

**Prüfungsordnung  
für den Bachelorstudiengang  
Informatik  
an der Fachhochschule Bielefeld  
(University of Applied Sciences)  
vom 22.12.2010  
in der Fassung der Änderung vom 17.05.2011, 25.07.2012, 17.06.2013 und  
vom 19.05.2014**

Aufgrund des § 2 Abs. 4 und des § 64 Abs. 1 des Gesetzes über die Hochschulen des Landes Nordrhein-Westfalen (Hochschulgesetz - HG) vom 31. Oktober 2006 (GV. NRW. S. 474), zuletzt geändert durch Artikel 1 des Gesetzes vom 3. Dezember 2013 (GV. NRW. S. 723), hat der Fachbereich Technik der Fachhochschule Bielefeld die folgende Ordnung erlassen:

**Inhaltsübersicht**

**I. Allgemeines**

- § 1 Geltungsbereich der Prüfungsordnung
- § 2 Ziel des Studiums, Zweck der Prüfung, Akademischer Grad
- § 3 Allgemeine Zulassungsvoraussetzungen
- § 4 Anrechnung von Studienzeiten, Studien- und Prüfungsleistungen
- § 5 Regelstudienzeit, Semesterstruktur, Studienumfang
- § 6 Arten des Lehrangebots

**II. Studienbegleitende Prüfungen und Prüfungsabläufe**

- § 7 Umfang und Gliederung der Prüfungen
- § 8 Organisation der Prüfungen, Prüfungsorgane
- § 9 Prüfende und Beisitzende
- § 10 Ziel, Umfang und Form der Modulprüfungen
- § 11 Klausurarbeiten und sonstige schriftliche Arbeiten
- § 12 Mündliche Prüfungen
- § 13 Hausarbeiten
- § 14 Projektarbeiten
- § 15 Performanzprüfungen
- § 16 Abzuleistende Modulprüfungen, Credits
- § 17 Fortschrittsregelung
- § 18 Zulassung zu Klausuren und mündlichen Prüfungen
- § 19 Zulassung zu sonstigen Prüfungsarten
- § 20 Durchführung von Modulprüfungen
- § 21 Bewertung von Prüfungsleistungen
- § 22 Wiederholung von Prüfungsleistungen
- § 23 Versäumnis, Rücktritt, Täuschung, Ordnungsverstoß

**III. Praxisphase / Auslandssemester**

- § 24 Praxisphase
- § 25 Eignung der Praxisstelle
- § 26 Betreuung der Studierenden in der Praxisphase
- § 27 Fakultatives Auslandssemester

#### **IV. Bachelorarbeit**

- § 28 Bachelorarbeit
- § 29 Zulassung zur Bachelorarbeit
- § 30 Ausgabe und Bearbeitung der Bachelorarbeit
- § 31 Abgabe und Bewertung der Bachelorarbeit
- § 32 Kolloquium

#### **V. Zusatzmodule, Bachelorprüfung**

- § 33 Zusatzmodule
- § 34 Bachelorprüfung
- § 35 Ergebnis der Bachelorprüfung
- § 36 Zeugnis, Gesamtnote, Bachelorurkunde, Diploma Supplement

#### **VI. Schlussbestimmungen**

- § 37 Einsicht in die Prüfungsakten
- § 38 Ungültigkeit von Prüfungen
- § 39 In-Kraft-Treten, Veröffentlichung

- Anlage 1 Studienplan
- Anlage 2 Modulbeschreibungen (Modulhandbuch)

## I. Allgemeines

### § 1

#### **Geltungsbereich der Prüfungsordnung**

Diese Prüfungsordnung gilt für den Abschluss des Studiums in dem Bachelorstudiengang Informatik an der Fachhochschule Bielefeld. Sie regelt die Prüfungen, den Inhalt und den Aufbau des Studiums unter Berücksichtigung der fachlichen und hochschuldidaktischen Entwicklungen und enthält die inhaltliche Beschreibung der Prüfungsgebiete in diesem Studiengang.

### § 2

#### **Ziel des Studiums, Zweck der Prüfung, Akademischer Grad**

- (1) Die Bachelorprüfung bildet den ersten berufsqualifizierenden Abschluss eines Hochschulstudiums und dient des Weiteren der Qualifizierung für ein Masterstudium an einer Fachhochschule oder an einer Universität.
- (2) Das Bachelorstudium gewährleistet auf der Grundlage wissenschaftlicher Erkenntnisse und Methoden und unter Beachtung der allgemeinen gesetzlichen Studienziele (§ 58 HG) eine deutliche Berufsqualifizierung. Der Studiengang vermittelt daher den Absolventinnen und Absolventen Qualifikationsbündel bzw. -attribute, die ihnen die Aufnahme einer qualifikations-adäquaten beruflichen Tätigkeit nach dem Studium ermöglichen.
- (3) Ziel des Studiums ist eine praxisnahe berufsqualifizierende Ausbildung zu allgemeinen Techniken der Informatik und deren Anwendungsgebieten und eine Qualifizierung, die den Zugang zu Masterstudiengängen im Bereich Informatik ermöglicht.
- (4) Im Rahmen des Pflicht- oder Wahlpflichtbereiches sind unter Beachtung der Maßgaben des Absatzes 2 folgende überfachliche Qualifikationen zu gewährleisten:
  1. Fähigkeit zu wissenschaftlichem Arbeiten einschließlich der dazu erforderlichen Informations- und Medienkompetenz;
  2. Verständnis für betriebswirtschaftliche Zusammenhänge;
  3. fremdsprachliche Kompetenz;
  4. Fähigkeit, Ideen, Konzepte, Projekte oder Produkte in mündlicher, schriftlicher und digitaler Form zu präsentieren;
  5. Fähigkeit zur Teamarbeit, zur Moderation und zur Leitung von Arbeitsgruppen;
  6. Fähigkeit, auf dem Hintergrund wissenschaftlicher Erkenntnisse und Methoden konkrete Fragestellungen des Berufsfeldes in einem vorgegebenen Zeitrahmen zu bearbeiten.
- (6) Aufgrund der bestandenen Bachelorprüfung wird der akademische Grad „Bachelor of Science“ (B.Sc.) verliehen.

### § 3

#### **Allgemeine Zulassungsvoraussetzungen**

- (1) Die Qualifikation für das Studium wird durch ein Zeugnis der Fachhochschulreife, der allgemeinen Hochschulreife oder durch eine als gleichwertig anerkannte Vorbildung nachgewiesen. Das Nähere ergibt sich aus der Verordnung über die Gleichwertigkeit von Vorbildungsnachweisen mit dem Zeugnis der Fachhochschulreife (Qualifikationsverordnung Fachhochschule - QVO-FH vom 20.06.02; GV. NRW. S. 312) in der jeweils geltenden Fassung.
- (2) Studienbewerberinnen und -bewerber ohne den Nachweis der Qualifikation durch ein Zeugnis der Hochschulreife (allgemeine Hochschulreife oder Fachhochschulreife) können gemäß Zugangsprüfungsordnung der Fachhochschule Bielefeld zugelassen werden.
- (3) Trotz Erfüllung der allgemeinen Zulassungsvoraussetzungen kann die Einschreibung bzw. der Studiengangwechsel versagt werden, wenn die Studienbewerberin oder der Studienbewerber an einer Hochschule im Geltungsbereich des Grundgesetzes eine nach der Prüfungsordnung erforderliche Prüfung in einem verwandten oder vergleichbaren Studiengang endgültig nicht bestanden hat.
- (4) Studienbewerberinnen und -bewerber, die für ein erfolgreiches Studium erforderliche Kenntnisse und Fähigkeiten auf andere Weise als durch ein Studium erworben haben, sind nach dem Ergebnis einer Einstufungsprüfung berechtigt, das Studium in einem dem Ergebnis entsprechenden Abschnitt des Studiengangs aufzunehmen, soweit nicht Regelungen über die Vergabe von Studienplätzen entgegenstehen. Die Regelungen des Zulassungsrechts bleiben unberührt.
- (5) Nach dem Ergebnis der Einstufungsprüfung können die Teilnahme an Pflicht- und Wahlpflichtveranstaltungen und die entsprechenden Modulprüfungen ganz oder teilweise erlassen werden. Über die Entscheidung wird eine Bescheinigung ausgestellt.

- (6) Das Nähere über Art, Form und Umfang der Einstufungsprüfung regelt die Einstufungsprüfungsordnung für die Studiengänge der Fachhochschule Bielefeld in der jeweils geltenden Fassung

#### **§ 4**

##### **Anrechnung von Studienzeiten, Studien- und Prüfungsleistungen**

- (1) Studienzeiten, Studien- und Prüfungsleistungen im gleichen Studiengang an anderen Hochschulen im Geltungsbereich des Grundgesetzes werden von Amts wegen angerechnet. Studien- und Prüfungsleistungen in anderen Studiengängen werden anerkannt, soweit die Gleichwertigkeit festgestellt ist. Gleichwertigkeit ist festzustellen, wenn
  - entweder Studienzeiten, Studien- und Prüfungsleistungen in Inhalt, Umfang und in den Anforderungen denjenigen des Bachelorstudiengangs an der Fachhochschule Bielefeld im Wesentlichen entsprechen (§ 63 Abs. 2 Satz 2 HG NRW)
  - oder keine wesentlichen Unterschiede zwischen den zu vergleichenden Zeiten (Art. V Ziff. 1 Lissabon-Konvention) bzw. Leistungen (Art. VI Ziff. 1 Lissabon-Konvention) bestehen. Dabei ist kein schematischer Vergleich, sondern eine Gesamtbetrachtung und Gesamtbewertung vorzunehmen.
- (2) Gleichwertige Studien- und Prüfungsleistungen an Hochschulen außerhalb des Geltungsbereichs des Grundgesetzes werden auf Antrag angerechnet. Für die Gleichwertigkeit sind die von der Kultusministerkonferenz und Hochschulrektorenkonferenz gebilligten Äquivalenzvereinbarungen maßgebend. Soweit Äquivalenzvereinbarungen nicht vorliegen, entscheidet der Prüfungsausschuss über die Anrechnung. Bei Zweifeln in Fragen der Gleichwertigkeit werden die Prüfenden des Fachbereichs oder die Zentralstelle für ausländisches Bildungswesen beteiligt.
- (3) Sonstige Kenntnisse und Qualifikationen werden auf Antrag auf der Grundlage vorgelegter Unterlagen angerechnet, sofern sie nicht bereits Voraussetzung für die Zulassung waren.
- (4) Fehlversuche in verwandten oder vergleichbaren Prüfungsleistungen sind anzurechnen. Alle Pflichtmodule sind in diesem Studiengang zu erbringen und können nicht als Fremdleistung in einem anderen Studiengang belegt und angerechnet werden.
- (5) Über die Anrechnung nach den Absätzen 1 bis 4 entscheidet der Prüfungsausschuss nach den Richtlinien des ECTS, im Zweifelsfall nach Anhörung von den für die Fächer zuständigen Prüfenden.

#### **§ 5**

##### **Regelstudienzeit, Semesterstruktur, Studienumfang**

- (1) Die Regelstudienzeit beträgt sieben Semester. Der Studiengang ist modular aufgebaut. Jedes Modul schließt mit einer Prüfung ab. Der für ein Modul aufzuwendende Arbeitsaufwand wird durch Leistungspunkte (Credit Points) beschrieben. Credits umfassen sowohl den unmittelbaren Lehrbetrieb als auch Zeiten für die Vor- und Nachbereitung der Module, den Prüfungsaufwand und die Prüfungsvorbereitungen. Nach bestandener Prüfung werden die entsprechenden Leistungspunkte gutgeschrieben und getrennt von den erzielten Prüfungsnoten ausgewiesen. Entsprechend dem European Credit Transfer System (ECTS – Europäisches System zur Anrechnung von Studienleistungen) werden pro Semester 30 Credits vergeben und den Modulen zugeordnet. Die spezifischen Prüfungsanforderungen, die Pflichtmodule und die Wahlpflichtmodule sowie auch die entsprechenden Credits sind in den Anlagen 1 und 2 verbindlich geregelt.
- (2) Der Studienplan (Anlage 1) legt den Arbeitsaufwand und den Zeitumfang der einzelnen Module in Credits und Semesterwochenstunden sowie deren empfohlene Zeitlage im Studienverlauf fest. Er ist nach Studiensemestern gegliedert. Die Lehrveranstaltungen werden im Jahresrhythmus angeboten.
- (3) Der Leistungsumfang beträgt in dem siebensemestrigen Studiengang 210 Credits.
- (4) Abweichend von Absatz 1 können einzelne Module gem. Modulhandbuch (Anlage 2) nur mit den Prädikaten „bestanden“ oder „nicht bestanden“ bewertet werden. Die Bewertung dieser Module geht nicht in die Ermittlung der Gesamtnote des Bachelorabschlusses ein.
- (5) Um den Studierenden den Zugang zum Lehrangebot zu erleichtern, werden zu Beginn des ersten Semesters Einführungsveranstaltungen angeboten.

#### **§ 6**

##### **Arten des Lehrangebots**

- (1) Das notwendige Lehrangebot enthält Pflicht- und Wahlpflichtmodule (siehe Anlage 1).
- (2) Wahlpflichtmodule sind Module aus Vertiefungsbereichen, die gewählt und mit einer Modulprüfung abgeschlossen werden müssen.

- (3) Zusatzmodule sind freiwillig erbrachte Leistungen, für deren Anerkennung sich die Studierenden einer Prüfung (§ 33 PO) unterziehen müssen.
- (4) Formen der Lehrveranstaltungen sind:
- **Vorlesung (V):** Zusammenhängende Darstellung eines Lehrstoffes, Vermittlung von Fakten und Methoden.
  - **Übung (Ü):** Systematisches Durcharbeiten von Lehrstoffen und Zusammenhängen, Anwendung an praktischen Beispielen auf Fälle aus der Praxis. Die Lehrenden leiten die Veranstaltungen, geben eine Einführung, stellen Aufgaben, geben Lösungshilfen. Die Studierenden bereiten die Veranstaltung vor, arbeiten einzeln oder in Gruppen, lösen Aufgaben teilweise selbständig, aber in enger Rückkopplung mit den Lehrenden.
  - **Praktikum, Labor (P):** Erwerben und Vertiefen von Kenntnissen durch Bearbeitung und Programmierung praktischer und experimenteller Aufgaben im Rechnerlabor. Die Lehrenden leiten die Studierenden an und überwachen die Veranstaltung. Die Studierenden führen praktische Arbeiten und Versuche durch und erhalten darüber ein Testat.
  - **Seminaristischer Unterricht:** Didaktischer Mix mit abwechselnden Phasen der Gruppenarbeit, Stoffvermittlung als Referat oder Lehrgespräch durch den Dozenten und Kurzreferate der Studierenden.

## II. Studienbegleitende Prüfungen und Prüfungsabläufe

### § 7

#### Umfang und Gliederung der Prüfungen

- (1) Die studienbegleitenden Modulprüfungen sollen zu dem Zeitpunkt stattfinden, an dem das jeweilige Modul im Studium abgeschlossen wird. Hinsichtlich der Leistungen und der zeitlichen Bestimmungen im Zusammenhang mit der Bachelorarbeit gelten die Regelungen gemäß §§ 28-31.
- (2) Das Studium sowie das Prüfungsverfahren sind so zu gestalten, dass das Studium einschließlich der Bachelorarbeit mit Ablauf des siebten Semesters abgeschlossen sein kann.
- (3) Die Prüfungsverfahren müssen die Inanspruchnahme von Schutzbestimmungen entsprechend den §§ 3, 4, 6 und 8 des Mutterschutzgesetzes sowie entsprechend den Fristen des Bundeserziehungsgeldgesetzes über die Elternzeit und die Ausfallzeiten durch die Pflege von Personen nach § 48 Abs. 5 Satz 2 Nr. 5 HG berücksichtigen (§ 64 Abs. 2 Nr. 5 HG).

### § 8

#### Organisation der Prüfungen, Prüfungsorgane

- (1) Für die Prüfungsorganisation ist die Dekanin oder der Dekan gemäß § 27 Abs. 1 Satz 2 HG verantwortlich. Diese Aufgaben können durch einen Prüfungsausschuss wahrgenommen werden.
- (2) Die Dekanin oder der Dekan oder der Prüfungsausschuss fungieren entsprechend ihrer Bestimmung in der Prüfungsordnung als Behörde im Sinne des Verwaltungsverfahrensgesetzes NRW und der Verwaltungsgerichtsordnung.
- (3) Wenn ein Prüfungsausschuss als Prüfungsbehörde eingerichtet wird, sollen in der Regel diesem Gremium nicht mehr als sieben Mitglieder angehören. In diesem Fall entspricht folgende Zusammensetzung den Maßgaben des HG:
  1. vier Mitgliedern der Professorenschaft, darunter einem vorsitzenden Mitglied und einem stellvertretend vorsitzenden Mitglied,
  2. einem Mitglied der Mitarbeiterschaft in Lehre und Forschung mit Hochschulabschluss,
  3. zwei Studierenden.
- (4) Die Mitglieder des Prüfungsausschusses werden vom zuständigen Fachbereichsrat gewählt. Entsprechend wird durch die Wahl bestimmt, wer die Mitglieder, mit Ausnahme des vorsitzenden Mitglieds, und des stellvertretend vorsitzenden Mitglieds im Verhinderungsfall vertreten soll. Die Amtszeit der Mitglieder beträgt vier Jahre, die der studentischen Mitglieder ein Jahr. Die Wiederwahl eines Mitglieds ist möglich. Scheidet ein Mitglied vorzeitig aus, wird ein Nachfolger für die restliche Amtszeit gewählt.
- (5) Der Prüfungsausschuss achtet auf die Einhaltung der Prüfungsordnung. Er entscheidet insbesondere über Widersprüche gegen in Prüfungsverfahren getroffene Entscheidungen. Darüber hinaus hat der Prüfungsausschuss dem Fachbereichsrat über die Entwicklung der Prüfungen und Studienzeiten jährlich zu berichten. Er gibt Anregungen zur Reform der Prüfungsordnung und der Studienpläne. Der Prüfungsausschuss kann die Erledigung seiner Aufgaben für alle Regelfälle auf das vor-

sitzende Mitglied, bzw. das stellvertretend vorsitzende Mitglied des Prüfungsausschusses übertragen; dies gilt nicht für die Entscheidung über Widersprüche.

- (6) Der Prüfungsausschuss ist beschlussfähig, wenn das vorsitzende Mitglied (oder Stellvertretung), ein weiteres Mitglied der Professorenschaft und ein weiteres stimmberechtigtes Mitglied anwesend sind. Er beschließt mit einfacher Stimmenmehrheit. Bei Stimmgleichheit entscheidet die Stimme des vorsitzenden Mitglieds. Die studentischen Mitglieder wirken bei pädagogisch-wissenschaftlichen Entscheidungen, insbesondere bei der Anrechnung oder sonstigen Beurteilung von Studien- und Prüfungsleistungen und der Bestellung von Prüfenden und Beisitzenden, nicht mit. An der Beratung und Beschlussfassung über Angelegenheiten, welche die Festlegung von Prüfungsaufgaben oder die ihre eigene Prüfung betreffen, nehmen die studentischen Mitglieder des Prüfungsausschusses nicht teil.
- (8) Die Mitglieder des Prüfungsausschusses, mit Ausnahme der studentischen Mitglieder, die sich im gleichen Prüfungszeitraum der gleichen Prüfung unterziehen, haben das Recht, der Abnahme der Prüfungen beizuwohnen. Dieses Recht erstreckt sich nicht auf die Bekanntgabe der Note.
- (9) Die Mitglieder des Prüfungsausschusses (einschl. der Stellvertretung), die Prüfenden und die Beisitzenden unterliegen der Amtsverschwiegenheit. Sofern sie nicht im öffentlichen Dienst stehen, sind sie durch den Prüfungsausschuss zur Verschwiegenheit zu verpflichten.
- (10) Belastende Entscheidungen des Prüfungsausschusses sind dem betroffenen Studierenden unverzüglich schriftlich mitzuteilen. Der Bescheid ist mit einer Rechtsbehelfsbelehrung zu versehen.

