); }a.cjl:link { font-family: "Times New Roman"; }</p> <p>Cryptography is everywhere! We heavily rely on cryptography in our everyday life when we do, for example, online shopping and online banking, pay with credit or debit card, open doors with electronic keys, or when we use social networks, instant messengers, online games, WiFi, mobile networks, or electronic currencies. Here, cryptography is essential in order to guarantee various central security properties such as secrecy and integrity of messages as well as authenticity of the communication partners. This course provides an introduction to modern cryptography. In the traditional approach to cryptography, cryptographers proposed, for example, encryption algorithms, and then others, cryptanalysts, tried to break them. In modern cryptography, cryptographers try to prove that their cryptographic constructions are secure under certain assumptions, even when attacked by powerful adversaries. Hence, cryptography turned from pure art to science.</p> <p>The course covers several fundamental cryptographic primitives, including (symmetric and asymmetric) encryption, hash functions, digital signatures, and message authentication codes. These primitives are important building blocks for other cryptographic</p>		

constructions and for cryptographic protocols (TLS, SSH, WPA2, etc.), used by billions of people every day. The course presents common cryptographic constructions as used in practice, such as AES with various encryption modes (e.g., CBC, CTR), RSA, ElGamal, HMAC, PKCS#1, DSA. It also discusses public-key infrastructures and cryptographic protocols.

In the spirit of modern cryptography, we ask the following questions: What does it mean for an encryption algorithm, digital signature, etc. to be secure? Under which assumptions can we prove security? For several cryptographic constructions used in practice, including those mentioned above, we prove security or present attacks. This provides a deep understanding of the security/insecurity of the cryptography that surrounds us.

---

14. Literatur:	<ul style="list-style-type: none"> <li>• Ralf Küsters and Thomas Wilke. Moderne Kryptographie - Eine Einführung. Vieweg + Teubner, 2011.</li> <li>• Jonathan Katz and Yehuda Lindell. Introduction to Modern Cryptography - Second Edition. CRC Press 2015.</li> </ul>
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 789001 Vorlesung und Übung zu Introduction to Modern Cryptography</li> </ul>
16. Abschätzung Arbeitsaufwand:	Vorlesung und Übung zu Introduction to Modern Cryptography
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• V Vorleistung (USL-V),</li> <li>• 78901 Introduction to Modern Cryptography (PL), Schriftlich oder Mündlich, Gewichtung: 1</li> </ul> <p>Unbenotete Studienleistung als Vorleistung (USL-V); ausreichende Punktzahl in den Übungen                  Prüfungsleistung (PL): Klausur (90 Minuten) oder mündliche Prüfung (30 Minuten) zur Vorlesung und Übung Introduction to Modern Cryptography</p>
18. Grundlage für ... :	
19. Medienform:	Projector, blackboard
20. Angeboten von:	Informationssicherheit

---

## 400 Schlüsselqualifikationen fachaffin

---

Zugeordnete Module: 16610 Studienprojekt-Pr

---

## Modul: 16610 Studienprojekt-Pr

2. Modulkürzel:	051520191	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	15 LP	6. Turnus:	Wintersemester/ Sommersemester
4. SWS:	10	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Wagner		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2012, 6. Semester → Schlüsselqualifikationen fachaffin		
11. Empfohlene Voraussetzungen:	Das Modul Programmentwicklung / Programmentwicklung I + II muss vor Beginn des Studienprojekts absolviert sein.		
12. Lernziele:	Im Studienprojekt-Pr werden die Prinzipien der Kooperation in einem größeren, für die Praxis typischen Projekt angewendet und eingeübt. Dazu gehören die Kontakte zum Kunden (Anforderungsanalyse), die Projektplanung, die Kostenschätzung, die Qualitätssicherung und die Präsentation der Resultate in schriftlicher und mündlicher Form, auch die Techniken zur Konfliktlösung und zum Risiko-Management. Natürlich kommt auch das fachliche Wissen zur Realisierung eines Softwaresystems zum Zuge. Die Teilnehmer sind nach dem Projekt in der Lage, ein größeres Softwareprojekt zu organisieren und vollständig durchzuführen.		
13. Inhalt:	Die Teilnehmer entwickeln ein Softwaresystem nach Vorgaben des Kunden von der Angebotserstellung bis zur Übergabe. Störungen und Änderungen der Aufgabe im Projektverlauf sind normale Bestandteile des Projekts. Typisch beginnt das Studienprojekt mit der Erhebung der Anforderungen und der Anfertigung eines Angebots, darauf folgt die Entwicklung nach einem zu Beginn gewählten Prozessmodell. Das Projekt wird mit der Übergabe der Software in einer Präsentation abgeschlossen. Die Teilnehmer fertigen einen Bericht an, der die individuellen Leistungen erkennen lässt		
14. Literatur:			
15. Lehrveranstaltungen und -formen:	• 166101 Praktikum Studienprojekt-Pr		
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	<ul style="list-style-type: none"> <li>• 16611 Studienprojekt-Pr (LBP), Sonstige, Gewichtung: 1</li> <li>• V Vorleistung (USL-V), Sonstige</li> <li>• V Vorleistung (USL-V), Sonstige</li> </ul> <p>Ausreichende Leistungen im Studienprojekt-Pr werden in jedem der beiden Semester durch einen unbenoteten Schein bestätigt. Die Note im Studienprojekt-Pr wird auf der Grundlage der im Projekt gezeigten Leistungen und des Projektberichts, der die individuellen Beiträge der Teilnehmer angeben muss, bestimmt. Sie geht in die Gesamtnote des Studienprojekts mit dem Gewicht 5 ein, der andere Beitrag kommt von Studienprojekt-Th mit dem Gewicht 3.</p>		
18. Grundlage für ... :			

19. Medienform:

20. Angeboten von:

Formale Methoden der Informatik

---

## Modul: 81110 Bachelorarbeit Softwaretechnik

2. Modulkürzel:	050410106	5. Moduldauer:	Einsemestrig
3. Leistungspunkte:	12 LP	6. Turnus:	Wintersemester/ Sommersemester
4. SWS:	8	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Univ.-Prof. Dr. Stefan Funke		
9. Dozenten:	Dozenten der Informatik		
10. Zuordnung zum Curriculum in diesem Studiengang:	B.Sc. Softwaretechnik, PO 123-2017, B.Sc. Softwaretechnik, PO 123-2012,		
11. Empfohlene Voraussetzungen:	sämtliche Basis- und Kernmodule sowie mindestens 120 LP im Bachelorstudiengang Softwaretechnik		
12. Lernziele:	Die Bachelorarbeit soll zeigen, dass innerhalb einer vorgegebenen Frist eine Aufgabenstellung aus dem Bereich Softwaretechnik selbständig nach wissenschaftlichen Methoden bearbeitet und die Ergebnisse sachgerecht dargestellt werden können.		
13. Inhalt:	wird von Betreuer / Prüfer festgelegt		
14. Literatur:	wird vom Betreuer / Prüfer bekanntgegeben		
15. Lehrveranstaltungen und -formen:			
16. Abschätzung Arbeitsaufwand:			
17. Prüfungsnummer/n und -name:	Schriftliche Ausarbeitung zum vergebenen Thema sowie ein Vortrag über den Inhalt der Bachelorarbeit		
18. Grundlage für ... :			
19. Medienform:			
20. Angeboten von:	Software Engineering		