## § 9

### Prüfende und Beisitzende

- (1) Der Prüfungsausschuss bestellt die Prüfenden und Beisitzenden. Zur/zum Prüfenden darf nur bestellt werden, wer mindestens die Bachelorprüfung an einer Hochschule oder eine vergleichbare Prüfung abgelegt hat oder eine vergleichbare Qualifikation erworben hat und, sofern nicht zwingende Gründe eine Abweichung erfordern, in dem Studienabschnitt, auf den sich die Prüfung bezieht, eine einschlägige selbständige Lehrtätigkeit ausgeübt hat. Sind mehrere Prüfer zu bestellen, so soll mindestens eine prüfende Person in dem betreffenden Prüfungsfach gelehrt haben. Zu Beisitzenden dürfen nur Personen bestellt werden, die mindestens die Bachelorprüfung an einer Hochschule oder eine vergleichbare Prüfung abgelegt oder eine vergleichbare Qualifikation erworben haben (sachkundige Beisitzende). Die Prüfenden sind in ihrer Prüfungstätigkeit unabhängig. Die Prüfenden und die Beisitzenden unterliegen der Amtsverschwiegenheit. Der Prüfungsausschuss achtet darauf, dass die Prüfungsverpflichtung möglichst gleichmäßig auf die Prüfenden verteilt wird.
- (2) Das vorsitzende Mitglied des Prüfungsausschusses sorgt dafür, dass der/dem Studierenden die Namen der Prüfenden sowie die Prüftermine rechtzeitig (mind. zwei Wochen vor der Prüfung) bekannt gegeben werden. Die Bekanntmachung durch Aushang ist ausreichend.

## § 10

### Ziel, Umfang und Form der Modulprüfungen

- (1) Eine Modulprüfung ist eine studienbegleitende Prüfungsleistung. In den Modulprüfungen soll festgestellt werden, ob die Studierenden Inhalt und Methoden der Prüfungsmodule in den wesentlichen Zusammenhängen beherrschen und die erworbenen Kenntnisse, Fähigkeiten und Fertigkeiten selbständig anwenden können.
- (2) Die Prüfungsanforderungen sind an dem Inhalt der Lehrveranstaltungen und an den Kompetenzen zu orientieren, die für das betreffende Modul vorgesehen sind.
- (3) Eine Modulprüfung kann aus folgenden Leistungen bestehen:
  1. einer Klausur,
  2. einer mündlichen Prüfung,
  3. einer schriftlichen Hausarbeit,
  4. einer Projektarbeit,
  5. einer Prüfung, in der in einer Verknüpfung zwischen praktischen und theoretischen Anteilen eine Fähigkeit aktuell entwickelt und verwirklicht wird ("Performanzprüfung").
- (4) Prüfungsleistungen in einer Modulprüfung können innerhalb der ersten vier Semester durch gleichwertige Leistungen ersetzt werden, wenn sie in einer Einstufungsprüfung gemäß § 3 erbracht worden sind.
- (5) In Modulen, in denen ein Teil des Lehrstoffes in Praktika vermittelt wird, ist für die Vergabe von Credits die Teilnahme durch bestandene Teilmodulprüfung (Testat) nachzuweisen. Ein Testat wird sowohl für die regelmäßige Teilnahme an den Praktika als auch für regelmäßig abzuliefernde Praktikumsaufgaben ausgestellt (nach Vorgabe lt. Modulhandbuch, siehe Anlage 2).

- (6) Eine (Teil-) Modulprüfung ist bestanden, wenn die Prüfungsleistung mindestens als ausreichend bewertet worden ist.
- (7) Die Prüfenden legen gegenüber dem vorsitzenden Mitglied des Prüfungsausschusses in der Regel spätestens zwei Monate vor einem Prüfungstermin die Prüfungsform für alle Kandidatinnen und Kandidaten der jeweiligen Modulprüfung einheitlich und verbindlich fest. Im Fall einer Klausur gilt dies auch für die Zeit der Bearbeitung.

## **§ 11**

### **Klausurarbeiten und sonstige schriftliche Arbeiten**

- (1) In den Klausurarbeiten sollen Studierende nachweisen, dass sie in begrenzter Zeit und mit beschränkten Hilfsmitteln Probleme aus Gebieten des jeweiligen Moduls mit geläufigen Methoden der Fachrichtung erkennen und stringent zu einer Lösung finden können.
- (2) Eine Klausurarbeit findet unter Aufsicht statt. Über die Zulassung von Hilfsmitteln entscheiden die Prüfenden. Die Dauer einer Klausurarbeit soll 60 Minuten nicht unterschreiten und 180 Minuten nicht überschreiten.
- (3) Die Prüfungsaufgabe einer Klausurarbeit wird in der Regel von nur einer prüfenden Person gestellt. In fachlich begründeten Fällen, insbesondere wenn in einer Modulprüfung mehrere Fachgebiete zusammenfassend geprüft werden, kann die Prüfungsaufgabe auch von mehreren Prüfenden gestellt werden. In diesem Fall legen die Prüfenden die Gewichtung der Anteile an der Prüfungsaufgabe vorher gemeinsam fest.
- (4) Die Bewertung von Klausurarbeiten durch eine Prüferin oder einen Prüfer ist ausreichend. In den Fällen des Absatzes 3 Satz 2 bewerten die Prüfenden in der Regel nur den eigenen Aufgabenteil; Satz 1 bleibt unberührt.

## **§ 12**

### **Mündliche Prüfungen**

- (1) Durch mündliche Prüfungsleistungen soll die/der Studierende nachweisen, dass sie/er die Zusammenhänge des Prüfungsgebietes erkennt und spezielle Fragestellungen in diese Zusammenhänge einzuordnen vermag. Ferner soll festgestellt werden, ob die/der Studierende über ein breites Grundlagenwissen verfügt. Die Dauer der Prüfung beträgt je Studierender/Studierendem mindestens 15 Minuten und höchstens 45 Minuten.
- (2) Mündliche Prüfungen sind von mindestens zwei Prüfenden (Kollegialprüfung) oder von einer/einem Prüfenden in Gegenwart einer/eines sachkundigen Beisitzenden als Gruppenprüfung oder als Einzelprüfung abzunehmen. Hierbei wird jede/jeder Studierende in einer Modulprüfung im Regelfall nur von einer Person geprüft. Vor der Festsetzung der Note hat die prüfende Person die anderen an der Prüfung mitwirkenden Prüferinnen/Prüfer bzw. die/den sachkundigen Beisitzenden zu hören.
- (3) Die sachkundigen Beisitzenden haben während der Prüfung kein Fragerecht.
- (4) Die wesentlichen Gegenstände und Ergebnisse der Prüfung, insbesondere die für die Benotung maßgeblichen Tatsachen, sind in einem Protokoll festzuhalten. Das Ergebnis der Prüfung ist der/dem Studierenden im Anschluss an die mündliche Prüfung bekannt zu geben. Bei der Bekanntgabe des Ergebnisses sind die Bestimmungen des Datenschutzes zu beachten.
- (5) Studierende, die sich der gleichen Prüfung unterziehen wollen, werden nach Maßgabe der räumlichen Verhältnisse als Zuhörende zugelassen, sofern nicht widersprochen wird. Die Zulassung erstreckt sich nicht auf die Beratung und Bekanntgabe des Prüfungsergebnisses.

## **§ 13**

### **Hausarbeiten**

- (1) Hausarbeiten sind Ausarbeitungen, die in der Regel 20 Seiten nicht überschreiten und die im Rahmen einer Lehrveranstaltung oder in Verbindung mit einer Projektarbeit begleitend zu dieser erstellt werden. Sie können je nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden. § 12 Abs. 2 bis 5 sind auf den Fachvortrag entsprechend anzuwenden.
- (2) In Hausarbeiten sollen die Studierenden in begrenzter Zeit nachweisen, dass sie die Zusammenhänge des Moduls im jeweiligen Fachgebiet erkennen, spezielle Fragestellungen in diese Zusammenhänge einzuordnen vermögen und stringent fachspezifische Probleme lösen können.
- (3) Über Art, Umfang, zeitlichen Rahmen und Ausführung der Hausarbeit entscheidet die/der Lehrende im Rahmen der Maßgabe des Absatzes 1.  
Die Bewertung der Hausarbeit durch eine Prüferin oder einen Prüfer ist ausreichend.

- (4) Die Hausarbeit ist innerhalb einer von der/dem Lehrenden festzulegenden Frist beim zuständigen Prüfungsamt abzuliefern. Die Frist ist durch Aushang bekannt zu machen. Bei der Abgabe der Hausarbeit hat die/der Studierende zu versichern, dass sie/er ihre/seine Arbeit – bei einer Gruppenarbeit ihren/seinen gekennzeichneten Anteil der Arbeit – selbständig angefertigt und keine anderen als die angegebenen und bei Zitaten kenntlich gemachten Hilfsmittel benutzt hat. Der Abgabezeitpunkt der schriftlichen Hausarbeit ist aktenkundig zu machen. Bei Zustellung der Arbeit durch die Post ist der Zeitpunkt der Einlieferung bei der Post maßgebend. Wird die Hausarbeit nicht fristgemäß abgeliefert, gilt sie als mit „nicht ausreichend“ (5,0) bewertet.

#### **§ 14**

##### **Projektarbeiten**

- (1) Die Projektarbeit besteht aus einer schriftlichen Ausarbeitung, einer Softwareübergabe (Lösung) und ggfs. – nach Angabe der/des Lehrenden - einer Präsentation.
- (2) Ein Projekt ist eine Aufgabe, die von der/dem Lehrenden in Zusammenarbeit mit den Studierenden nach Möglichkeit interdisziplinär geplant und ausgewählt wird. Die Durchführung erfolgt möglichst selbständig unter Beratung durch einen oder mehrere Lehrende, die auch interdisziplinär zusammengestellt sein können. In ihnen werden konkrete Problemstellungen ganzheitlich und unter praxisnahen Bedingungen bearbeitet. Bei Gruppenarbeiten werden die inhaltliche und gleichmäßige Verteilung der Arbeitsinhalte an die Studierenden durch die Lehrenden vorgenommen.
- (3) Die Prüfungsleistungen der/des einzelnen Studierenden werden nach Abschluss des jeweiligen Semesters von der/dem zuständigen Lehrenden nach einer Auswahl aus den Kriterien
- Dokumentation
  - Qualität der Software-Übergabe (Lösung)
  - Ggf. Präsentation durch die einzelne Studierende/den einzelnen Studierenden
  - Ggf. Beitrag zum Teamergebnis bei einer Gruppenarbeit
  - Ggf. Teamfähigkeit
- bewertet. Die Ergebnisse werden in einer Liste erfasst.
- (4) Die Prüfung der Projektarbeit kann durch eine Präsentation von 30 bis 45 Minuten abgelegt werden. Bei Gruppenarbeiten sind von allen am jeweiligen Projekt beteiligten Studierenden die Einzelbeiträge und Ergebnisse vorzutragen. Die Präsentation findet in Gegenwart der/des Lehrenden, die/der die Projektarbeit begleitet hat, statt. § 12 Abs. 4 Satz 3 gilt entsprechend.

#### **§ 15**

##### **Performanzprüfungen**

- (1) In fachlich geeigneten Fällen kann eine Modulprüfung durch eine Performanzprüfung abgelegt werden. Im Rahmen einer Performanzprüfung werden von der oder dem Studierenden erworbenes Wissen und die angelegten Kompetenzen exemplarisch auf die zukünftige Berufspraxis angewendet. Es soll festgestellt werden, ob die Studierenden ihr theoretisches Wissen praktisch anwenden können.
- (2) Eine Performanzprüfung ist dadurch gekennzeichnet, dass sie sich aus zwei Anteilen (theoretisch und praktisch) zusammensetzt. Eine Teilleistung ist bestanden, wenn sie mindestens mit ausreichend bewertet worden ist. Der theoretische Anteil besteht aus einer schriftlichen oder mündlichen Prüfungsleistung entsprechend der §§ 11 und 12. Der praktische Anteil besteht aus praxisnahen Aufgaben im Rechnerlabor, welche im Laufe des Semesters absolviert und bewertet werden. Einzelne bestandene Anteile können auf die Folgesemester übertragen werden. Jede Teilleistung wird in der Regel von nur einer prüfenden Person durchgeführt.
- (3) Die wesentlichen Gegenstände und Ergebnisse der jeweiligen Teilleistung, insbesondere die für die Benotung maßgeblichen Tatsachen, sind in einem Protokoll festzuhalten. Das Ergebnis der Prüfung ist der oder dem Studierenden im Anschluss an die Teilleistung bekannt zu geben. Bei der Bekanntgabe des Ergebnisses sind die Bestimmungen des Datenschutzes zu beachten.
- (4) Die Gesamtnote ergibt sich als arithmetisches Mittel aus den Bewertungen der beiden bestandenen Einzelleistungen. Wenn eine Teilleistung endgültig nicht bestanden ist, gilt die gesamte Leistung als endgültig nicht bestanden.

#### **§ 16**

##### **Abzuleistende Modulprüfungen, Credits**

Der Studienplan legt fest, welche Pflicht- und welche Wahlpflichtmodule mit einer Prüfung abzuschließen sind. Er ordnet auch die entsprechenden Credits zu.

**§ 17****Fortschrittsregelung**

Es gilt ein semesterweise aufbauender Mindest-Leistungsfortschritt wie folgt:

Für die Anmeldung und Zulassung zu Modulprüfungen des 4. Semesters sind die bestandenen Modulprüfungen des 1. Semesters erforderlich.

Für die Anmeldung und Zulassung zu Modulprüfungen des 5. Semesters sind die bestandenen Modulprüfungen des 1. und 2. Semesters erforderlich.

Für die Anmeldung und Zulassung zu Modulprüfungen des 6. Semesters sind die bestandenen Modulprüfungen des 1., 2. und 3. Semesters erforderlich.

Die Zulassung zur Praxisphase regelt der § 24. Die Zulassung zur Bachelorarbeit wird in den entsprechenden §§ 28 bis 31 gesondert geregelt.

Kann die Studierende/der Studierende aus von ihr oder ihm nicht zu vertretenden Gründen eine Mindestforderung gemäß geltender Fortschrittsregelung nicht erfüllen, so entscheidet der Prüfungsausschuss auf Antrag über eine Absenkung des entsprechenden Mindestfortschritts.

**§ 18****Zulassung zu Klausuren und mündlichen Prüfungen**

- (1) An den jeweiligen Modulprüfungen darf nur teilnehmen, wer
  1. für den Studiengang eingeschrieben oder gemäß § 52 Abs. 1 HG als Zweithörender zugelassen ist,
  2. die nach § 3 und § 17 geforderten Voraussetzungen erfüllt,
  3. erforderliche Prüfungsvoraussetzungen gem. Modulhandbuch erbracht hat
  4. den Prüfungsanspruch in dem Studiengang oder in einem verwandten Studiengang nicht verloren hat.
- (2) Dem Antrag auf Zulassung zu einer Modulprüfung ist eine Erklärung darüber abzugeben, ob bei einer mündlichen Prüfung einer Zulassung von Zuhörenden widersprochen wird.
- (3) Der Antrag auf Zulassung zu einer Modulprüfung kann schriftlich beim Prüfungsamt bis zum Ablauf des achten Tages vor dem festgesetzten Prüfungstermin ohne Anrechnung auf die Zahl der möglichen Prüfungsversuche zurückgenommen werden, so dass eine Frist von sieben Tagen besteht.
- (4) Über die Zulassung entscheidet das vorsitzende Mitglied des Prüfungsausschusses und im Zweifelsfall der Prüfungsausschuss.
- (5) Die Zulassung ist zu versagen, wenn
  1. die im Absatz 1, 1 bis 3 genannten Voraussetzungen nicht erfüllt sind oder
  2. eine entsprechende Modulprüfung in einem Bachelorstudiengang oder in einem verwandten Studiengang endgültig nicht bestanden wurde. Dies gilt entsprechend für eine Bachelorprüfung im Geltungsbereich des Grundgesetzes.

Im Übrigen darf die Zulassung nur versagt werden, wenn die/der Studierende im Geltungsbereich des Grundgesetzes ihren/seinen Prüfungsanspruch im gleichen Studiengang durch Versäumen einer Wiederholungsfrist verloren hat.
- (6) Über die Zulassung bzw. Nicht-Zulassung ist die/der Studierende in der vom Prüfungsamt festgelegten Form zu informieren.

**§ 19****Zulassung zu sonstigen Prüfungen**

- (1) Teilnahmeberechtigt an sonstigen Prüfungen sind nur Studierende, welche die allgemeinen Zulassungsvoraussetzungen nach § 18 Abs. 1 erfüllen.
- (2) Der Antrag auf Zulassung zur einer Hausarbeit, einer Projektarbeit oder einer Performanzprüfung ist zu einem vom Prüfungsausschuss festgesetzten und bekannt gegebenen Termin schriftlich dem Prüfungsausschuss vorzulegen. Eine entsprechende Erklärung gem. § 18 Abs. 2 ist ggf. abzugeben.
- (3) § 18 Abs. 3 bis 6 gilt entsprechend.

**§ 20****Durchführung von Modulprüfungen**

- (1) Für die Modulprüfungen ist ein Prüfungstermin anzusetzen. Die Modulprüfungen sollen innerhalb eines Prüfungszeitraums stattfinden, der vom Prüfungsausschuss festgesetzt und bei Semesterbeginn oder zum Ende des vorhergehenden Semesters bekannt gegeben wird.

- (2) Der Prüfungstermin wird der/dem Studierenden rechtzeitig, spätestens zwei Wochen vor der betreffenden Prüfung, bekannt gegeben. Die Bekanntmachung durch Aushang ist ausreichend.
- (3) Die/der Studierende hat sich auf Verlangen der Aufsicht führenden Person mit einem amtlichen Ausweis auszuweisen.
- (4) Macht die/der Studierende durch ein ärztliches Zeugnis oder auf andere Weise glaubhaft, dass sie/er wegen ständiger körperlicher Behinderung nicht in der Lage ist, die Prüfung ganz oder teilweise in der vorgesehenen Form abzulegen, kann gestattet werden, gleichwertige Prüfungsleistungen in einer anderen Form zu erbringen. Es ist dafür zu sorgen, dass durch die Gestaltung der Prüfungsbedingungen eine Benachteiligung für behinderte Menschen nach Möglichkeit ausgeglichen wird. Im Zweifel können weitere Nachweise angefordert werden.
- (5) Das Prüfungsergebnis wird dem Prüfungsamt durch die Prüfende/den Prüfenden entsprechend der für die jeweilige Prüfungsform festgelegten Art und Weise innerhalb des in Absatz 6 festgelegten Zeitrahmens mitgeteilt.
- (6) Den Studierenden ist die Bewertung von Prüfungen und der Bachelorarbeit nach spätestens sechs Wochen mitzuteilen. Die Bekanntmachung durch Aushang ist ausreichend.

## § 21

### Bewertung von Prüfungsleistungen

- (1) Prüfungsleistungen sind durch Noten differenziert zu beurteilen. Die Noten für die einzelnen Prüfungsleistungen werden von den jeweiligen Prüfenden festgesetzt.
- (2) Sind mehrere Prüfende an einer Prüfung beteiligt, so bewerten sie die gesamte Prüfungsleistung gemeinsam, sofern nicht nachfolgend etwas anderes bestimmt ist. Bei nicht übereinstimmender Beurteilung ergibt sich die Note aus dem arithmetischen Mittel der Einzelbewertungen.
- (3) Für die Bewertung der Prüfungsleistungen sind folgende Noten zu verwenden:
  - 1 = sehr gut = eine hervorragende Leistung;
  - 2 = gut = eine Leistung, die erheblich über den durchschnittlichen Anforderungen liegt;
  - 3 = befriedigend = eine Leistung, die durchschnittlichen Anforderungen entspricht;
  - 4 = ausreichend = eine Leistung, die trotz ihrer Mängel noch den Anforderungen genügt;
  - 5 = nicht ausreichend = eine Leistung, die wegen erheblicher Mängel den Anforderungen nicht mehr genügt.

Zur weiteren Differenzierung der Bewertung können um 0,3 verminderte oder erhöhte Notenziffern gebildet werden; die Noten 0,7, 4,3, 4,7 und 5,3 sind ausgeschlossen.
- (4) Besteht eine Prüfung aus mehreren Prüfungsleistungen, errechnet sich die Note aus dem nach Credits gewichteten Durchschnitt (gewichtetes arithmetisches Mittel) der Noten der einzelnen Prüfungsleistungen. Die Note lautet:
 

bei einem Durchschnitt bis 1,5	= die Note „sehr gut“
bei einem Durchschnitt von 1,6 bis 2,5	= die Note „gut“
bei einem Durchschnitt von 2,6 bis 3,5	= die Note „befriedigend“
bei einem Durchschnitt von 3,6 bis 4,0	= die Note „ausreichend“
bei einem Durchschnitt ab 4,1	= die Note „nicht ausreichend“.

Hierbei werden Zwischenwerte nur mit der ersten Dezimalstelle berücksichtigt; alle weiteren Stellen hinter dem Komma werden ohne Rundung gestrichen.
- (5) Für jede bestandene Modulprüfung werden Credits nach Maßgabe der Anlagen 1 und 2 vergeben. (6) Das Praxismodul wird statt mit einer Note mit dem Ergebnis „mit Erfolg teilgenommen“ abgeschlossen.

## § 22

### Wiederholung von Prüfungsleistungen

- (1) Eine nicht bestandene Modulprüfung kann zweimal wiederholt werden. Die erste Wiederholungsprüfung wird im nächsten Semester angeboten. Die zweite Wiederholungsprüfung soll zum nächstmöglichen regulären Prüfungstermin erfolgen. Die nächstmöglichen Prüfungstermine werden im Prüfplan nach Festlegung durch den Prüfungsausschuss bekannt gegeben.
- (2) Die 2. Wiederholungsprüfung ist in jedem Fall von zwei Prüferinnen/Prüfern zu bewerten.
- (3) Modulprüfungen aus dem Wahlpflichtbereich mit teamorientierter Projektarbeit können erst wieder angeboten werden, wenn das Wahlfach stattgefunden hat. Die Modulprüfung kann hier auf Antrag in einem anderen Wahlfach aus der gleichen Vertiefungsrichtung wiederholt werden
- (4) Praxismodul und Bachelorarbeit können je einmal wiederholt werden.
- (5) Eine mindestens als „ausreichend“ bewertete Prüfungsleistung kann nicht wiederholt werden.

**§ 23****Versäumnis, Rücktritt, Täuschung, Ordnungsverstoß**

- (1) Eine Prüfungsleistung gilt als „nicht ausreichend“ (5,0) bewertet, wenn die/der Studierende zu einem Prüfungstermin ohne triftige Gründe nicht erscheint oder nach Beginn der Prüfung ohne triftige Gründe von der Prüfung zurücktritt oder die Prüfungsleistung nicht vor Ablauf der Prüfung erbringt. Satz 1 gilt entsprechend, wenn die Bachelorarbeit nicht fristgemäß abgeliefert wird. Wird die gestellte Prüfungsarbeit nicht bearbeitet, steht dies der Säumnis nach Satz 1 gleich. Belastende Entscheidungen sind den Betroffenen unverzüglich schriftlich mitzuteilen, zu begründen und mit einer Rechtsbehelfsbelehrung zu versehen.
- (2) Die für den Rücktritt oder das Versäumnis geltend gemachten Gründe müssen dem Prüfungsausschuss unverzüglich schriftlich angezeigt und glaubhaft gemacht werden. Bei Krankheit kann die Vorlage eines amtsärztlichen Attestes verlangt werden. Erkennt der Prüfungsausschuss die Gründe an, so kann die Zulassung der entsprechenden Prüfungsleistung erneut beantragt werden.
- (3) Versucht eine Studierende/ein Studierender, das Ergebnis einer Prüfungsleistung durch Täuschung oder Benutzung nicht zugelassener Hilfsmittel zu beeinflussen, gilt die betreffende Prüfungsleistung als „nicht ausreichend“ (5,0) bewertet. Wer als Studierende/Studierender den ordnungsgemäßen Ablauf der Prüfung stört, kann von der jeweiligen Aufsicht, in der Regel, nach Abmahnung von der Fortsetzung der Prüfungsleistung ausgeschlossen werden; in diesem Fall gilt die betreffende Prüfungsleistung als „nicht ausreichend“ (5,0) bewertet. Die Gründe für den Ausschluss sind aktenkundig zu machen. Wenn die/der Studierende davon ausgeschlossen wird, eine weitere Prüfungsleistung zu erbringen, kann sie/er verlangen, dass der Prüfungsausschuss diese Entscheidung überprüft. Dies gilt entsprechend auch bei den Feststellungen gemäß Satz 1.

**III. Praxisphase / Auslandssemester****§ 24 Praxisphase**

- (1) Der Bachelorstudiengang Informatik beinhaltet im 7. Semester eine berufspraktische Tätigkeit von 13 Wochen, deren Arbeitsaufwand 18 ECTS-Punkte beträgt. Auf Antrag wird zur Praxisphase zugelassen, wer mindestens die Module der ersten 4 Semester bestanden hat. Über die Zulassung entscheidet der Prüfungsausschuss.
- (2) Das Praxisprojekt soll die Studierenden an die berufliche Tätigkeit durch konkrete Aufgabenstellungen und praktische Mitarbeit heranführen, die mit den Zielen und Inhalten des Studienganges in einem fachlichen Zusammenhang stehen. Es soll insbesondere dazu dienen, die im bisherigen Studium erworbenen Kenntnisse und Fähigkeiten anzuwenden und die bei der praktischen Tätigkeit gemachten Erfahrungen zu reflektieren und auszuwerten.
- (3) Die/der betreuende Lehrende bescheinigt die Anerkennung der Praxisphase, wenn die Studierenden nach dem Zeugnis der Ausbildungsstätte die ihnen übertragenden Arbeiten zufriedenstellend ausgeführt haben und ein schriftlicher Bericht, der 13 Seiten Umfang nicht unterschreitet und 20 Seiten Umfang nicht überschreitet, innerhalb von 6 Wochen nach Abschluss der Praxisphase vorgelegt worden ist.
- (4) Bei Nichtanerkennung der Praxisphase kann die Praxisphase einmal wiederholt werden

**§ 25****Eignung der Praxisstelle**

- (1) Als Praxisstelle kommen alle Einrichtungen in Betracht, deren Aufgaben den ständigen Einsatz von Mitarbeitern mit der Qualifikation des Studienganges Informatik erlaubt. Die Einrichtungen müssen über Personal verfügen, die von ihrer Qualifikation her geeignet sind, die Studierenden während der Praxisphase zu betreuen. Die Einrichtungen müssen in der Lage sein, eine dem Ziel der Praxisphase entsprechende innerbetriebliche Tätigkeit sicherzustellen.
- (2) Die Eignung einer Praxisstelle wird nach Möglichkeit von einer/einem betreuenden Lehrenden des Fachbereichs vor Antritt der Praxisphase festgestellt und dem Prüfungsausschuss mitgeteilt. Anerkannte Praxisstellen können in eine im Fachbereich geführte Liste aufgenommen werden.
- (3) Die Studierenden können von sich aus eine Praxisstelle vorschlagen; eine Eignung soll gem. Abs. 2 festgestellt werden.
- (4) Ein entsprechendes Formular zur Anmeldung der Praxisphase hält der Fachbereich für die Studierenden bereit.

**§ 26****Betreuung der Studierenden in der Praxisphase**

Die Studierenden werden während der Praxisphase von einer Lehrkraft betreut. Wenigstens einmal während der Praxisphase sollte sich diese nach Absprache mit den Studierenden im Unternehmen einen Eindruck in die von Ihnen ausgeübte Tätigkeit verschaffen.

**§ 27****Fakultatives Auslandssemester**

- (1) Den Studierenden wird die Möglichkeit gegeben, an ausländischen Hochschulen zur Erweiterung ihres fachlichen Wissens, ihrer Sprachkenntnisse und ihrer interkulturellen Qualifikation ein Auslandssemester zu absolvieren.
- (2) Für die Möglichkeit, ein Urlaubssemester in Anspruch zu nehmen, wird auf die Einschreibungsordnung der Fachhochschule Bielefeld in der jeweils gültigen Fassung verwiesen. Für die Anerkennung der an der ausländischen Hochschule erbrachten Leistung gilt § 4 Absatz 2.

**IV. Bachelorarbeit****§ 28****Bachelorarbeit**

- (1) Die Bachelorarbeit hat zu zeigen, dass die/der Studierende befähigt ist, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus ihrem/seinem Fachgebiet, sowohl in ihren fachlichen Einzelheiten als auch in den fachübergreifenden Zusammenhängen nach wissenschaftlichen Methoden selbständig zu bearbeiten. Die Bachelorarbeit umfasst die selbständige Bearbeitung einer wissenschaftlich angewandten Problemstellung, deren Dokumentation sowie im Regelfall die Konzeption und Entwicklung einer Softwarelösung. Der Umfang der Bachelorarbeit soll 60 Textseiten nicht unterschreiten und 80 Textseiten nicht überschreiten.
- (2) Die Bachelorarbeit kann von jeder prüfenden Person, welche die Voraussetzungen gemäß § 9 erfüllt, ausgegeben und betreut werden. Auf Antrag der/des Studierenden kann der Prüfungsausschuss auch eine Honorarprofessorin oder einen Honorarprofessor oder mit entsprechenden Aufgaben betrauten Lehrenden gem. § 9 Abs. 1 mit der Betreuung beauftragen. Die Bachelorarbeit darf mit Zustimmung des Prüfungsausschusses in einer Einrichtung außerhalb der Hochschule durchgeführt werden. In diesem Fall kann der externe Betreuer, sofern er ebenfalls die Voraussetzungen des § 9 Abs. 1 erfüllt, als Zweitprüfer zugelassen werden.
- (3) Die Studierende/der Studierende reicht nach Abstimmung mit der/dem gewünschten Erst- und Zweitprüferin/-prüfer ein Thema für die Bearbeitung der Bachelorarbeit beim Prüfungsausschuss ein. Auf den Vorschlag der/des Studierenden ist nach Möglichkeit Rücksicht zu nehmen. Auf Antrag sorgt das vorsitzende Mitglied des Prüfungsausschusses dafür, dass die Studierenden rechtzeitig ein Thema für die Bachelorarbeit erhalten.
- (4) Die Bachelorarbeit kann auch in Form einer Gruppenarbeit zugelassen werden, wenn der als Prüfungsleistung zu bewertende Beitrag der/des einzelnen Studierenden aufgrund der Angabe von Abschnitten, Seitenzahlen oder anderen objektiven Kriterien, die eine eindeutige Abgrenzung ermöglichen, deutlich unterscheidbar und bewertbar ist und die Anforderungen nach Abs. 1 erfüllt. Die Bearbeitung der Bachelorarbeit als Gruppenarbeit erfordert eine Beantragung durch die Studierende/den Studierenden und eine Genehmigung von Seiten des Prüfungsausschusses.

**§ 29****Zulassung zur Bachelorarbeit**

- (1) Zur Bachelorarbeit wird zugelassen, wer die Modulprüfungen bis auf vier bestanden hat.
- (2) Der Antrag auf Zulassung ist schriftlich an den Prüfungsausschuss zu richten. Dem Antrag sind folgende Unterlagen beizufügen, sofern sie nicht bereits früher vorgelegt wurden:
  1. die Nachweise über die in Absatz 1 genannten Zulassungsvoraussetzungen,
  2. eine Erklärung über bisherige Versuche zur Bearbeitung einer Bachelorarbeit.
  3. ein im Fachbereich erhältliches Anmeldeformular, welches das zu bearbeitende Thema sowie die Unterschrift (Zustimmung) der gewünschten betreuenden Personen enthält.
- (3) Der Antrag auf Zulassung kann schriftlich bis zur Bekanntgabe der Entscheidung über den Antrag ohne Anrechnung auf die Zahl der möglichen Prüfungsversuche zurückgenommen werden.
- (4) Über die Zulassung entscheidet das vorsitzende Mitglied des Prüfungsausschusses und im Zweifelsfall der Prüfungsausschuss. Die Zulassung ist zu versagen, wenn

1. die in Absatz 1 genannte Voraussetzung nicht erfüllt ist oder
2. die Unterlagen unvollständig sind oder
3. eine in der Anlage 2 genannte Prüfung endgültig nicht bestanden wurde oder
4. eine entsprechende Bachelorarbeit ohne Wiederholungsmöglichkeit als "nicht ausreichend" bewertet worden ist.

Im Übrigen darf die Zulassung nur versagt werden, wenn die/der Studierende im Geltungsbereich des Grundgesetzes ihren/seinen Prüfungsanspruch im gleichen Studiengang durch Versäumen einer Wiederholungsfrist verloren hat.

### **§ 30**

#### **Ausgabe und Bearbeitung der Bachelorarbeit**

- (1) Der Prüfungsausschuss gibt die Bachelorarbeit aus und legt die Bearbeitungszeit fest. Als Zeitpunkt der Ausgabe gilt der Tag, an dem das Prüfungsamt das von den betreuenden Personen bestätigte Thema der Bachelorarbeit der Kandidatin oder dem Kandidaten bekannt gibt; der Zeitpunkt ist aktenkundig zu machen.
- (2) Die Bearbeitungszeit (Zeitraum von der Ausgabe bis zur Abgabe der Bachelorarbeit) beträgt höchstens drei Monate. Das Thema und die Aufgabenstellung müssen so beschaffen sein, dass die Bachelorarbeit innerhalb der vorgesehenen Frist abgeschlossen werden kann. Wird die Bachelorarbeit nicht fristgerecht abgegeben, gilt sie als mit „nicht ausreichend“ (5,0) bewertet. Im Ausnahmefall kann das Prüfungsamt auf einen vor Ablauf der Frist gestellten begründeten Antrag die Bearbeitungszeit um bis zu drei Wochen verlängern. Die Personen, welche die Bachelorarbeit betreuen, sollen zu dem Antrag gehört werden.
- (3) Das Thema der Bachelorarbeit kann nur einmal und nur innerhalb der ersten zwei Wochen der Bearbeitungszeit ohne Angabe von Gründen zurückgegeben werden. Im Fall der Wiederholung gemäß § 22 ist die Rückgabe nur zulässig, wenn bei der Anfertigung der ersten Bachelorarbeit von dieser Möglichkeit kein Gebrauch gemacht worden ist.

### **§ 31**

#### **Abgabe und Bewertung der Bachelorarbeit**

- (1) Die Bachelorarbeit ist fristgemäß beim Prüfungsamt abzuliefern. Der Zeitpunkt der Abgabe ist aktenkundig zu machen; bei Zustellung der Arbeit durch die Post ist der Zeitpunkt der Einlieferung bei der Post maßgebend. Bei der Abgabe der Bachelorarbeit ist schriftlich zu versichern, dass die Arbeit - bei einer Gruppenarbeit der entsprechend gekennzeichnete Anteil der Arbeit - selbständig angefertigt wurde und keine anderen als die angegebenen und bei Zitaten kenntlich gemachten Quellen und Hilfsmittel benutzt worden sind.
- (2) Die Bachelorarbeit ist von zwei Personen zu bewerten, von denen mind. eine die Bachelorarbeit betreut haben soll. Bei Ausfall einer prüfenden Person wird die Vertretung vom Prüfungsausschuss bestimmt. Der Erstprüfer soll grundsätzlich der Professorenschaft angehören. Bei nicht übereinstimmender Bewertung durch die Prüfenden soll die Note der Bachelorarbeit aus dem arithmetischen Mittel der Einzelbewertungen gebildet werden, wenn die Differenz der beiden Noten weniger als 2,0 beträgt. Beträgt die Differenz 2,0 oder mehr, wird vom Prüfungsausschuss eine dritte prüfende Person bestimmt. In diesem Fall ergibt sich die Note der Bachelorarbeit aus dem arithmetischen Mittel der beiden besseren Einzelbewertungen. Die Bachelorarbeit kann jedoch nur dann als "ausreichend" (4,0) oder besser bewertet werden, wenn mindestens zwei der Noten "ausreichend" (4,0) oder besser sind. Alle Bewertungen sind schriftlich zu begründen.

## **V. Zusatzmodule, Bachelorprüfung**

### **§ 32**

#### **Zusatzmodule**

Die Studierenden können sich in weiteren als den vorgeschriebenen Modulen einer Prüfung unterziehen. Das Ergebnis dieser Modulprüfungen wird auf Antrag in das Zeugnis aufgenommen, jedoch bei der Festsetzung der Gesamtnote nicht berücksichtigt.

### **§ 33**

#### **Bachelorprüfung**

Das Studium wird mit der Bachelorprüfung abgeschlossen. Die Bachelorprüfung gliedert sich in studienbegleitende Modulprüfungen, die Praxisphase und die Bachelorarbeit.

**§ 34****Ergebnis der Bachelorprüfung**

- (1) Die Bachelorprüfung ist bestanden, wenn 210 Credits erreicht werden.
- (2) Die Bachelorprüfung ist nicht bestanden, wenn
  - die Gesamtnote nicht mindestens „ausreichend“ (4,0) ist oder
  - die Bachelorarbeit im zweiten Versuch nicht bestanden ist oder als nicht bestanden gilt.
- (3) Wird die Bachelorprüfung nicht bestanden, ist ein Bescheid zu erteilen, der mit einer Belehrung über den Rechtsbehelf zu versehen ist.
- (4) Studierende, welche die Hochschule ohne Studienabschluss verlassen, erhalten auf Antrag ein Zeugnis über die erbrachten Studien- und Prüfungsleistungen gemäß § 66 Abs. 4 HG.

**§ 35****Zeugnis, Gesamtnote, Bachelorurkunde, Diploma Supplement**

- (1) Über die bestandene Bachelorprüfung wird unverzüglich, möglichst innerhalb von sechs Wochen nach der letzten Prüfungsleistung, ein Zeugnis ausgestellt. Das Zeugnis enthält die Noten und Credit Points der Modulprüfungen, das Thema und die Note der Bachelorarbeit sowie die Gesamtnote der Bachelorprüfung. In dem Zeugnis wird ferner das erfolgreich abgeleistete Praxisprojekt aufgeführt.
- (2) Zur Ermittlung der Gesamtnote für das Bachelor-Studium werden die Noten für die einzelnen benoteten Prüfungsleistungen mit den jeweiligen ausgewiesenen Credits multipliziert. Die Summe der gewichteten Noten wird anschließend durch die Gesamtzahl der einbezogenen Credits dividiert.
- (3) Das Zeugnis ist von dem vorsitzenden Mitglied des Prüfungsausschusses zu unterzeichnen und trägt das Datum des Tages, an dem die letzte Prüfungsleistung erbracht worden ist.
- (4) Für die Umrechnung der Abschlussnote in ECTS-Grades wird bei Vorliegen einer ausreichend großen Kohorte die folgende Tabelle zugrunde gelegt:
  - A = die besten 10%
  - B = die nächsten 25%
  - C = die nächsten 30%
  - D = die nächsten 25%
  - E = die nächsten 10%
  - FX/F = nicht bestanden - es sind (erhebliche) Verbesserungen erforderlich.
- (5) Gleichzeitig mit dem Zeugnis erhält die Kandidatin/der Kandidat die Bachelorurkunde mit dem Datum des Zeugnisses. Darin wird die Verleihung des Bachelorgrades gemäß § 2 Abs. 6 beurkundet. Die Bachelorurkunde wird von der Präsidentin bzw. dem Präsidenten der Fachhochschule Bielefeld unterzeichnet und mit deren Siegel versehen.
- (6) Zusätzlich erhält die Kandidatin/der Kandidat ein in englischer Sprache ausgestelltes Diploma Supplement mit dem Datum des Zeugnisses. Das Diploma Supplement wird vom vorsitzenden Mitglied des Prüfungsausschusses unterzeichnet.
- (7) Auf Antrag ist eine englischsprachige Fassung der Urkunde beizufügen (§ 66 Abs. 3 HG).

**VI. Schlussbestimmungen****§ 36****Einsicht in die Prüfungsakte**

- (1) Nach Abschluss des Prüfungsverfahrens wird der/dem Studierenden auf Antrag Einsicht in ihre/seine schriftlichen Prüfungsarbeiten, die darauf bezogenen Gutachten der Prüferinnen/Prüfer und in die Prüfungsprotokolle gewährt.
- (2) Die Einsichtnahme ist binnen eines Jahres nach Aushändigung des Prüfungszeugnisses oder des Bescheides über die nicht bestandene Bachelorprüfung zu beantragen. § 32 des Verwaltungsvorgangsgesetzes für das Land Nordrhein-Westfalen über die Wiedereinsetzung in den vorigen Stand gilt entsprechend. Der Antrag ist bei dem vorsitzenden Mitglied des Prüfungsausschusses zu stellen. Dieser bestimmt Ort und Zeit der Einsichtnahme.
- (3) Die Einsichtnahme in die Prüfungsunterlagen, die sich auf eine Modulprüfung oder eine ergänzende Studienleistung beziehen, wird auf Antrag bereits nach Ablegung der jeweiligen Prüfung gestattet. Der Antrag ist binnen eines Monats nach Bekanntgabe des Prüfungsergebnisses zu stellen.

### **§ 37**

#### **Ungültigkeit von Prüfungen**

- (1) Hat eine Studierende/ein Studierender bei einer Prüfung getäuscht und wird diese Tatsache erst nach Aushändigung des Zeugnisses und der Urkunde bekannt, so kann der Prüfungsausschuss nachträglich die betroffenen Noten entsprechend berichtigen und die Prüfung ganz oder teilweise für nicht bestanden erklären.
- (2) Waren die Voraussetzungen für die Zulassung zu einer Prüfung nicht erfüllt, ohne dass die/der Studierende hierüber täuschen wollte und wird diese Tatsache erst nach Aushändigung des Zeugnisses und der Urkunde bekannt, so wird dieser Mangel durch das Bestehen der Prüfung geheilt. Wurde die Zulassung vorsätzlich zu Unrecht erwirkt, so entscheidet der Prüfungsausschuss unter Beachtung des Verwaltungsverfahrensgesetzes des Landes Nordrhein-Westfalen über die Rechtsfolgen.
- (3) Den Betroffenen ist vor einer Entscheidung Gelegenheit zur Äußerung zu geben.
- (4) Das unrichtige Prüfungszeugnis und die Urkunde sind einzuziehen und gegebenenfalls neu zu erteilen. Eine Entscheidung nach Absatz 1 und Absatz 2 Satz 2 ist nach einer Frist von fünf Jahren ab dem Datum des Prüfungszeugnisses und der Urkunde ausgeschlossen.

### **§ 38**

#### **In-Kraft-Treten, Veröffentlichung**

Diese Bachelorprüfungsordnung wird im Verkündungsblatt der Fachhochschule Bielefeld – Amtliche Bekanntmachungen – bekannt gegeben. Sie tritt einen Tag nach ihrer Veröffentlichung in Kraft.

-----  
Ausgefertigt aufgrund des Beschlusses des Gründungsdekans des Fachbereichs Technik (im Aufbau) der Fachhochschule Bielefeld.

Bielefeld, 19. Mai 2014

Die Präsidentin  
der Fachhochschule Bielefeld

gez. Rennen-Allhoff

Prof. Dr. B. Rennen-Allhoff



**Anlage 2 Studienverlaufsplan**

<b>1. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Einführung in das Berufsfeld	5	4	2		2	-
Mathematik für Informatiker I	8	6	3	-	3	-
Objektorientierte Programmierung	7	4	2	-	-	2
Einführung in die Programmierung mit Skriptsprachen	5	4	2	-	-	2
Technische Informatik	5	4	2	-	2	-
<b>Summen</b>	<b>30</b>	<b>22</b>	<b>11</b>	<b>-</b>	<b>7</b>	<b>4</b>
<b>2. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Mathematik für Informatiker II	8	6	3	-	3	-
Programmiermethoden	7	5	2	-	-	3
Algorithmen und Datenstrukturen	5	4	2	-	-	2
Theoretische Informatik	5	4	2	-	2	-
Useability und Datenvisualisierung	5	4		4	-	-
<b>Summen</b>	<b>30</b>	<b>23</b>	<b>9</b>	<b>4</b>	<b>5</b>	<b>5</b>
<b>3. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Software Engineering	7	4	2	-	-	2
Verteilte Systeme und Kommunikationsnetze	5	4	2	-	-	2
Datenbanken	5	4	2	-	-	2
Systemprogrammierung	8	6	3	-	-	3
Technical English	5	4	-	4	-	-
<b>Summen</b>	<b>30</b>	<b>22</b>	<b>9</b>	<b>4</b>	<b>-</b>	<b>9</b>
<b>4. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Softwareprojekt	5	4	-	-	-	4
Betriebssysteme	5	4	2	-	-	2
Embedded Systems	5	4	2	-	-	2
Computergraphik	5	4	2	-	-	2
Webbasierte Anwendungen	5	4	2	-	-	2
SW-Projektmanagement	5	4	-	4	-	-
<b>Summen</b>	<b>30</b>	<b>24</b>	<b>8</b>	<b>4</b>	<b>-</b>	<b>12</b>
<b>5. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Grundlagen der BWL	5	4	2	-	2	-
Wahlfach aus Liste 1	5	4	2	-	-	2
Wahlfach aus Liste 1	5	4	2	-	-	2
Wahlfach aus Liste 2	15	6	2	-	-	4
<b>Summen</b>	<b>30</b>	<b>18</b>	<b>8</b>	<b>-</b>	<b>2</b>	<b>8</b>
<b>6. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
IT-Recht	5	4	-	4	-	-
Fachseminar	5	4	-	4	-	-
Wahlfach aus Liste 1	5	4	2	-	-	2

Wahlfach aus Liste 2	15	6	2	-	-	4
<b>Summen</b>	<b>30</b>	<b>18</b>	<b>4</b>	<b>8</b>	<b>-</b>	<b>6</b>
<b>7. Semester</b>	<b>cps</b>	<b>SWS</b>	<b>V</b>	<b>SU</b>	<b>Ü</b>	<b>P</b>
Praxisphase	18	-	-	-	-	-
Bachelorarbeit	12	-	-	-	-	-
		-	-	-	-	-
<b>Summen</b>	<b>30</b>	<b>-</b>	<b>-</b>	<b>-</b>	<b>-</b>	<b>-</b>
<b>Gesamtsummen</b>	<b>210</b>	<b>122</b>	<b>49</b>	<b>20</b>	<b>14</b>	<b>38</b>

**Liste 1****„Methoden der Informatik“****(2 V, 2 P)**

- Komponentenbasierte SW-Entwicklung
- Spezielle Methoden der Programmierung
- Mustererkennung & Bildverarbeitung
- Ausgewählte Konzepte von Datenbanksystemen
- Künstliche Intelligenz
- Sicherheit und Zuverlässigkeit
- Methoden der Computergraphik

**Liste 2****„Anwendungen der Informatik“****(2 V, 4 P)**

(Vertiefungsprojekt in Teamarbeit, von einem oder mehreren Professoren auch fachübergreifend betreut.)

- Anwendungen der Computergraphik
- Softwarequalität
- Webengineering
- Mobile Applikationen
- Softwareengineering für Embedded Systems
- Datenbankanwendungen
- Anwendungen der künstlichen Intelligenz
- Rich Client Application Development

In jedem Semester stehen pro Liste mindestens 3 Wahlmodule zur Auswahl.

Im Vorfeld findet eine Bedarfsanalyse durch Befragung der Studierenden statt.

Um die Lehrkapazitäten effektiv einsetzen zu können, wird ein Wahlmodul nur angeboten, wenn sich mind. 7 Studierende dafür anmelden.

Die Fächervielfalt und Wahlmöglichkeiten werden gewährleistet, indem sich Listenfächer höchstens im Jahresrhythmus wiederholen.

Änderungen bleiben vorbehalten. Es gilt die jeweils aktuelle Fassung des Studienverlaufsplans.

**Anlage 3:****Modulhandbuch für den Bachelor-Studiengang Informatik****Inhaltsverzeichnis**

Einführung in das Berufsfeld .....	194
Mathematik für Informatiker I .....	196
Objektorientierte Programmierung .....	198
Einführung in die Programmierung mit Skriptsprachen .....	200
Technische Informatik .....	202
Mathematik für Informatiker II.....	204
Programmiermethoden .....	206
Algorithmen und Datenstrukturen .....	208
Theoretische Informatik .....	210
Useability und Datenvisualisierung .....	212
Software Engineering.....	214
Verteilte Systeme und Kommunikationsnetze.....	216
Datenbanken.....	218
Systemprogrammierung.....	220
Technical English.....	223
Softwareprojekt.....	225
Betriebssysteme.....	227
Embedded Systems .....	229
Computergrafik.....	231
Webbasierte Anwendungen .....	233
SW-Projektmanagement .....	235
Grundlagen der BWL .....	237
Wahlfach aus Liste 1 "Methoden der Informatik" Komponentenbasierte SW-Entwicklung .....	239
Wahlfach aus Liste 1 „Methoden der Informatik“ Spezielle Methoden der Programmierung.....	242
Wahlfach aus Liste 1 "Methoden der Informatik" Mustererkennung & Bildverarbeitung.....	244
Wahlfach aus Liste 1 „Methoden der Informatik“ Ausgewählte Konzepte von DB-systemen .....	246
Wahlfach aus Liste 1 „Methoden der Informatik“ Künstliche Intelligenz.....	248
Wahlfach aus Liste 1 "Methoden der Informatik" Sicherheit und Zuverlässigkeit.....	250
Wahlfach aus Liste 1 „Methoden der Informatik“ Methoden der Computergrafik .....	252
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Anwendungen der Computergrafik .....	254
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Software Qualität .....	256
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Webengineering.....	258
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Mobile Applikationen.....	260
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Softwareengineering für Embedded Systems .....	262
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Datenbankanwendungen.....	264
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Anwendungen der künstlichen Intelligenz.....	266
Wahlfach aus Liste 2 „Anwendungen der Informatik“ Rich Client Application Development .....	268

IT-Recht .....	270
Fachseminar.....	272
Praxisphase .....	274
Bachelorarbeit.....	276

<b>Einführung in das Berufsfeld</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>1.1</b>	150 h	5 cps	1. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Übung 2 SWS	<b>Kontaktzeit</b> 4 SWS /60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Es sollen die Wurzeln und Entwicklungsgeschichte der Informatik verstanden werden. Die Studierenden sollen die wechselseitigen Einwirkungen der Gesellschaft auf die Informatik und umgekehrt erkennen und selbst lernen, dazu Position zu beziehen. Die Studierenden sollen vielfältige Berufsbilder von Informatikern aus verschiedenen Branchen kennen lernen. Das Fachgebiet soll in seiner Vielfalt und Breite und den begrifflichen Teilgebieten verstanden werden. Die Studierenden sollen befähigt werden in ihrem nachfolgenden Studium eine gezielte Fächerwahl zu treffen, unter Beachtung ihrer Stärken und Neigungen und im Hinblick auf ihr zukünftig angestrebtes Berufsfeld. Die Studierenden sollen in Ihrer Studienwahl bestärkt werden und für ein erfolgreiches Studium motiviert werden.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Geschichte der Informatik; Errungenschaften der Informatik</li> <li>• Informatik und Gesellschaft</li> <li>• Teilgebiete der Informatik</li> <li>• Berufsbilder von Informatikern</li> <li>• Branchen und Bereiche für Absolventen der Informatik</li> <li>• Persönliche Entwicklungsperspektiven für Absolventen der Informatik</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminar, Exkursionen				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> -				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				
<b>7</b>	<b>Prüfungsformen</b>  Klausur				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Bestandene Klausur				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Grit Behrens, Prof. Dr. Christoph Thiel
<b>12</b>	<b>Sonstige Informationen</b> -

<b>Mathematik für Informatiker I</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>1.2</b>	240 h	8 cps	1. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 3 SWS Übung 3 SWS		<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 150 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden erwerben den korrekten Gebrauch der mathematischen Grundbegriffe und Grundlagen der Mengenlehre und Aussagenlogik. Sie beherrschen die wichtigsten Beweisverfahren und können Zählprinzipien der Kombinatorik anwenden. Es wird die Fähigkeit, mit Vektoren und Matrizen zu rechnen, sowie lineare Gleichungssysteme zu lösen, erlangt. Darüber hinaus erlernen sie den sicheren Umgang mit Funktionen und beherrschen die Differential- und Integralrechnung.				
<b>3</b>	<b>Inhalte</b>  <b>Grundlagen</b> <ul style="list-style-type: none"> <li>• Zahlen</li> <li>• Mengenlehre</li> <li>• Aussagenlogik</li> <li>• Vollständige Induktion</li> <li>• Kombinatorik</li> </ul> <b>Lineare Algebra</b> <ul style="list-style-type: none"> <li>• Vektoren und Vektorräume</li> <li>• Matrizen und lineare Abbildungen</li> <li>• Lineare Gleichungssysteme</li> </ul> <b>Analysis I</b> <ul style="list-style-type: none"> <li>• Folgen und Reihen</li> <li>• Reelle Funktionen einer Variablen</li> <li>• Differentialrechnung</li> <li>• Integralrechnung</li> </ul>				
<b>4</b>	<b>Lehrformen</b>  Vorlesung, Übung				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> - <b>Inhaltlich:</b> -				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				
<b>7</b>	<b>Prüfungsformen</b>  Klausur				

<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Klausur
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 8/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Kerstin Müller, Dipl. Inf. Birgit Christina George, Prof. Dr. Christoph Thiel
<b>12</b>	<b>Sonstige Informationen</b> <b>Literatur</b> Hartmann, Peter: Mathematik für Informatiker, Vieweg. Manfred Brill: Mathematik für Informatiker, Hanser Verlag Bronstein, Semendjajev: Taschenbuch der Mathematik

<b>Objektorientierte Programmierung</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>1.3</b>	210 h	7 cps	1. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 150 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden sollen befähigt werden, für einfache Problemstellungen objektorientierte Lösungen zu finden. Die Studierenden lernen den praktischen Umgang mit Programmierwerkzeugen. Selbst gefundene Lösungskonzepte können von den Studierenden modelliert und implementiert werden. Die Studierenden lernen, Standardkomponenten und -bibliotheken in ihren eigenen Implementierungen zielgerichtet einzusetzen und zu nutzen. Es wird die Programmiersprache Java erlernt.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung in objektorientiertes Design, einfache Klassendiagramme</li> <li>• Einführung des Klassen- und Objektbegriffs</li> <li>• Nutzung einer Entwicklungsumgebung und eines Debuggers</li> <li>• Datentypen und Kontrollstrukturen</li> <li>• Einführung in die Vererbung und Polymorphie</li> <li>• Nutzung von Interfaces vs. abstrakte Klassen</li> <li>• Einführung in die Ausnahmebehandlung</li> <li>• Einführung von Collection-Klassen</li> <li>• Einführung in Swing</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum: die lt. Studienverlaufsplan 2 SWS Praktika werden durch 1 SWS tutorbetreutes Praktikum vor- bzw. nachbereitet				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal: -</b>  <b>Inhaltlich: -</b>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				
<b>7</b>	<b>Prüfungsformen</b>  Performanzprüfung				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Bestandene Teilleistung der Performanzprüfung				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 7/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Jörg Brunsmann
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"><li>• Krüger, G.: "Handbuch der Java-Programmierung", Addison-Wesley, 2011</li><li>• Ullenboom, C.: "Java ist auch eine Insel", Galileo Computing, 2011</li></ul>

<b>Einführung in die Programmierung mit Skriptsprachen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>1.4</b>	150 h	5 cps	1. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS / 60 h	<b>Selbst-studium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können <ul style="list-style-type: none"> <li>• XML-Dokumente erstellen und verarbeiten</li> <li>• Sprachkonzepte von Skriptsprachen in eigenen Applikationen anwenden</li> </ul>				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung: Bedeutung der XML-Datenformate, Bedeutung von Skriptsprachen</li> <li>• XML: Geschichte, Eigenschaften, Anwendungsbeispiele</li> <li>• Aufbau von XML-Datenformaten</li> <li>• Skriptsprachen: Anwendungsgebiete, Sprachkonzepte, z. B. Python</li> <li>• Erstellung von XML-basierten Programmoberflächen mithilfe von Skriptsprachen</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal: -</b> <b>Inhaltlich: Keine</b>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b> -				
<b>7</b>	<b>Prüfungsformen</b> Performanzprüfung				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Teilleistungen der Performanzprüfung				
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -				
<b>10</b>	<b>Stellenwert der Note für die Endnote</b>				

	5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dipl.-Inf. B.C. George
<b>12</b>	<b>Sonstige Informationen</b>

<b>Technische Informatik</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>1.5</b>	150 h	5 cps	1. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Übung 2 SWS	<b>Kontaktzeit</b> 4 SWS /60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die grundlegenden Begriffe und Konzepte der technischen Informatik kennen lernen. <ul style="list-style-type: none"> <li>• Grundbegriffe der technischen Informatik</li> <li>• Prinzipien der Computerarchitektur</li> <li>• Grundlegende Kenntnisse über den Aufbau und die Funktionsweise von Digitalrechnern</li> <li>• Einführung einer hardwarenahen Programmiersprache (z.B. C, Assembler)</li> </ul>				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Grundlagen Elektrotechnik (Gleichstromkreis)               <ul style="list-style-type: none"> <li>○ Ohmsches Gesetz</li> <li>○ Kirchhoffsche Sätze</li> </ul> </li> <li>• Grundlagen Digitaltechnik               <ul style="list-style-type: none"> <li>○ Boolesche Algebra, Normalform</li> <li>○ Schaltnetze</li> <li>○ Schaltwerke</li> </ul> </li> <li>• Grundlagen Rechnerarchitekturen               <ul style="list-style-type: none"> <li>○ Einführung und Überblick</li> <li>○ Aufbau von Digitalrechnern</li> <li>○ Speicher und Peripherie</li> <li>○ Bauformen und Anschlüsse</li> </ul> </li> <li>• Hardwarenahe Programmierung               <ul style="list-style-type: none"> <li>○ Entwicklungsumgebung, Cross-Compiler</li> <li>○ Grundlagen der hardwarenahen Programmierung</li> </ul> </li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Übung				
<b>5</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal: -</b> <b>Inhaltlich: -</b>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Keine				
<b>7</b>	<b>Prüfungsformen</b> Klausur				

<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Klausur
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dipl.-Ing. Angela Kreienkamp (MPrComp)
<b>12</b>	<b>Sonstige Informationen</b> Literatur (z.B.): <ul style="list-style-type: none"><li>• Computerarchitektur, A. Tanenbaum</li><li>• Mikrocomputertechnik mit Controllern der Atmel ..., G. Schmitt</li><li>• Mikrocontroller und Mikroprozessoren, T. Ungerer</li><li>• Digitaltechnik - Eine praxisnahe Einführung, A. Biere</li><li>• Technische Informatik 1, W. Schiffmann</li></ul>

<b>Mathematik für Informatiker II</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>2.1</b>	240 h	8 cps	2. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 3 SWS Übung 3 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 150 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden erlernen den sicheren Umgang mit Funktionen mehrerer Variablen und beherrschen die zugehörige Differential- und Integralrechnung. Sie erwerben ein Grundverständnis von linearen Differentialgleichungen. Relevante Zusammenhänge aus dem Bereich der Numerik sind bekannt, und elementare numerische Verfahren können durch die Studierenden angewandt werden. Im Rahmen der Veranstaltung werden die Studierenden befähigt, elementare stochastische Probleme mit geeigneten Methoden zu lösen. Darüber hinaus entwickeln die Studierenden ein Verständnis für wahrscheinlichkeitstheoretischer Grundbegriffe und kennen wichtige Verteilungen und ihre Bedeutung, sowie grundlegende statistische Methoden.				
<b>3</b>	<b>Inhalte</b>  <b>Analysis II</b> <ul style="list-style-type: none"> <li>• Lokale und globale Approximation</li> <li>• Differentialgleichungen</li> <li>• Reellwertige Funktionen mit mehrerer Variablen</li> <li>• Differential- und Integralrechnung für Funktionen mehrerer Variablen</li> </ul> <b>Numerik</b> <ul style="list-style-type: none"> <li>• Fehler und Fehlerfortpflanzung</li> <li>• Elementare numerische Verfahren</li> <li>• Optimierung</li> </ul> <b>Diskrete Wahrscheinlichkeitsrechnung und Statistik</b> <ul style="list-style-type: none"> <li>• Klassische Wahrscheinlichkeit</li> <li>• Bedingte Wahrscheinlichkeit</li> <li>• Zufallsgrößen</li> <li>• Verteilungen</li> <li>• Statistik</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Übung				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Mathematik 1				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				

<b>7</b>	<b>Prüfungsformen</b> Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Klausur
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 8/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Kerstin Müller, Dipl. Inf. Birgit Christina George, Prof. Dr. Christoph Thiel
<b>12</b>	<b>Sonstige Informationen</b> <b>Literatur</b> Hartmann, Peter: Mathematik für Informatiker, Vieweg. Bronstein, Semendjajev: Taschenbuch der Mathematik.

<b>Programmiermethoden</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>2.2</b>	210 h	7 cps	2. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS  Praktikum 3 SWS, davon 1 SWS Tutor betreut	<b>Kontaktzeit</b>  5 SWS/75 h	<b>Selbststudium</b>  135 h	<b>geplante Gruppengröße</b>  Vorlesung 60  Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden kennen und beherrschen ein umfassendes Instrumentarium an Techniken und Lösungsmustern zur Softwareentwicklung in der Programmiersprache Java. Die Fähigkeit zum Entwickeln von eigenen Bibliotheken und komplexen Anwendungen in einzelnen Komponenten wurde erworben. Standardarchitekturmuster werden beherrscht.				
<b>3</b>	<b>Inhalte</b>  <ul style="list-style-type: none"> <li>- Modellierung, Pattern, OOD: <ul style="list-style-type: none"> <li>■ Vertiefung objektorientierter Entwurf/Design</li> <li>■ Vertiefung Klassendiagramme mit UML</li> <li>■ Einführung in Basis-Entwurfsmuster (Strategy, Decorator, Observer, MVC)</li> <li>■ Vererbung vs. Komposition vs. Generics</li> </ul> </li>   <li>- Programmiermethoden: <ul style="list-style-type: none"> <li>■ Einführung in Versionsverwaltung am Beispiel Git</li> <li>■ Einführung in Coding Conventions und Metriken, CheckStyle</li> <li>■ Einführung in das Refactoring mit Eclipse</li> <li>■ Systematischer Unit-Test mit JUnit</li> <li>■ Systematische Dokumentation mit JavaDoc</li> <li>■ Buildsysteme (ANT, Maven)</li> <li>■ Einführung in die Nutzung von CI</li> </ul> </li>   <li>- Fortgeschrittene Java-Kenntnisse: <ul style="list-style-type: none"> <li>■ Annotationen und Enumerationen</li> <li>■ Vertiefung Fehler- und Ausnahmebehandlung, Exception-Hierarchie</li> <li>■ Reguläre Ausdrücke mit Java</li> <li>■ Generische Programmierung (Generics)</li> <li>■ Logging unter Java</li> <li>■ Konfiguration (Properties, Preferences, CLI)</li> <li>■ I/O: File, Streams</li> <li>■ Anonyme und innere Klassen</li> </ul> </li> </ul>				

	<ul style="list-style-type: none"> <li>■ Interfaces: Cloneable, Comparable, Iterable, Serializable</li> <li>■ Beziehung zwischen equals, hashCode und compareTo</li> <li>■ Nutzung des Java Collections Frameworks</li> <li>■ Ereignisorientierte Programmierung (Swing oder JavaFX, Java2D), Modelle und Event-Hierarchien</li> <li>■ Einführung in parallele Programmierung mit Java: Threads, synchronize, wait, notify</li> <li>■ Nutzung von Softwarebibliotheken (APIs): beispielsweise Apache CLI, Apache POI, JFreeChart</li> </ul>
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Objektorientierte Programmierung
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine
<b>7</b>	<b>Prüfungsformen</b>  Performanzprüfung
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Bestandene Teilleistung der Performanzprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b>  -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b>  7/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b>  Prof. Dr. Carsten Gips
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Deitel, Deitel: "Java – How to Program", Pearson Education Limited, 2011</li> <li>• Bloch, J.: "Effective Java: A Programming Language Guide", Addison-Wesley, 2008</li> <li>• Krüger, G.: "Handbuch der Java-Programmierung", Addison-Wesley, 2011</li> </ul>

<b>Algorithmen und Datenstrukturen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>2.3</b>	150 h	5 cps	2. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS / 60h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Die Studierenden sollen einige wichtige klassische Algorithmen zum Sortieren und Suchen in verschiedenen Datenstrukturen kennenlernen.</p> <p>Sie sollen Erfahrungen sammeln, wie man Algorithmen schreiben, lesen und wie man ihren dynamischen Ablauf darstellen kann.</p> <p>Sie sollen üben, genau und verständlich über Algorithmen zu sprechen und zu schreiben. Sie sollen wichtige Fachbegriffe in Ihr aktives Vokabular aufnehmen.</p> <p>Sie sollen theoretisch und praktisch etwas über das Verhältnis von abstrakten Algorithmen und konkreter Programmierung erfahren.</p> <p>Sie sollen Ihre Fertigkeiten in qualitativ guter Programmierung in Java erweitern.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>• Begrifflichkeiten (Algorithmus, Komplexität)</li> <li>• Gleichwertige Lösungen <ul style="list-style-type: none"> <li>○ Maximale Teilsumme</li> <li>○ Komplexitätsformel</li> <li>○ Datenstrukturen (Reihungen, verkettete Listen)</li> </ul> </li> <li>• Rekursion und Wiederholung (Rekursive Algorithmen, Abarbeitung von Datenstrukturen)</li> <li>• Suchen in <ul style="list-style-type: none"> <li>○ Texten</li> <li>○ Sammlungen</li> <li>○ Reihungen</li> <li>○ verketteten Listen</li> <li>○ Hashtabellen</li> </ul> </li> <li>• Sortierverfahren (quadratische, unterquadratische, rekursive, logarithmische)</li> <li>• Baumstrukturen (Suchen, Darstellen, Sortieren, Durchwandern, Operationen)</li> </ul>				
<b>4</b>	<p><b>Lehrformen</b> Vorlesung, Praktikum</p>				
<b>5</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal:</b> -</p> <p><b>Inhaltlich: Programmierkenntnisse in Java</b></p>				

<b>6</b>	<b>Prüfungsvoraussetzungen</b> Keine
<b>7</b>	<b>Prüfungsformen</b> Performanzprüfung
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Teilleistungen der Performanzprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dipl.-Inf. B.C. George

<b>Theoretische Informatik</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>2.4</b>	150 h	5 cps	2. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Übung 2 SWS	<b>Kontaktzeit</b> 4 SWS / 60h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden können <ul style="list-style-type: none"> <li>- zu gegebenen Automaten, Kellerautomaten, unterschiedlichen Arten von Grammatiken und Turingmaschinen die entsprechende Sprache angeben und umgekehrt</li> <li>- einen regulären Ausdruck in einen Automaten (deterministisch, nicht deterministisch) umwandeln und umgekehrt</li> <li>- die Chomsky-Hierarchie aufstellen und den Klassen Sprachen zuordnen</li> <li>- die Probleme der Berechenbarkeit und der Entscheidbarkeit und das Halteproblem erklären und diskutieren</li> <li>- das P-NP-Problem an Hand von Beispielen diskutieren</li> <li>- die Phasen des Compilers nennen und dabei die Anwendung von Automaten und Grammatiken erklären</li> </ul>				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>- Automaten (mit Ausgabe, deterministisch, nicht deterministisch)</li> <li>- reguläre Ausdrücke</li> <li>- Grammatiken, kontextfreie Sprachen</li> <li>- Kellerautomaten</li> <li>- kontextsensitive und Typ 0 - Sprachen, Turingmaschine</li> <li>- Chomsky-Hierarchie</li> <li>- Berechenbarkeit, Entscheidbarkeit, Church'sche These, Halteproblem</li> <li>- Komplexitätstheorie, P-NP-Problem</li> <li>- Compilerbau</li> </ul>				

<b>4</b>	<b>Lehrformen</b> Vorlesung, Übung
<b>5</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal: -</b> <b>Inhaltlich: -</b>
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Keine
<b>7</b>	<b>Prüfungsformen</b> Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Klausur
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dipl.-Inf. Ludger Franzen
<b>12</b>	<b>Sonstige Informationen</b> - Hopcroft, Ullman, Motwani, Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson, 2002  - Schöning, Theoretische Informatik - kurz gefasst, Spektrum, 2008  - Socher, Theoretische Grundlagen der Informatik, Hanser, 2008

<b>Useability und Datenvisualisierung</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>2.5</b>	150 h	5 cps	2. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Seminaristischer Unterricht 4 SWS	<b>Kontaktzeit</b> 4 SWS / 60h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Seminaristischer Unterricht 35	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Studierende lernen die Grundbegriffe von Useability Engineering kennen. Sie kennen die gängigen Useability Engineering Methoden und Praktiken. Darüber hinaus können sie ausgewählte Engineering-Methoden anwenden, indem sie einen besonderen Wert auf die Userverhalten und -anforderungen legen. Die Studierenden erlernen Techniken einer nutzerzentrierten Datenvisualisierung.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung: Definitionen, Grundbegriffe</li> <li>• Grundlagen von Benutzerschnittstellen</li> <li>• Gestaltungsrichtlinien</li> <li>• Vorgehensmodelle und Methoden des Useability Engineering</li> <li>• Technische Rahmenbedingungen</li> <li>• Gestaltungsprinzipien, Werkzeuge</li> <li>• Wahrnehmungspsychologische Aspekte der Datenvisualisierung</li> <li>• Techniken der Datenvisualisierung</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminaristischer Unterricht				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> - <b>Inhaltlich:</b> keine				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				
<b>7</b>	<b>Prüfungsformen</b>  Klausur				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Teilnahme am Seminaristischen Unterricht mit Testat Bestandene Klausur				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Kerstin Müller, Prof. Dr. Dominic Becking
<b>12</b>	<b>Sonstige Informationen</b>

<b>Software Engineering</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>3.1</b>	210 h	7 cps	3. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 150 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden erlernen ein systematisches und ingenieurmäßiges Vorgehen zur erfolgreiche Planung und Durchführung eines Softwareentwicklungsprojektes von der Anforderungsanalyse über Grobdesign, Feindesign, Implementierung bis hin zur Qualitätssicherung. Sie erlernen Standards und Tooleinsatz, und den gezielten Einsatz von UML. Die Programmierkenntnisse in Java werden erweitert u.a. um JUnit-Tests, Pattern, Frameworks und komponentenbasiertes Design.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung in Softwareengineering</li> <li>• Prozessmodellierung (Unternehmensprozesse, Aktivitätsdiagramme, Risikomanagement)</li> <li>• Vorgehensmodelle (Wasserfallmodell, Prototypische Entwicklung, Iterative Entwicklung, Iterativ-inkrementelle Entwicklung, Allgemeines V-Modell, Agile Entwicklung wie Scrum und Extreme Programming)</li> <li>• Anforderungsanalyse (Stakeholder, Ziele, Uses Cases, Ableitung funktionaler Anforderungen, nicht funktionale Anforderungen, Lasten- und Pflichtenheft)</li> <li>• Grobdesign (Systemarchitektur, Ableitung der grundlegenden Klassen, Methoden, Sequenzdiagramm, Überlegungen zur Oberflächenentwicklung)</li> <li>• Programmgenerierung (CASE-Tools, Übersetzung von Klassen und Assoziationen, Arten der Objektzugehörigkeit, Software-Architektur)</li> <li>• Feindesign (Details im Kleinen, Model View Controller, GoF-Pattern)</li> <li>• Implementierungen (Verteilte Systeme, XML, Bibliotheken, Komponenten, Frameworks, Persistente Datenhaltung, Model Driven Architecture)</li> <li>• Oberflächengestaltung (Useability, Ergonomie, Prüfung)</li> <li>• SW-Qualitätssicherung (Zusicherungen, Unit-Tests, Testverfahren, Metriken)</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Objektorientierte Programmierung, Programmiermethoden				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				

<b>7</b>	<b>Prüfungsformen</b> Performanzprüfung oder Praktikum mit Testat
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Teilleistungen der Performanzprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 7/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Jörg Brunsmann
<b>12</b>	<b>Sonstige Informationen</b> Literaturangaben:  Stephan Kleuker: „Grundkurs Software Engineering mit UML: Der pragmatische Weg zu erfolgreichen Softwareprojekten“, Vieweg&Teubner, Wiesbaden 2010  Bernd Oesterreich: "Analyse und Design mit UML" Oldenbourg, München 2004

<b>Verteilte Systeme und Kommunikationsnetze</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>3.2</b>	150 h	5 cps	3. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Kommunikation über Netze und das Internet bilden einen grundlegenden Bestandteil der modernen Berufswelt. Darüber hinaus sind Ethernet-Technologien und TCP/IP-basierte Kommunikation ein elementarer Bestandteil der meisten verteilten informationstechnischen Systeme.  Die Studierenden erlernen grundlegende Kenntnisse digitaler Kommunikationsnetze mit dem Schwerpunkt Rechnerkommunikation.  Sie lernen die Architektur und wichtige Methoden und Werkzeuge für verteilte Systeme kennen. Sie entwickeln die Fähigkeit, eigenständige verteilte Systeme und Netzwerk-Anwendungen zu entwickeln.  Sie lernen das Bewerten der Stärken und Schwächen unterschiedlicher Ansätze für verteilte Anwendungen.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Grundlagen der Kommunikationsnetze <ul style="list-style-type: none"> <li>- Kommunikationsmodelle/-protokolle und Standards</li> <li>- Vermittlungsprinzipien</li> </ul> </li> <li>• Technologien für lokale Netze (LAN) <ul style="list-style-type: none"> <li>- Übertragungsmedien</li> <li>- Stochastische und deterministische Medienzugriffsverfahren</li> <li>- Ethernet-Technologien und Protokolle</li> </ul> </li> <li>• Protokolle der TCP/IP-Protokollfamilie (IP, ICMP, UDP, TCP, Protokolle der Anwendungsschicht)</li> <li>• Übersicht über Technologien für Weitverkehrsnetze</li> <li>• Grundlagen IP-basierter Routing-Protokolle</li> <li>• Router und Routerkonfiguration</li> <li>• Routing-Protokolle</li> <li>• Objektorientierte, verteilte Systeme (Java RMI etc.)</li> <li>• Zeiten und logische Uhren</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>				

	<p><b>Formal: -</b></p> <p><b>Inhaltlich:</b> Objektorientierte Programmierung, Datenstrukturen und Algorithmen</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Keine.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Performanzprüfung</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Bestandene Teilleistungen der Performanzprüfung</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>-</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>5/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Martin Hoffmann</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <p>Literatur wird in der Vorlesung bekanntgegeben</p>

<b>Datenbanken</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>3.3</b>	150 h	5 cps	4. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Datenbanken sind eine Basistechnologie der Informatik. Ziel der Veranstaltung ist ein sicheres Verständnis der grundlegenden Konzepte und Sprachen von Datenbanksystemen zu vermitteln.</p> <p>Die Studierenden wenden ein einheitliches konsistentes Begriffsgebäude bezüglich der Datenbankthematik an. Sie können die theoretischen Grundlagen von Datenbanksystemen an Beispielen erläutern, insbesondere funktionale Abhängigkeiten, relationale Algebra und Normalisierung, sie wissen um die Aufgaben und den Sinn der grundlegenden Architektur von DBMS. Sie können die Dokumentation mehrerer wichtiger DBMS gezielt problemlösend durchsuchen und kennen die grundlegenden Funktionen der Clients mehrerer DBMS und nutzen Sie zur Kommunikation und Programmierung von Datenbanken. Sie modellieren komplexe Sachverhalte sicher in auch umfangreiche Datenmodelle und implementieren sie in verschiedenen DBMS. Sie fällen dabei begründete Entscheidungen für die Anwendungen von Constraints, Domänen und Datentypen. Sie wenden SQL sicher zur Lösung komplexer Informationsbedarfe an und erstellen umfangreiche nichttriviale Abfragen. Dabei verwenden sie sowohl den aktuellen SQL Standard (zur Zeit SQL:2011) als auch die Dialekte mehrerer wichtiger DBMS. Sie verstehen den Transaktionsbegriff, beschreiben Probleme/Phänomene der Mehrbenutzersynchronisation und Nebenläufigkeit in Read/Write Notation, und entscheiden, wie man sie durch Isolation von Transaktionen verhindert – sowohl durch Standard-Isolationslevel, als auch durch spezifische Implementierungen in mehreren DBMS. Sie greifen über Datenbankschnittstellen aus eigenen Programmen auf Datenbanken zu und verarbeiten Datensätze in Programmen und Datenbanken. Sie programmieren Persistent Stored Modules in einem der besprochenen DBMS.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>• Aufgaben und Architektur von Datenbanksystemen</li> <li>• Clients und Schnittstellen zu Datenbanksystemen</li> <li>• Grundlagen des relationalen Modells</li> <li>• E/R Modellierung, logische und physische Datenmodelle, SQL Datentypen, Implementierung in wichtigen DBMS</li> <li>• Constraints, Assertions, Integrität, Domänen, Datentypen</li> <li>• SQL:2011, insbesondere SQL-schema statements, SQL-data statements, SQL-</li> </ul>				

	<p>data change statements, SQL-transaction statements und SQL-connection statements</p> <ul style="list-style-type: none"> <li>• Transaktionskonzepte, Nebenläufigkeit, Isolationslevel</li> <li>• Datenbankschnittstellen (JDBC, ODBC)</li> <li>• Grundlagen von Persistent Stored Modules, Programmierung von PSM, Trigger</li> </ul>
<b>4</b>	<p><b>Lehrformen</b> Vorlesung, Praktikum</p>
<b>5</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal: -</b></p> <p><b>Inhaltlich: -</b></p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Keine</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Performanzprüfung</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Bestandene Teilleistungen der Performanzprüfung</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>keine</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>5/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Dominic Becking</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <ul style="list-style-type: none"> <li>• Kleuker, S., Grundkurs Datenbankentwicklung, Vieweg Teubner, 2011</li> <li>• Kemper, A, Eickler, A, Datenbanksysteme - Eine Einführung, Oldenbourg, 2011</li> <li>• Piepmeyer, L., Grundkurs Datenbanksysteme, Hanser, 2011</li> <li>• Saake, S., Sattler, K.-U., Heuer, A., Datenbanken - Konzepte und Sprachen, mitp, 2010</li> </ul>

<b>Systemprogrammierung</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>3.4</b>	240 h	8 cps	3. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 3 SWS Praktikum 3 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 150 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Die Studierenden erwerben umfassende Kompetenzen zur Entwicklung systemnaher Software in den aktuellen Systemprogrammiersprachen C und C++ unter Berücksichtigung gebräuchlicher Standards (z.B. ANSI-C und C++11). Sie beherrschen aktuelle Tools in diesem Umfeld, beispielsweise die Gnu-Compiler (gcc, g++) samt Debugger (gdb) und Make, und kennen verschiedene aktuelle Standardbibliotheken. Die Programmentwicklung unter Nutzung wesentlicher Teile der UNIX/Linux-Programmierschnittstelle (API) wird beherrscht.</p> <p>Die Studierenden können dieses Wissen selbständig auf komplexere Aufgabenstellungen anwenden (Praktikumsaufgaben).</p>				
<b>3</b>	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>- Grundlagen C/C++ <ul style="list-style-type: none"> <li>■ Typen, Ausdrücke, Operatoren, Kontrollfluss</li> <li>■ Strukturen und Enumerationen, typedef</li> <li>■ Bitoperationen</li> <li>■ Funktionen, Deklaration vs. Definition, Prototypen, Call-by-Value</li> <li>■ Sichtbarkeiten und Scopes, globale vs. lokale Variablen, static vs. extern, dateiübergreifend</li> <li>■ Speicherklassen</li> <li>■ Referenzen in C++</li> <li>■ Überladen von Operatoren und Funktionen in C++</li> <li>■ Nutzung eines Debuggers, z.B. gdb</li> <li>■ Unit-Test in C++, z.B. mit cppunit</li> </ul> </li> <li>- Objektorientierte Programmierung in C++ <ul style="list-style-type: none"> <li>■ Klassen, Konstruktoren, Destruktoren, Copy-Konstruktor, Zuweisungsoperator</li> <li>■ Friends</li> <li>■ Trennung Interface und Implementierung</li> <li>■ Vererbung, Polymorphie, virtual, Slicing, abstrakte Klassen</li> </ul> </li> <li>- Modulare Programmierung</li> </ul>				

	<ul style="list-style-type: none"> <li>■ Aufteilung auf Header- und Implementierungsdateien</li> <li>■ One Definition Rule</li> <li>■ Präprozessor: Include, Makros, bedingte Übersetzung, Konstanten</li> <li>■ Statische und dynamische Bibliotheken, Linker</li> <li>■ Makefiles</li> </ul> <p>- Speicherverwaltung</p> <ul style="list-style-type: none"> <li>■ Speicherverwaltung unter Linux, virtueller Speicher, Stack vs. Heap</li> <li>■ Pointer und Adressen, Deklaration, Dereferenzierung, Zuweisung</li> <li>■ Dynamische Speicherverwaltung mit malloc/free und new/delete</li> <li>■ Probleme bei Speicherverwaltung: Memory Leaks, Stale Pointer, Double Delete</li> <li>■ Call-by-Reference in C durch Pointer</li> <li>■ Zusammenhang Pointer und Arrays, mehrdimensionale Arrays, CMD-Parameter</li> <li>■ Adress-Arithmetik</li> <li>■ C-Strings und Funktionen aus der C-Std-Lib (z.B. strcpy, strcat, strtok)</li> <li>■ Funktionspointer</li> <li>■ Lesen komplexer Deklarationen</li> </ul> <p>- Ein- und Ausgabe</p> <ul style="list-style-type: none"> <li>■ Systemfunktionen unter Linux, Datei-Abstraktion, Standard-E/A-Kanäle</li> <li>■ Umgang mit Dateien unter C, scanf/printf</li> <li>■ Streams in C++, Fehlerzustände, Manipulation</li> </ul> <p>- Fehlerbehandlung</p> <ul style="list-style-type: none"> <li>■ Signalisierung von Fehlern in der Linux-Systemschnittstelle (Rückgabewert, errno), abort/exit/atexit</li> <li>■ Umgang mit Exceptions (C++)</li> <li>■ Assertions</li> </ul> <p>- Metaprogrammierung mit Templates (Funktionen, Klassen)</p> <p>- Einführung in Standardbibliotheken (z.B. STL, Boost)</p> <p>- Prozess- und Thread-Manipulation (Erzeugen, Beenden, Zustände)</p> <p>- Interprozess-Kommunikation (Überblick, Signale, Sockets)</p> <p>- Einbettung/Anbindung anderer Sprachen (z.B. Python, Lua)</p> <p>- Nutzung von Bibliotheken, z.B. SQLite3, libXML2, libCurl</p> <p>- Sichere und defensive Programmierung</p> <p>- Wechselnde Inhalte der Praktika zu aktuellen Themen</p>
<p><b>4</b></p>	<p><b>Lehrformen</b> Vorlesung, Praktikum</p>
<p><b>5</b></p>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal:</b> -</p> <p><b>Inhaltlich:</b> Programmiermethoden, Algorithmen und Datenstrukturen</p>
<p><b>6</b></p>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Keine</p>

<b>7</b>	<b>Prüfungsformen</b> Performanzprüfung
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Teilleistung der Performanzprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 8/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Carsten Gips
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Breymann, U.: "Der C++ Programmierer", Hanser, 2011.</li> <li>• Klemens, B.: "21st Century C", O'Reilly, 2012.</li> <li>• Love, O.: "Linux System Programming", O'Reilly Media, 2007.</li> <li>• Kerrisk, M.: "The Linux Programming Interface", no starch press, 2011.</li> </ul>

<b>Technical English</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>3.5</b>	150 h	5 cps	3. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Seminaristischer Unterricht 4 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Seminaristischer Unterricht 35	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Teilnehmer sollen in der Lage sein sich mit technischem Englisch auszudrücken. Sie sollen diese erworbenen Sprachkenntnisse in einem internationalen Umfeld sowohl schriftlich als auch gesprochen anwenden können. Die Studierenden erlernen hierzu die spezifische Terminologie, welche im Bereich Informatik angewendet wird, als auch allgemeines technisches Englisch und werden in der Lage sein Prozesse zu beschreiben und zu erklären, technische Probleme zu diskutieren und IT-Lösungen zu vermitteln.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Computer Hardware: Ein- und Ausgabemedien, Drucker, Speichermedien</li> <li>• Wesentliche und kreative Software: Betriebssystem, Textverarbeitung, Tabellenkalkulation und Datenbanken, Desktop-Publishing, Multimedia</li> <li>• Programmierung : Programmiersprachen, Java, Webgestaltung</li> <li>• Zukunftstechnologien: Kommunikationssysteme, Netzwerke, Videospiele</li> <li>• Internet: E-Mail, das Web, Videokonferenzführung</li> <li>• Beschäftigungsmöglichkeiten im Bereich IKT</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminaristischer Unterricht				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> - <b>Inhaltlich:</b> -				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Keine				
<b>7</b>	<b>Prüfungsformen</b>  Klausur				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Teilnahme am Seminaristischen Unterricht mit Testat  Bestandene Klausur				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> Keine
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Cathrine Stones
<b>12</b>	<b>Sonstige Informationen</b>

<b>Softwareprojekt</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.1</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Praktikum 4 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>In enger Verzahnung mit dem Modul "Projektmanagement"</p> <p>Die Studierenden wenden die wesentlichen Grundlagen des Projektmanagements an. Sie planen ein größeres Softwareprojekt, führen es durch, leiten es und dokumentieren und präsentieren regelmäßig Projektfortschritt und –ergebnisse.</p> <p>Sie entscheiden sich begründet in dem gegebenen Projektzusammenhang für ein Vorgehensmodell. Sie wenden Vorgehensweise, Organisationsformen und Methoden eines anerkannten formalen Projektmanagementsystems auf ihr Projekt an.</p> <p>Sie realisieren ein Projekt in einer größeren Projektgruppe (ca. 8 Personen) mit Rollenverteilung.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>• Einrichtung und Durchführung eines Projektes</li> <li>• Erstellung eines Pflichtenhefts auf Basis der Vorgaben eines fiktiven „Kunden“, Durchführung einer Aufwandsschätzung auf Basis dieses Pflichtenhefts</li> <li>• Aufstellen eines Projektplanes und Verfahren zur Projektverfolgung und des Risikomanagements</li> <li>• Implementierung im Team (Version-Management, Build-Verfahren, Abstimmungsprozesse, Schnittstellen)</li> <li>• Präsentation von Ergebnissen und Zwischenergebnissen</li> <li>• Einsatz aktueller Technologien zur Implementierung der Anwendung</li> <li>• Abschlusspräsentation</li> </ul>				
<b>4</b>	<p><b>Lehrformen</b></p> <p>Praktikum</p>				
<b>5</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal:</b> -</p> <p><b>Inhaltlich:</b> Objektorientierte Programmierung</p>				
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>				
<b>7</b>	<p><b>Prüfungsformen</b></p>				

	Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> Keine
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, Kreienkamp, König, Müller, Thiel)
<b>12</b>	<b>Sonstige Informationen</b>

<b>Betriebssysteme</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.2</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Betriebssysteme verwalten die Ressourcen eines Rechnersystems und bestimmen dadurch wesentlich dessen Leistungsfähigkeit und Nutzbarkeit. Gute Kenntnisse wesentlicher Betriebssystem-Konzepte sind daher für das Verständnis moderner IT Systeme unerlässlich. Studierende besitzen nach Abschluss der Lehrveranstaltung die folgenden Fähigkeiten:  - Systembezogene Abstraktionen kennen, verstehen und wiedererkennen, wie sie von Betriebssystemen verwendet und bereitgestellt werden - Wichtige Verfahren und Algorithmen aus dem Bereich Betriebssysteme sicher anwenden - Methoden kennen und verstehen, die bei der Konstruktion von Betriebssystemen Verwendung finden - beides an Beispielsystemen detailliert erläutern können (primär UNIX/Linux, z.T. Windows und andere aktuelle Betriebssysteme)				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Systemarchitektur moderner Rechnersysteme</li> <li>• Aufgaben, Arten und Strukturen von Betriebssystemen</li> <li>• Kurzvorstellung aktuelle Rechnerarchitekturen (Pipeline, Cache, Branch Prediction)</li> <li>• Nebenläufigkeit, Prozessverwaltung, Scheduling</li> <li>• Synchronisation und Kommunikation</li> <li>• Speicherverwaltung: Virtueller Speicher, Segmentierung, Paging, Seitenersetzungsalgorithmen</li> <li>• Ein-/Ausgabe: zeichen- und blockorientierte Geräte</li> <li>• Dateisysteme: Beispiele, Aufbau, Fehlertoleranz</li> <li>• Treiber-Modelle und -Programmierung</li> <li>• Sicherheit: Authentifikation, Schutzmechanismen, Autorisierung, vertrauenswürdige Systeme</li> <li>• Wechselnde Inhalte der Praktika zu aktuellen Themen (z.B. Bash-Skripting, Entwicklung einer Shell, Entwicklung eines Linux Kernels)</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal: -</b>				

	<p><b>Inhaltlich:</b> Systemprogrammierung</p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Performanzprüfung</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Bestandene Teilleistung der Performanzprüfung</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>-</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>5/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Martin Hoffmann</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <ul style="list-style-type: none"> <li>• Tanenbaum: "Moderne Betriebssysteme", Pearson Studium, 2009</li> <li>• Stallings: "Operating Systems: Internals and Design Principles", Prentice Hall, International, 2011</li> </ul>

<b>Embedded Systems</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.3</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden erlernen die grundlegenden Kenntnisse zur Umsetzung von eingebetteten Systemen (embedded systems). Sie erhalten ein Verständnis der Besonderheiten des Entwurfs eingebetteter Systeme und der zur Implementierung eingebetteter Systeme benötigten Kompetenzen. Insbesondere werden den Studierenden als "Handwerkszeug" die entsprechenden Spezifikations- und Programmieretechniken, Modelle der Ablaufplanung und Software- und Systemarchitekturen für eingebettete Systeme mitgegeben. Den Studierenden werden anhand von exemplarischen Anwendungsszenarien die typischen Entwurfsschritte vermittelt, um Software für eingebettete Systeme zu entwickeln (Praktikum).				
<b>3</b>	<b>Inhalte</b>  <ul style="list-style-type: none"> <li>• Überblick und Anwendungsgebiete</li> <li>• Besonderheiten und Anforderungen</li> <li>• Zusammenspiel Software und Hardware</li> <li>• Entwurfsschritte</li> <li>• Spezifikation und Modellierung</li> <li>• Realisierung und Implementierung</li> <li>• Validierung und Evaluation</li> </ul> Praktische Umsetzung eingebetteter Systeme an Anwendungsbeispielen.				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Kenntnisse aus der Technischen Informatik und C++-Programmierung  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  Performanzprüfung				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Bestandene Teilleistungen der Performanzprüfung				
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b>				

	-
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Matthias König
<b>12</b>	<b>Sonstige Informationen</b> Literatur: P. Marwedel, Eingebettete Systeme, Springer 2008

<b>Computergrafik</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.4</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden erlernen die fundamentalen Methoden und Algorithmen in der Computergrafik. Grundlegende Begriffe werden eingeführt und sowohl Verfahren aus der 2D wie auch 3D Computergrafik erläutert. Ebenso werden die mathematischen Grundlagen für die 2D und 3D Computergrafik erworben und in Anwendungsbeispielen eingeübt. Die Studierenden kennen gängige Tools und Anwender-Werkzeuge aus der Computergrafik und sind mit den zugehörigen Technologien vertraut. Sie sind in der Lage mit einem aktuellen Tool eine virtuelle Szene zu modellieren und zu rendern.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Historie, Überblick, Beispiele: Anwender-Werkzeuge, Tools, Anwendungsbeispiele.</li> <li>• Begriffe und Grundlagen: Grafische Eingabegeräte, Bildschirmtechnologien, 3D-Sichtsysteme, Rastergrafik.</li> <li>• Objekt- und Sichttransformationen: Koordinatensysteme, Transformationen, Projektionen, Clipping.</li> <li>• Repräsentation und Modellierung von Objekten: Polygonale Repräsentation, Raumteilungsverfahren, Szenenbeschreibung.</li> <li>• Rendering und Visibilität: Farbmodelle, Visibilitätsverfahren, Beleuchtung und Schattierung, lokale Beleuchtungsmodelle, interpolative Schattierungstechniken, globale Beleuchtungsmodelle, Rendering-Pipelines.</li> <li>• Tools zur Modellierung/Rendering: Modellierung und Rendern einer kleinen Szene mit Hilfe z.B. von Autodesk Maya, Blender.</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>				

	<p><b>Formal: -</b></p> <p><b>Inhaltlich:</b> Mathematik 1</p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Performanzprüfung</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Bestandene Teilleistungen der Performanzprüfung</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>-</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>5/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Kerstin Müller</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <p>Literaturangaben:</p> <ul style="list-style-type: none"> <li>• Bender M., Brill, M.: Computergrafik, 2. Auflage, Hanser Verlag, 2005. <a href="http://www.vislab.de">http://www.vislab.de</a></li> <li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li> <li>• Foley J., van Dam A., Feiner S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li> </ul>

<b>Webbasierte Anwendungen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.5</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Webbasierte Systeme sind eine der verbreitetsten Ausprägungen verteilter Informationssysteme und sind heute weder aus dem privaten noch aus dem beruflichen Bereich wegzudenken. Solche Systeme professionell planen, realisieren und beurteilen zu können, stellt daher eine wichtige Qualifikation dar.  <ul style="list-style-type: none"> <li>• Kennen typischer Anwendungsgebiete Web-basierter Anwendungen</li> <li>• Fähigkeit zum problemadäquaten Entwurf und zur Realisierung Web-basierter Anwendungen, einschl. Beurteilung und Auswahl geeigneter Basistechnologien</li> </ul>				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung, Klassifikation von Web-Anwendungen, Architekturen</li> <li>• Grundlagen (HTTP, Sessionmanagement, Standardisierung, W3C)</li> <li>• Sicherheit von Webanwendungen</li> <li>• Webseiten gestalten (XHTML, Cascading Stylesheets, HTML5)</li> <li>• Clientseitige Technologien: JavaScript, Ajax, DOM</li> <li>• Serverseitige Technologien: Servlets, JSP's</li> <li>• Mehrschichtenarchitekturen, Frameworks zu deren Umsetzung: z.B. JSF, Struts</li> <li>• Applikationsserver (Aufgaben, Dienste, Beispiele): z.B. Glassfish</li> <li>• Webservices z.B. REST</li> <li>• Datenaustauschformate z.B. JSON</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Programmieren in Java, Softwaretechnik, Skriptsprachen und XML  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>				

	Performanzprüfung
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Teilleistungen der Performanzprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Grit Behrens
<b>12</b>	<b>Sonstige Informationen</b> Literatur:  Bernd Müller: „JavaSever Faces 2.0: Ein Arbeitsbuch für die Praxis“, Hanser 2010  Martin Marinschek: „JavaServer Faces 2.0:Grundlagen und erweiterte Konzepte“, dpunkt 2009  Sven Haiges: „HMTL5-Apps für Iphone und Android“, Franzis 2011  Stefan Koch: „Javascript: Einführung, Programmierung und Referenz“, dpunkt 2011

<b>SW-Projektmanagement</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>4.6</b>	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Seminaristischer Unterricht 4 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Seminaristischer Unterricht 35	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden kennen die wesentlichen Grundlagen des Projektmanagements. Sie sind in der Lage, Projekte zu planen, durchzuführen, in Teilen zu leiten sowie Projektfortschritt und -ergebnisse zu dokumentieren und zu präsentieren. Sie können sich begründet in einem Projektzusammenhang für ein Vorgehensmodell entscheiden. Sie kennen Vorgehensweise, Organisationsformen und Methoden eines anerkannten formalen Projektmanagementsystems.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Projekte als Problemlösungsprozesse</li> <li>• Gründung, Organisation und Strukturierung von Projekten</li> <li>• Projektplanung</li> <li>• Projektsteuerung</li> <li>• Management von Software-Projekten</li> <li>• Werkzeuge im Projektmanagement</li> <li>• Kommunikation und Dokumentation als Querschnittsaufgabe</li> <li>• Qualitätssicherung</li> <li>• Projektmanagementsysteme</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminaristischer Unterricht				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> - <b>Inhaltlich:</b> -				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  Klausur, Projektarbeit				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Teilnahme am Seminaristischen Unterricht mit Testat Bestehen der Modulprüfung				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b>  -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b>  5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b>  Prof. Dr. Dominic Becking
<b>12</b>	<b>Sonstige Informationen</b>

<b>Grundlagen der BWL</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.1</b>	150 h	5 cps	5. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Übung 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Übung 30	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden erwerben Wissen um betriebswirtschaftliche Zusammenhänge im beruflichen Umfeld eines Informatikers. Sie kennen wesentliche betriebswirtschaftliche Verfahren und Grundbegriffe, haben einen Überblick zu Rechtsformen von Unternehmen, zur Investition und Finanzierung und zur Produktionsplanung und -steuerung. Sie haben einen Überblick zu Marketingstrategien.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Grundbegriffe der BWL</li> <li>• strategische Ausrichtung des Unternehmens (Portfolio-Analyse, Wettbewerb),</li> <li>• Investition und Finanzierung,</li> <li>• Betriebsorganisation</li> <li>• Controlling</li> <li>• Marketing</li> <li>• Personalwesen</li> <li>• Personalführung unter Berücksichtigung von Führungsstrukturen</li> <li>• Einsatz moderner Softwaretools</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Übung				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> -				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  Hausarbeit, mündliche Prüfung, Klausur				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Bestandene Modulprüfung				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Michael Mohe, Lehrbeauftragte
<b>12</b>	<b>Sonstige Informationen</b>

<b>Wahlfach aus Liste 1 "Methoden der Informatik"</b>					
<b>Komponentenbasierte SW-Entwicklung</b>					
<b>Kenn-Nr.</b>	<b>Workload</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.2</b>	150 h	5 cps	5. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b>	<b>Kontaktzeit</b>	<b>Selbststudium</b>	<b>geplante Gruppengröße</b>	
	Vorlesung 2 SWS Praktikum 2 SWS	4 SWS/60 h	90 h	Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>				
	<p>Studierende sollen komponentenbasierende Software-Entwicklung nutzen lernen. Der Vergleich zur objektorientierten Software-Entwicklung soll insbes. in Bezug auf die losere Kopplung und die verteilte Nutzbarkeit erkannt werden. Marktübliche Frameworks und deren Möglichkeiten sollen kennen gelernt werden.</p> <p>Die Studierenden verfügen über Kenntnisse über Frameworks der komponentenbasierten Software-Entwicklung.</p> <p>Die Studierenden sind in der Lage die Möglichkeiten marktüblicher Frameworks zu bewerten und diese entsprechend der Anforderungen einzusetzen.</p> <p>Die Studierenden können komponentenbasierte Frameworks zur Implementation von Anwendungen einsetzen.</p>				

<b>3</b>	<p><b>Inhalte</b></p> <ol style="list-style-type: none"> <li>1. Einführung und Geschichte COM/DCOM, DLLs</li> <li>2. Vergleich Objektorientierte/komponentenbasierende Software-Entwicklung</li> <li>3. Applikationsserver</li> <li>4. Grundlegende Dienste von Komponentenstandards             <ol style="list-style-type: none"> <li>4.1 Verteilung, Erzeugung, Aktivierung Deaktivierung von Komponenten</li> <li>4.2 Naming/Verzeichnisdienste</li> <li>4.3 Messaging</li> <li>4.4 Transaktionen</li> <li>4.5 Sicherheit/Redundanz</li> <li>4.6 Persistenz, Datenbankschnittstellen</li> <li>4.7 Weitere Dienste</li> </ol> </li> <li>5 Übersicht aktuelle Komponentenstandards             <ol style="list-style-type: none"> <li>5.1 CCM (Corba 3.0)</li> <li>5.2 COM+</li> <li>5.3 EJB (J2EE)</li> </ol> </li> <li>6. Nutzung aktueller Komponentenstandards anhand eines praktischen Beispiels</li> <li>7. Ausblick</li> </ol>
<b>4</b>	<p><b>Lehrformen</b> Vorlesung, Praktikum</p>
<b>5</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal:</b> -</p> <p><b>Inhaltlich:</b> Objektorientierte Programmierung, Datenbanken, Software Engineering, Webbasierte Anwendungen, Verteilte Systeme</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Mündliche Prüfung, Klausur</p>

<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Praktikum mit Testat oder Performanzprüfung  Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b>  -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b>  5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b>  Prof. Dr. Jörg Brunsmann
<b>12</b>	<b>Sonstige Informationen</b>  Dustdar, S.; Gall, H.; Hauswirth, M.: Software-Architekturen für verteilte Systeme, Springer-verlag, Berlin Heidelberg 2003  Andresen, A.: Komponentenbasierte Software-Entwicklung (Kapitel 6-10), Carl Hanser Verlag München Wien 2004  Emmerich.W.: Konstruktion von verteilten Objekten, dpunkt-Verlag Heidelberg, 2003

<b>Wahlfach aus Liste 1 „Methoden der Informatik“ Spezielle Methoden der Programmierung</b>					
<b>Kenn-Nr.</b> <b>5.3</b>	<b>Work-load</b> 150 h	<b>Credits</b> 5 cps	<b>Studien-semester</b> 5. /6. Sem.	<b>Häufigkeit des Angebots</b> Bei Nachfrage alle 2 Semester	<b>Dauer</b> 1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Teilnehmer erhalten einen Überblick über verschiedene Programmierparadigmen und Sprachkonzepte sowie ihre Anwendungsgebiete. Sie lernen ausgewählte Konzepte an entsprechenden Programmiersprachen genauer kennen und verstehen Aufbau und Prinzipien von Programmiersprachen. Die Teilnehmer werden befähigt, selbständig neue Sprachkonstrukte zu erlernen und einzuordnen. Nach erfolgreichem Abschluss des Moduls sind sie in der Lage, problem- und lösungsorientiert ein geeignetes Paradigma und eine entsprechende Programmiersprache auszuwählen und einzusetzen.				
<b>3</b>	<b>Inhalte</b>  Auswahl an Themen zur Wissensvermittlung:  - Überblick über Programmierparadigmen und -konzepte - Vertiefung ausgewählter Konzepte <ul style="list-style-type: none"> <li>■ funktionale Programmierung (beispielsweise Closure oder Haskell)</li> <li>■ hybride objektorientierte und funktionale Programmierung (beispielweise Scala oder Groovy)</li> <li>■ nebenläufige/parallele Programmierung (beispielsweise Erlang)</li> <li>■ logische Programmierung (beispielsweise Prolog)</li> <li>■ Domainspezifische Sprachen (intern, extern, Beispiele)</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Theoretische Informatik, Programmiermethoden, Software Engineering				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				

<b>7</b>	<b>Prüfungsformen</b> mündliche Prüfung, Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Praktikum mit Testat Bestandene Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Carsten Gips
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Tate, B.A.: "Seven Languages in Seven Weeks", Pragmatic Bookshelf Inc., 2010</li> <li>• Scott, M.L.: "Programming Language Pragmatics", Morgan Kaufmann, 2009</li> <li>• Lipovaca, M.: "Learn You a Haskell", No Starch Press, 2011</li> <li>• Horstmann, C.: "Scala for the Impatient", Addison Wesley, 2012</li> <li>• Odersky, M.: "Programming in Scala", Artima, 2011</li> <li>• Subramaniam, V.: "Programming Groovy 2", O'Reilly, 2013</li> <li>• Halloway, S.: "Programming Clojure", Pragmatic Programmers, 2009</li> </ul>

<b>Wahlfach aus Liste 1 "Methoden der Informatik"</b>					
<b>Mustererkennung &amp; Bildverarbeitung</b>					
<b>Kenn-Nr.</b>	<b>Workload</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.4</b>	150 h	5 cps	5. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b>	<b>Kontaktzeit</b>	<b>Selbststudium</b>	<b>geplante Gruppengröße</b>	
	Vorlesung 2 SWS Praktikum 2 SWS	4 SWS/60 h	90 h	Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>				
	<p>Die Studierenden kennen die wesentlichen Methoden der Mustererkennung und Bildverarbeitung. Insbesondere verstehen die Studierenden das Zusammenspiel von Bildverarbeitung und Klassifikation zur Extraktion von Informationen aus digitalen Bildern, um beispielsweise Problemstellungen in den Bereichen Computer Vision und Medizinische Bildverarbeitung lösen zu können.</p> <p>Die Studierenden erlernen im Praktikum sowohl das Erstellen von eigenen Programmen zur Bildverarbeitung als auch das Arbeiten mit einer Bildverarbeitungssoftware.</p>				
<b>3</b>	<b>Inhalte</b>				
	<ul style="list-style-type: none"> <li>Überblick und Anwendungsgebiete</li> <li>Bildaufnahme</li> <li>Bildrepräsentation und -kompression</li> <li>Bildverbesserung und Filter</li> <li>Segmentierung</li> <li>Registrierung</li> <li>Merkmalsextraktion und -reduktion</li> <li>Klassifikationsverfahren</li> <li>Clustering</li> </ul>				
<b>4</b>	<b>Lehrformen</b>				
	Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>				
	<p><b>Formal:</b> -</p> <p><b>Inhaltlich:</b> Kenntnisse aus Analysis, Linearer Algebra und C++-Programmierung</p>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>				
	Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				

<b>7</b>	<b>Prüfungsformen</b> Mündliche Prüfung, Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Praktikum mit Testat Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Matthias König
<b>12</b>	<b>Sonstige Informationen</b> Literatur: K. D. Tönnies, Grundlagen der Bildverarbeitung, Pearson Studium, 2005 R. C. Gonzales, R. E. Woods, Digital Image Processing, 3rd. Ed., Prentice Hall International, 2008 G. Bradski, A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, 2008

<b>Wahlfach aus Liste 1 „Methoden der Informatik“ Ausgewählte Konzepte von Datenbanksystemen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.5</b>	150 h	5 cps	6. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Datenbanksysteme sind hochkomplexe Softwaresysteme, zu deren sicherer Anwendung und Nutzung ein vertieftes Wissen der Implementierung und Administration gehört. Die Studierenden erwerben theoretische Kenntnisse in fortgeschrittenen Datenbanktechniken. Sie fällen begründete Entscheidungen für die Anwendung dieser Techniken. Sie können Datenbanken installieren und administrieren und dabei die Anforderungen der Anwendungssoftware analysieren und umsetzen. Sie können die Performanz von Datenbanken analysieren und durch geeignete technische Maßnahmen steigern. Sie können fortgeschrittene SQL formulieren und sinnvoll anwenden. Sie können SQL zur Performanzsteigerung analysieren und optimieren (SQL-Tuning).				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Architektur von Datenbanksystemen</li> <li>• Verwaltung des Hintergrundspeichers</li> <li>• Pufferverwaltung</li> <li>• Dateiorganisation und Zugriffsstrukturen</li> <li>• Spezielle Indexstrukturen</li> <li>• Basisalgorithmen für Datenbankoperationen</li> <li>• Fortgeschrittene SQL</li> <li>• Optimierung von Anfragen</li> <li>• Transaktionsmodelle</li> <li>• Transaktionsverwaltung</li> <li>• Wiederherstellung und Datensicherung</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Inhalte des Moduls „Datenbanken“				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				

<b>7</b>	<b>Prüfungsformen</b> Mündliche Prüfung, Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Praktikum mit Testat Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Dominic Becking
<b>12</b>	<b>Sonstige Informationen</b> Literatur: Saake, G., Sattler, K.-U., Datenbanken: Implementierungstechniken, Heidelberg 2011 Härder, Th., Rahm, E., Datenbanksysteme: Konzepte und Techniken der Implementierung, Berlin 2001 Aktuelle Literatur zu neueren Datenbanktechnologien

<b>Wahlfach aus Liste 1 „Methoden der Informatik“ Künstliche Intelligenz</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.6</b>	150 h	5 cps	5. /6. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Die Veranstaltung führt in die Prinzipien und Methoden und Verfahren der modernen Künstlichen Intelligenz und deren Anwendung in intelligenten Systemen ein. Insbesondere werden die grundlegenden Methoden im Bereich der Künstlichen Intelligenz wie Konzepte des Problemlösens, der Wissensrepräsentation, der logischen Programmierung, des maschinellen Lernens und von neuronalen Netzen sowie des Planens vermittelt.</p> <p>Mit der erfolgreichen Teilnahme an der Vorlesung und den Übungen werden Kompetenzen zur Auswahl und Anwendung von Methoden der Künstlichen Intelligenz für eine konkrete Problemstellung erworben. Die Teilnehmer sind in der Lage, die erlernten Methoden auf andere Bereiche und Problemstellungen zu übertragen und anzuwenden.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <p>Auswahl an Themen zur Wissensvermittlung:</p> <ul style="list-style-type: none"> <li>- Intelligente Agenten</li> <li>- Problemlösen: <ul style="list-style-type: none"> <li>■ Informierte und uninformierte Suche (Tiefensuche, Breitensuche, Branch-and-Bound, Best First, A*)</li> <li>■ Lokale Suche (Gradientensuche, Simulated Annealing)</li> <li>■ Genetische und Evolutionäre Algorithmen</li> <li>■ Constraint Satisfaction Problems, Backtracking-Suche mit Heuristiken, AC3 (Kantenkonsistenz)</li> <li>■ Spiele (Minimax Algorithmus, alpha-beta-Pruning)</li> </ul> </li> <li>- Wissensrepräsentation und Schliessen: <ul style="list-style-type: none"> <li>■ Aussagenlogik</li> <li>■ Prädikatenlogik, Syntax und Semantik, Modelle</li> <li>■ Unifikation, Normalform, Resolution</li> <li>■ Logische Programmierung (Prolog)</li> <li>■ Schliessen mit Unsicherheit, Bayes'sche Regel, Bayes'sche Netze</li> <li>■ Semantische Netze und Ontologien</li> <li>■ Planen</li> </ul> </li> </ul>				

	<ul style="list-style-type: none"> <li>- Maschinelles Lernen: <ul style="list-style-type: none"> <li>■ Entscheidungsbaumverfahren (CAL2, CAL3, ID3)</li> <li>■ Neuronale Netze: Perzeptron, MLP, Backpropagation</li> <li>■ Support Vector Machines</li> <li>■ Reinforcement Learning</li> <li>■ Generalisierung, Overfitting, Crossvalidierung, Boosting</li> <li>■ Unüberwachtes Lernen</li> </ul> </li> <li>- Unschärfe Textsuche, Ähnlichkeiten zw. Texten</li> <li>- Empfehlungssysteme</li> <li>- Autonome mobile Systeme</li> <li>- Wechselnde Inhalte der Praktika zu aktuellen Themen</li> </ul>
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum
<b>5</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> - <b>Inhaltlich:</b> Theoretische Informatik, Algorithmen und Datenstrukturen, Programmiermethoden
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO.
<b>7</b>	<b>Prüfungsformen</b> Mündliche Prüfung, Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Teilnahme am Praktikum mit Testat Bestandene Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Carsten Gips
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Russel, S., Norvig, P: "Artificial Intelligence. A Modern Approach", Prentice Hall, 2010</li> <li>• Bishop, C.: "Pattern Recognition and Machine Learning", Springer, 2007</li> <li>• Haykin, S.: "Neural Networks and Learning Machines: A Comprehensive Foundation", Prentice Hall, 2008</li> <li>• Ertel, W.: "Grundkurs Künstliche Intelligenz", Springer Vieweg, 2013</li> </ul>

<b>Wahlfach aus Liste 1 "Methoden der Informatik"</b>					
<b>Sicherheit und Zuverlässigkeit</b>					
<b>Kenn-Nr.</b>	<b>Workload</b>	<b>Credits</b>	<b>Studien- semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.7</b>	150 h	5 cps	5. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden verstehen die Zusammenhänge und Unterschiede der Sicherheitsbegriffe Dependability, Safety und Security. Sie sind mit typischen Schwachstellen und Bedrohungen vertraut und kennen geeignete Gegenmaßnahmen und -mechanismen zur Erhöhung der Zuverlässigkeit und Sicherheit, deren Einsatz sie anhand von Übungen erlernen. Die Studierenden können erste Einschätzungen der Sicherheit und Zuverlässigkeit von Systemen und Software vornehmen und Möglichkeiten und Grenzen von Sicherheitslösungen bewerten				
<b>3</b>	<b>Inhalte</b>  Grundlegende Begriffe und Probleme; Ziele der Verlässlichkeit und Sicherheit (Vertraulichkeit, Integrität, Verfügbarkeit, Safety, Wartbarkeit, ...); Schwachstellen-, Bedrohungs- und Risikoanalysen sowie Sicherheitsplan; Maßnahmen und Mechanismen zur Erhöhung der Zuverlässigkeit und Sicherheit von Software und Systemen (Kryptographie, Authentifizierung, Zugriffskontrolle, Protokolle, Firewalls, etc.)				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Mathematik für Informatiker I und II, Programmierkenntnisse, Technische Informatik, Betriebssysteme, Datenbanken				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				

<b>7</b>	<b>Prüfungsformen</b> Mündliche Prüfung, Klausur
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Teilnahme am Praktikum mit Testat Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Christoph Thiel
<b>12</b>	<b>Sonstige Informationen</b>  Literatur: N. Storey, Safety Critical Computer Systems, Addison Wesley, 1996 C. P. Pfleeger, Security in Computing, 4th ed., Prentice Hall, 2007 C. Eckert, IT-Sicherheit, Oldenbourg Wissenschaftsverlag, 2013

<b>Wahlfach aus Liste 1 „Methoden der Informatik“ Methoden der Computergrafik</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.8</b>	150 h	5 cps	5. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 2 SWS	<b>Kontaktzeit</b> 4 SWS/60 h	<b>Selbststudium</b> 90 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik erlangen die Studierenden vertiefende Kenntnisse in den Bereichen Algorithmen und Datenstrukturen der Computergrafik, geometrische Modellierung, Visualisierung und Human Computer Interaction. In praktischen Beispielen werden die behandelnden Verfahren eingeübt.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Algorithmen und Datenstrukturen: Verfahren aus der algorithmische Geometrie.</li> <li>• Einführung in die geometrische Modellierung: Bezier, B-Spline Kurven und Flächen.</li> <li>• Globale Beleuchtungsmethoden: Raytracing, Radiosity.</li> <li>• Virtual Reality Systeme.</li> <li>• Human Computer Interaction: Wahrnehmungspsychologie, Informationsverarbeitung.</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Computergrafik				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  mündliche Prüfung, Klausur				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>				

	Teilnahme am Praktikum mit Testat Bestandene Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Kerstin Müller
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"><li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li><li>• Foley J., van Dam A., Feiner S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li></ul>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Anwendungen der Computergrafik</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.9</b>	450 h	15 cps	5./6. Sem.	bei Nachfrage alle zwei Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik  bearbeiten die Studierenden aktuelle Problemstellungen aus der Computergrafik und sind in der Lage geeignete Methoden zu evaluieren und aktuelle Werkzeuge aus der Computergrafik auszuwählen. Die zu erstellenden Softwareapplikationen kommen aus den Anwendungsfeldern Mustererkennung & Bildverarbeitung, algorithmische Geometrie, geometrische Modellierung, Animation, physikalische Simulation, Virtual Reality, Informationsvisualisierung und Human Computer Interaction. Optional können auch interdisziplinäre Projekte bearbeitet werden.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Projektarbeiten im Team zu computergrafisch spezifischen Anwendungen aus aktuellen Forschungsgebieten und Forschungskooperationen mit Industriepartnern.</li> <li>• Visualisierungstechniken: Methoden zur optimierten Darstellung.</li> <li>• Effiziente Datenstrukturen der Computergrafik,</li> <li>• Geometrische Modellierung: Bezier und B-Spline Techniken.</li> <li>• Human Computer Interaction und effektive Informationsverarbeitung.</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Computergrafik				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>				

	Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Erfolgreicher Abschluss der Projektarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 15/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Kerstin Müller
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li> <li>• Foley J., van Dam A., Feiner S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li> </ul>

## Wahlfach aus Liste 2 „Anwendungen der Informatik“ Software Qualität

<b>Kenn-Nr.</b> <b>5.10</b>	<b>Work-load</b> 450 h	<b>Credits</b> 15 cps	<b>Studien-semester</b> 5./6. Sem.	<b>Häufigkeit des Angebots</b> bei Nachfrage alle zwei Semester	<b>Dauer</b> 1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Die Veranstaltung führt in Techniken und Methoden zur Softwarequalitätssicherung bei der Entwicklung und beim Betrieb von Softwaresystemen ein.</p> <p>Mit der erfolgreichen Teilnahme an der Veranstaltung wird ein Verständnis für Softwarequalität und die Bedeutung von systematischem Softwaretest erworben. Die Teilnehmer kennen den allgemeinen Testprozess sowie die Aufgaben des Testmanagements. Sie kennen verschiedene Teststufen und -arten und sind in der Lage, verschiedene statische und dynamische Prüftechniken und Werkzeuge bedarfsorientiert auszuwählen und einzusetzen. Die Teilnehmer sind in der Lage, unter Anwendung der erlernten Methoden Software von höherer Qualität zu entwickeln.</p> <p>Die Veranstaltung dient der Vermittlung grundlegender theoretischer Kenntnisse und Fähigkeiten, wobei die praktische Anwendung im Projekt erlernt und vertieft werden soll. Durch die teamorientierte Projektarbeit werden die Projektmanagement- und Selbst-Kompetenzen der Teilnehmer vertieft.</p> <p>Nach der erfolgreichen Teilnahme an diesem Modul ist optional eine Prüfung zum "Certified Tester – Foundation Level" nach ISTQB bei einem vom German Testing Board zertifizierten Prüfungsinstitut möglich.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <p>Auswahl an Themen zur Wissensvermittlung:</p> <ul style="list-style-type: none"> <li>- Qualitätsaspekte von Softwaresystemen</li> <li>- Grundlagen des Softwaretestens, Testprinzipien, fundamentaler Testprozess</li> </ul>				

	<ul style="list-style-type: none"> <li>- Testen im Software-Life-Cycle, Teststufen und -arten</li> <li>- Statische Testtechniken</li> <li>- Dynamische Testtechniken, Testdesign <ul style="list-style-type: none"> <li>■ Spezifikationsbasiert (Black-Box)</li> <li>■ Strukturbasiert (White-Box)</li> </ul> </li> <li>- Testmanagement</li> <li>- Werkzeuge</li> <li>- Testautomatisierung</li> <li>- Testfallgenerierung</li> <li>- Wechselnde Inhalte der Projekte zu aktuellen Themen</li> </ul>
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Programmiermethoden, Software Engineering, Softwareprojekt
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.
<b>7</b>	<b>Prüfungsformen</b>  Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Erfolgreicher Abschluss der Projektarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b>  -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b>  15/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b>  Prof. Dr. Carsten Gips
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Spillner, A., Linz, T.: "Basiswissen Softwaretest", dpunkt-Verlag, 2010</li> <li>• Kleuker, S.: "Qualitätssicherung durch Softwaretests", Springer Vieweg, 2013</li> <li>• Liggesmeyer, P.: "Software-Qualität", Spektrum Akademischer Verlag, 2009</li> <li>• Graham et.al.: "Foundations of Software Testing", Thomson Learning, 2008</li> <li>• Myers, G.J.: "The Art of Software Testing", John Wiley, 2011</li> </ul>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Webengineering</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.11</b>	450 h	15 cps	5. /6. Sem.	Bei Nachfrage jedes Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Verständnis von aktuellen Konzepten, Methoden, Techniken, Werkzeugen und Erfahrungen zur ingenieurmäßigen Entwicklung von Web-Anwendungen sowie ihre praktische Anwendung in der eigenen Projektarbeit in Web-Entwicklerteams.</p> <ul style="list-style-type: none"> <li>• Bewerten von potentiellen Risiken von Web-Anwendungen</li> <li>• Befähigung, zukünftige Entwicklungen im Bereich des Web-Engineering zu verfolgen und zu beurteilen</li> </ul> <p>Die erworbenen Fähigkeiten tragen damit in besonderem Maße zur Ausprägung von spezifischen Analyse-, Design- und Realisierungskompetenzen und zur Erweiterung von spezifischen technologischen Kompetenzen bei. Teamorientierte Projektarbeit erhöht ferner Projektmanagement- und Selbst-Kompetenzen.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <p>Einführung in Web-Engineering (Motivation, Definition, Grundprinzipien)</p> <ul style="list-style-type: none"> <li>• Produktentwicklung</li> <li>• Requirements Engineering für Web-Anwendungen</li> <li>• Modellierung von Web-Anwendungen</li> <li>• Architektur von Web-Anwendungen</li> <li>• Testen von Web-Anwendungen</li> <li>• Web-Projektmanagement</li> <li>• Qualitätsaspekte (Usability, Performanz, Sicherheit)</li> <li>• javabasierte Web-Frameworks z.B. JSF mit Primefaces, Richfaces und JPA</li> <li>• Webanwendungen für mobile Geräte mit z.B. PhoneGap od. Android</li> </ul>				
<b>4</b>	<b>Lehrformen</b>				

	Vorlesung, Praktikum
<b>5</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal: -</b> <b>Inhaltlich:</b> Webbasierte Anwendungen, Programmieren in Java, Softwaretechnik
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO.
<b>7</b>	<b>Prüfungsformen</b> Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Erfolgreicher Abschluss der Projektarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 15/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Grit Behrens
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Bernd Müller: "Java Server Faces", Hanser 2006</li> <li>• Kappel, Pröll, Reich, Teschitzegger: Web-Engineering, dpunkt 2004</li> <li>• Spiering Haiges: "HTML5-Apps" Franzis 2011</li> </ul>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Mobile Applikationen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.12</b>	450 h	15 cps	5./6. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Softwareentwicklung für mobile Geräte ist die Weiterentwicklung der traditionellen (Anwendungs-) Softwareentwicklung hin zu einer offenen, verteilten und dynamischen Technologie, die mobile, leistungsfähige Geräte mit traditionellen Strukturen des Internets und verteilten Software-Systemen verbindet.</p> <p>Ziel dieses Moduls ist es, dass die Studierenden</p> <ul style="list-style-type: none"> <li>• die Besonderheiten und Randbedingungen sowie Konzepte und Lösungsansätze der mobilen Applikationsentwicklung verstehen und auch quantitativ abschätzen können</li> <li>• darauf zugeschnittene Lösungen der Systemarchitektur und der Anwendungsentwicklung kennen lernen</li> <li>• und diese Kenntnisse zur Lösung von konkreten Aufgabenstellungen praktisch anwenden können.</li> </ul> <p>Die Studierenden erhalten einen Überblick über aktuelle Hardwareplattformen und mobile Betriebssysteme und lernen ein aktuelles System (beispielsweise Android) zu beherrschen. Nach dem Besuch der Veranstaltung sind sie in der Lage, unter Berücksichtigung von Usability-, Energie- und Sicherheitsaspekten sowohl native als auch Webapplikationen für dieses System zu erstellen, die internen Sensoren und Hardwarebausteine zu nutzen.</p> <p>Zur Vertiefung werden größere Projekte erarbeitet, beispielsweise Applikationen zur Steuerung mobiler Roboter oder mobile Anwendungen aus den Bereichen Computer Grafik oder Augmented Reality. Die teamorientierte Projektarbeit vertieft die Projektmanagement- und Selbst-Kompetenzen.</p>				
<b>3</b>	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>• Technologie mobiler Anwendungen, Hardware, Betriebssysteme</li> <li>• Anwendungen &amp; Anwendungsentwicklung <ul style="list-style-type: none"> <li>○ Aktuelle Software-Frameworks für mobile Anwendungen</li> <li>○ Ressourcenmanagement in mobilen Systemen</li> <li>○ Sicherheit für mobile Systeme und Anwendungen</li> <li>○ Energieaspekte</li> </ul> </li> <li>• Nutzung von Hardwarebausteinen mobiler Geräte</li> </ul>				

	<ul style="list-style-type: none"> <li>○ Einbindung vorhandener Sensoren und Schnittstellen</li> <li>○ Ansteuern externer Hardware, beispielsweise Arduino</li> <li>• Wechselnde Inhalte der Projekte zu aktuellen Themen</li> </ul>
<b>4</b>	<p><b>Lehrformen</b></p> <p>Vorlesung, Praktikum</p>
<b>5</b>	<p><b>Teilnahmevoraussetzungen</b></p> <p><b>Formal: -</b></p> <p><b>Inhaltlich:</b> Kenntnisse der Java- oder Objective-C-Programmierung, Software Engineering</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Projektarbeit</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Erfolgreicher Abschluss der Projektarbeit</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>-</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>15/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Dr. Matthias König, Prof. Dr. Martin Hoffmann</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <ul style="list-style-type: none"> <li>• J. Roth: "Mobile Computing - Grundlagen, Technik, Konzepte", dpunkt-Verlag, 2005</li> <li>• Mike Bach: "Mobile Anwendungen mit Android", Addison-Wesley, 2011</li> <li>• Tom Igoe: "Making Things Talk, 2nd Edition", O'Reilly, 2012</li> <li>• R. Siegwart, I. R. Nourbakhsh: "Introduction to Autonomous Mobile Robots", mitp, 2011</li> </ul>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Softwareengineering für Embedded Systems</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.13</b>	450 h	15 cps	5./6. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontakt-zeit</b> 6 SWS/90 h	<b>Selbst-studium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden wissen die Methoden des Softwareengineering für Embedded Systems anzuwenden unter Berücksichtigung der Randbedingungen dieser Systeme. Sie besitzen die Fähigkeit, die Entwicklung von Software für Embedded Systems von der Planung bis zur Realisierung durchzuführen.  Die Veranstaltung baut auf den Kenntnissen der Lehrveranstaltung „Embedded Systems“ auf.				
<b>3</b>	<b>Inhalte</b>  <ul style="list-style-type: none"> <li>• Anforderungen / Requirements</li> <li>• Software Architektur und Design Patterns</li> <li>• Modellbasierte Entwicklung</li> <li>• UML/SysML für Embedded Systems</li> <li>• Embedded Betriebssysteme</li> <li>• Distributed Embedded Systems</li> <li>• Besonderheiten der Programmierung von Embedded Systems</li> <li>• Qualitätssicherung</li> </ul> Als vertiefendes Projekt wird beispielsweise unter einem Betriebssystem für eingebettete Systeme (z.B. Embedded Linux, Android, RTOS) eine Anwendung entwickelt (z.B. für Computer Vision, Robotik, Mess- und Regelungstechnik).				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Kenntnisse eingebetteter Systeme, Software Engineering und C++-Programmierung				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>				

	Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Erfolgreicher Abschluss der Projektarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 15/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Matthias König
<b>12</b>	<b>Sonstige Informationen</b>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Datenbankanwendungen</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.14</b>	450 h	15 cps	5. Sem.	Bei Nachfrage alle 2 Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Datenbanken sind die Grundlagen jedes großen Softwaresystems. Das Zusammenwirken von Programmierung und Datenbank ist dabei von größter Wichtigkeit. Die Studierenden kennen die Anforderungen verschiedener Softwaresystemklassen an Datenbanken. Sie können spezielle Datenmodelle für unterschiedliche Systemklassen entwerfen und implementieren. Sie entwerfen und realisieren ein Softwaresystem bestehend aus Anwendungsprogramm und darauf abgestimmter Datenbank. Sie können Programmlogik in der Datenbank mit Persistent Stored Modules implementieren. Die Teilnehmer können verschiedene APIs zur Anbindung von Anwenderprogrammen an Datenbanken verwenden.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Softwaresysteme und Anforderungen an Datenbanken</li> <li>• Anforderungsanalyse für Datenbankanwendungen</li> <li>• Objektorientierte und ER-Modellierung</li> <li>• Persistent Stored Modules (SQL/PSM)</li> <li>• Trigger</li> <li>• Impedance Mismatch</li> <li>• Cursors</li> <li>• Objektrelationale Mappings – Object Language Bindings (SQL/OLB)</li> <li>• APIs und Frameworks für Datenbankzugriffe</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Inhalte des Moduls „Datenbanken“				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  Performanzprüfung				

<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestehen der Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Dominic Becking
<b>12</b>	<b>Sonstige Informationen</b>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Anwendungen der künstlichen Intelligenz</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>5.15</b>	450 h	15 cps	5./6. Sem.	Bei Nachfrage alle zwei Semester	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studenten sind in der Lage, geeignete Suchstrategien zur Lösungsermittlung auszuwählen und in Teams erfolgreich eine intelligente Applikation zu entwickeln. Mobile Geräte mit verschiedenen Sensoren wie Mobiltelefone und Mobile autonome Roboter werden erfolgreich in die Applikationen über Serviceschnittstellen integriert.				
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Grundlagen der Modernen KI</li> <li>• Ansatz des intelligenten Agenten</li> <li>• Verschiedene Suchstrategien theoretisch und beispielhaft</li> <li>• Implementierung ausgesuchter Suchstrategien in passenden Programmiersprachen</li> <li>• Aktuelle Forschungs- und Anwendungsprojekte in Teamarbeit</li> <li>• eventbasierte Serviceschnittstellen</li> <li>• Middleware</li> <li>• Elemente der Robotik</li> <li>• autonome Service Roboter</li> <li>• Sensoren (Video, Audio)</li> <li>• Elemente der Bildverarbeitung, Sprachverarbeitung und Mustererkennung</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>				

	<p><b>Formal: -</b></p> <p><b>Inhaltlich:</b> Programmiersprachen Java, C++ und Python, webbasierte Anwendungen, verteilte Systeme und Rechnerkommunikation</p>
<b>6</b>	<p><b>Prüfungsvoraussetzungen</b></p> <p>Es gilt § 17 „Fortschrittsregelung“ dieser BPO.</p>
<b>7</b>	<p><b>Prüfungsformen</b></p> <p>Projektarbeit</p>
<b>8</b>	<p><b>Voraussetzungen für die Vergabe von Kreditpunkten</b></p> <p>Erfolgreicher Abschluss der Projektarbeit</p>
<b>9</b>	<p><b>Verwendung des Moduls (in anderen Studiengängen)</b></p> <p>-</p>
<b>10</b>	<p><b>Stellenwert der Note für die Endnote</b></p> <p>15/210</p>
<b>11</b>	<p><b>Modulbeauftragte/r und hauptamtlich Lehrende</b></p> <p>Prof. Dr. Grit Behrens</p>
<b>12</b>	<p><b>Sonstige Informationen</b></p> <p>Karsten Berns: "Autonomous Land Vehicles", Vieweg+Teubner 2009</p> <p>Russel Norvig: "Künstliche Intelligenz", Pearson 2004</p>

<b>Wahlfach aus Liste 2 „Anwendungen der Informatik“ Rich Client Application Development</b>					
<b>Kenn-Nr.</b> <b>5.16</b>	<b>Work-load</b> 450 h	<b>Credits</b> 15 cps	<b>Studien-semester</b> 5. /6. Sem.	<b>Häufigkeit des Angebots</b> Bei Nachfrage alle 2 Semester	<b>Dauer</b> 1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Vorlesung 2 SWS Praktikum 4 SWS	<b>Kontaktzeit</b> 6 SWS/90 h	<b>Selbststudium</b> 360 h	<b>geplante Gruppengröße</b> Vorlesung 60 Praktikum 15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden sind in der Lage, aktuelle Architekturen, Technologien und Werkzeuge zur Entwicklung von modernen Rich Client Anwendungen für Desktops und Browser einzusetzen. Die Studierenden lernen, für praxisnahe Problemstellungen eigenständig geeignete Lösungen zu finden und diese im praktischen Umgang mit Softwarewerkzeugen zielgerichtet zu implementieren und zu testen. Die erworbenen Fähigkeiten erweitern insbesondere die Technologie- und Realisierungskompetenzen im Bereich der Entwicklung von Anwendungen, die auf unterschiedlichen Endgeräten lauffähig sind.				
<b>3</b>	<b>Inhalte</b>  Grundlagen, Motivation, Übersicht  Architektur moderner Anwendungen Thin Server Architecture, MVP, MVVM  Übersicht relevanter Programmiersprachen  Javascript Entwurfsmuster  Persistenz NoSQL, MongoDB  Komponenten und Frameworks GWT, SWT, RCP				
<b>4</b>	<b>Lehrformen</b>  Vorlesung, Praktikum				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal:</b> -  <b>Inhaltlich:</b> Objektorientierte Programmierung, Datenbanken, Software Engineering,				

	Webbasierte Anwendungen
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO.
<b>7</b>	<b>Prüfungsformen</b> Projektarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Erfolgreicher Abschluss der Projektarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 15/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Jörg Brunsmann
<b>12</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Stoyan Stefanov: "Build Better Applications with Coding and Design Patterns", O'Reilly Media 2010</li> <li>• Mike Wilson: "Building Node Applications with MongoDB and Backbone", O'Reilly Media 2012</li> <li>• Mark Ethan Trostler: "Testable JavaScript", O'Reilly Media 2012</li> </ul>

<b>IT-Recht</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>6.1</b>	150 h	5 cps	6. Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Seminaristischer Unterricht 4 SWS	<b>Kontakt- unterricht</b> 4 SWS/60 h	<b>Selbst- studium</b> 90 h	<b>geplante Gruppengröße</b> Seminaristischer Unterricht 35	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die Studierenden kennen die wichtigsten rechtlichen Aspekte, mit denen sie während einer Tätigkeit im Gebiet der Informatik in Berührung kommen können. Insbesondere können sie einschätzen, welche Rechte und Pflichten bei Verträgen bezüglich Herstellung, Vertrieb und Nutzung von (Software/Hardware-) Produkten entstehen, mit welchen Schutzrechten geistiges Eigentum geschützt werden kann, wie Datenschutz beachtet werden muss, sowie mit welchen Folgen bei Rechtsverstößen zu rechnen ist.				
<b>3</b>	<b>Inhalte</b>  Die Informationstechnologie betreffenden Inhalte sind querschnittlich im Wesentlichen aus:  <ul style="list-style-type: none"> <li>• Zivilrecht und Vertragsrecht</li> <li>• Produkthaftung</li> <li>• Datenschutz</li> <li>• Strafrecht</li> <li>• Telekommunikations-, Telemedien- und Internetrecht</li> <li>• Gewerblicher Rechtsschutz (u.a. Urheber-, Patent-, Markenrecht)</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminaristischer Unterricht				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal: -</b>  <b>Inhaltlich: -</b>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				
<b>7</b>	<b>Prüfungsformen</b>  Hausarbeit, Klausur, mündliche Prüfung				
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b>  Teilnahme am Seminaristischen Unterricht mit Testat. Bestandene Modulprüfung				

<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Matthias König, Lehrbeauftragte
<b>12</b>	<b>Sonstige Informationen</b> Literatur: Schneider, Computerrecht – IT- und Computerrecht, 9. Aufl., DTV, 2010 Dreier, Vogel, Software- und Computerrecht, UTB, 2008

<b>Fachseminar</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>6.2</b>	150 h	5 cps	6.Sem.	Jedes SoSe	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Seminaristischer Unterricht 4 SWS	<b>Kontakt- unterricht</b> 4 SWS/60 h	<b>Selbst- studium</b> 90 h	<b>geplante Gruppengröße</b> Seminaristischer Unterricht 35	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Befähigung zum selbständigen wissenschaftlichen Arbeiten mit der Erschließung von Inhalten sowie deren verständliche Darstellung von fachlichen Themen sind für den beruflichen Alltag unabdingbar. Absolventen des Moduls <ul style="list-style-type: none"> <li>• beherrschen selbstständiges Erarbeiten eines Fachthemas anhand von Fachliteratur und anderen Quellen</li> <li>• können ein Fachthema verständlich darstellen und präsentieren</li> <li>• erwerben kommunikative Kompetenz</li> <li>• vertiefen informatische Fachkompetenzen auf dem ausgewählten Themengebiet des Seminars</li> </ul>				
<b>3</b>	<b>Inhalte</b>  Selbstorganisation und selbstständiges Erarbeiten eines Fachthemas <ul style="list-style-type: none"> <li>• Wissensmanagement und Literaturstudium (Recherche, Umgang mit Zitaten und Zitieren von Fachliteratur)</li> <li>• Fachliches Schreiben für die schriftliche Ausarbeitung</li> <li>• Präsentationstechnik und Rhetorik für die Präsentation des Themas</li> <li>• Diskussion im Rahmen der Seminarteilnehmer und betreuenden Dozenten</li> </ul>				
<b>4</b>	<b>Lehrformen</b> Seminaristischer Unterricht				
<b>5</b>	<b>Teilnahmevoraussetzungen</b>  <b>Formal: -</b>  <b>Inhaltlich: -</b>				
<b>6</b>	<b>Prüfungsvoraussetzungen</b>  Es gilt § 17 „Fortschrittsregelung“ dieser BPO.				

<b>7</b>	<b>Prüfungsformen</b> Hausarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Teilnahme am Seminaristischen Unterricht mit Testat. Bestandene Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 5/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Prof. Dr. Dominic Becking
<b>12</b>	<b>Sonstige Informationen</b> Publikationen zum gewählten Themengebiet in deutscher und englischer Sprache

<b>Praxisphase</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>7.1</b>	450 h	18 cps	7. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b> Tätigkeit bei individueller Praxisstelle		<b>Kontakt-zeit</b> 450 h	<b>Selbst-studium</b> Erfolgt am Arbeitsplatz	<b>geplante Gruppengröße</b> 1
<b>2</b>	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Die Praxisphase bietet den Studierenden die Möglichkeit, ihre in mehreren Semestern an der Hochschule erworbenen fachlichen Fähigkeiten in der Praxis zu erproben und zusätzlich wichtige Kompetenzen im außerfachlichen Bereich zu erwerben. Es spielt daher im Rahmen einer praxisorientierten und arbeitsmarktgerechten Ausbildung sowie zur Persönlichkeitsbildung eine zentrale Rolle. Die Lernergebnisse umfassen:</p> <ul style="list-style-type: none"> <li>• Orientierung im angestrebten Berufsfeld</li> <li>• Erwerb praktischer Kenntnisse und Kennenlernen berufstypischer Arbeitsweisen</li> <li>• Kennenlernen technischer und organisatorischer Zusammenhänge, die für das Berufsfeld typisch sind.</li> <li>• Beteiligung am Arbeitsprozess entsprechend dem Ausbildungsstand</li> <li>• Praktische Ausbildung an fest umrissenen, konkreten Projekten</li> </ul>				
<b>3</b>	<p><b>Inhalte</b></p> <p>Praktische Tätigkeit mit deutlichem Informatik-Schwerpunkt, z.B.</p> <ul style="list-style-type: none"> <li>• Durchführung von Anforderungsanalysen, Erstellung Softwareentwurf</li> <li>• Programmierung</li> <li>• Datenbankentwurf und -implementierung</li> <li>• Realisierung von Web-Anwendungen</li> <li>• Netzwerkplanung, Sicherheitsanalysen</li> <li>• Verarbeitung von Graphikdaten, Visualisierung</li> </ul> <p>Rahmenbedingungen:</p> <ul style="list-style-type: none"> <li>• Kürzere tägliche Arbeitszeiten als ein halber Tag sind nicht erlaubt.</li> <li>• Das gesamte Praktikum ist innerhalb von 9 Monaten abzuleisten.</li> <li>• Krankheits- und andere Ausfallzeiten zählen dabei nicht mit.</li> </ul>				

	<ul style="list-style-type: none"> <li>• Das Berufspraktikum muss bei einer Praktikumsstelle absolviert werden.</li> <li>• Über das Berufspraktikum ist ein Bericht von 20 Seiten anzufertigen.</li> <li>• Das Praktikum wird von einem hochschulseitigen Betreuer überwacht.</li> </ul>
<b>4</b>	<b>Lehrformen</b> Arbeitstätigkeit
<b>5</b>	<b>Teilnahmevoraussetzungen</b> Alle Modulprüfungen aus Semester 1 bis 4 bestanden (siehe auch § 24 BPO):
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO. Fristgerechte Abgabe des Praxisberichtes.
<b>7</b>	<b>Prüfungsformen</b> Bewerteter Praxisbericht als Testat
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Modulprüfung
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 18/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, König, Kreienkamp, Müller, Thiel)
<b>12</b>	<b>Sonstige Informationen</b> Arbeitsmaterialien und Literatur entsprechen der individuellen Aufgabenstellung

<b>Bachelorarbeit</b>					
<b>Kenn-Nr.</b>	<b>Work-load</b>	<b>Credits</b>	<b>Studien-semester</b>	<b>Häufigkeit des Angebots</b>	<b>Dauer</b>
<b>7.2</b>	360 h	12 cps	7. Sem.	Jedes WS	1 Semester
<b>1</b>	<b>Lehrveranstaltungen</b>  0,3 SWS Individuelle dozentengebundene  Betreuung	<b>Kontakt-zeit</b>  10 h	<b>Selbst-studium</b>  350 h	<b>geplante Gruppengröße</b>  1	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>  Die selbständige Lösung eines praxisbezogenen Themas nach wissenschaftlichen Grundsätzen gehört zu den beruflichen Fähigkeiten eines Informatikers. Die systematische Bearbeitung und praxisbezogene Umsetzung einer Aufgabenstellung sowie die zusammenhängende Darstellung von Berichten und Publikationen dient der Kommunikation zwischen Fachleuten und stellt sicher, dass erworbene Kenntnisse und Fähigkeiten erhalten bleiben. Die Studierenden lernen, wie eine Aufgabe methodisch bearbeitet und in einem vorgegebenen Zeitrahmen mit einem klar strukturierten Ergebnis dargestellt wird, indem sie sich schnell in eine neue Aufgabenstellung einarbeiten und ihr Wissen in einem speziellen Gebiet selbstständig vertiefen. Die Studierenden lernen übliche Werkzeuge und Methoden zur Arbeitsunterstützung einzusetzen, wenden eine Reihe fachspezifischer Fähigkeiten, Fertigkeiten und Techniken an, um Aufgaben selbstständig zu lösen, diese zu analysieren, zu bewerten und in einem Gesamtkontext darzustellen.				
<b>3</b>	<b>Inhalte</b>  Mit der Bachelorarbeit soll unter Beweis gestellt werden, dass Studierende in der Lage sind, eine komplexe Fragestellung mit wissenschaftlichen Methoden innerhalb eines begrenzten Zeitraums zu lösen und das dabei erworbene theoretische und praktische Wissen nachvollziehbar zu dokumentieren.  1. Konkretisieren der Aufgabenstellung  2. Erstellung eines Zeitplans  3. Evaluation und Aufstellung der zu verwendenden Techniken und Methoden  4. Erstellung eines Software-Konzeptes  5. Implementierung und Dokumentation der Software-Lösung  6. Gesamtbetrachtung, Test und Bewertung der Lösung  7. Darstellung der Lösung in Form der Bachelor-Arbeit				
<b>4</b>	<b>Lehrformen</b> Individuelle dozentengebundene Betreuung				

	Selbständige Erstellung der Bachelorarbeit
<b>5</b>	<b>Teilnahmevoraussetzungen</b> Bestandene Modulprüfungen gem. Studienverlaufsplan bis auf vier Modulprüfungen (siehe auch BPO §29). Kenntnisse in der Breite des studierten Faches
<b>6</b>	<b>Prüfungsvoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO. Fristgemäße Abgabe der Bachelorarbeit im Prüfungsamt (BPO §31)
<b>7</b>	<b>Prüfungsformen</b> Von zwei Prüfern bewertete Bachelorarbeit
<b>8</b>	<b>Voraussetzungen für die Vergabe von Kreditpunkten</b> Bestandene Bachelorarbeit
<b>9</b>	<b>Verwendung des Moduls (in anderen Studiengängen)</b> -
<b>10</b>	<b>Stellenwert der Note für die Endnote</b> 12/210
<b>11</b>	<b>Modulbeauftragte/r und hauptamtlich Lehrende</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, König, Kreienkamp, Müller, Thiel)
<b>12</b>	<b>Sonstige Informationen</b> Arbeitsmaterialien und Literatur entsprechen der individuellen Aufgabenstellung