



Modulkatalog
Softwareengineering – Bachelor of Engineering
(B.Eng.)



ALG20 Algorithmen

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Grundlegende Ansätze zur Wahl von Algorithmen kennenlernen und anwenden können. Zur Problemlösung geeignete Datenstrukturen auswählen. Algorithmen zum Suchen, Sortieren kennen und anwenden lernen. Fragen zur Laufzeit von Algorithmen kennenlernen und für ausgewählte Algorithmen berechnen können. Praktische Umsetzung von Algorithmen in eine Programmiersprache durchführen können. Wichtige Fachbegriffe kennen und in das aktive Vokabular aufnehmen.
Inhalt	Algorithmen und Datenstrukturen Grundlagen Mengenproblem: Suchalgorithmen Sortieren Kürzester Weg Rundreiseproblem Flussproblem Gierige Algorithmen Sortieren Kürzeste Wege in Graphen Minimal aufspannende Graphen Flüsse in Graphen Teile und Herrsche Sortieren Quicksort Dynamisches Programmieren Kürzeste Wege Optimale Suchbäume Direkter Zugriff Sortieren durch Abzählen Haschen Prioritätswarteschlangen Binäre Heaps Heapsort
Voraussetzungen	Grundkenntnisse in einer Programmiersprache. Vorzugsweise Java oder Python
Modulbausteine	Fachbuch: Algorithmen und Datenstrukturen, K. und N. Weicker, Springer-Verlag 2013 ALG201-BH Begleitheft Algorithmen und Datenstrukturen mit Onlineübungen Onlinetutorium (1 Stunde)



Kompetenznachweis Klausur (2 Stunden)

Lernaufwand 125 Stunden, 5 Leistungspunkte

Sprache Deutsch

Studienleiter Andrea Herrmann



ANS43 **Geschäftsprozesse und Anwendungssysteme**

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Nach erfolgreicher Teilnahme am Modul ANS43 erwerben die Studierenden Grundlagenwissen zu betrieblichen Anwendungs- und Informationssystemen AS/IS.</p> <p>Sie klassifizieren AS/IS und skizzieren den grundlegenden Aufbau und Architektur von AS/IS.</p> <p>Sie können Formen und Ansätze zur Integration von AS/IS beschreiben und jeweils die Beispielsysteme benennen. Weiterhin beurteilen sie die prozessorientierte Ausrichtung von AS/IS.</p> <p>Die Studierenden beschreiben die ARIS Methode und wenden sie an. Sie kennen die Ansätze funktionsbezogener und integrierter Anwendungssysteme und beurteilen diese.</p> <p>Funktionale und integrierte Systemlösungen werden gegenübergestellt und bewertet. Anwendungssysteme werden definiert, systematisiert und abgegrenzt.</p> <p>Sie sind in der Lage ein Gesamtkonzept der integrierten Informationsverarbeitung und die Bedeutung der Prozessorientierung für Anwendungssysteme zu erläutern.</p> <p>Die Studierenden beschreiben ein konkretes Referenzmodell für Prozesse in Industriebetriebe. Weiterhin beschreiben sie aktuelle Trends und branchenspezifische Ausrichtungen von Anwendungssystemen.</p> <p>Darüber hinaus kennen und beurteilen sie Ansätze und Architekturen zur Enterprise Application Integration (EAI).</p> <p>Weiterhin sind sie in der Lage die charakteristischen Merkmale einer serviceorientierten Architektur (SOA) als einen wichtigen Integrationsansatz zu benennen.</p> <p>Sie entwickeln Bewertungskriterien für betriebswirtschaftliche Standardsoftware (SSW) und ihre Infrastruktur-Komponenten und wenden sie auf eine typische Unternehmenssituation an.</p> <p>Weiterhin sind sie in der Lage eine einfache EAI-Lösung im Rahmen einer vorgegebenen Fallbeschreibung zu konzipieren (Fach-, Methoden- und soziale Kompetenz).</p>
-----------------------	---

Inhalt	<p>Konzepte betrieblicher Anwendungssysteme</p> <p>Begriffsdefinition und Abgrenzung</p> <p>Aufgaben und Einteilung betrieblicher Anwendungssysteme</p> <p>Impulsgeber „Fehlende Integration“</p> <p>Aufbau betrieblicher Anwendungssysteme</p> <p>Architekturbeispiele</p> <p>Typen und Herkunft von Anwendungssoftware</p> <p>Geschäftsprozessmodellierung</p> <p>Geschäftsprozesse</p> <p>Das ARIS-Konzept</p> <p>Ereignisgesteuerte Prozessketten (EPK)</p> <p>Objektorientierte Geschäftsprozessmodellierung</p> <p>Ausblick: Prozessmodellierung heute und morgen</p> <p>Musterfallstudie</p> <p>Grundlagen und Hinführung</p>
---------------	---



Szenario und Hintergrund
In Phasen zum Ziel
ERP goes Internet: auf welche Standards geachtet werden muss
Anlagen zur Auswahlphase
Einsatzbereiche von Anwendungssystemen
Integration von Anwendungsmodellen
Integrationsmodelle
Architektur von Informationssystemen
Referenzmodell der integrierten Informationsverarbeitung in der Industrie
Funktionsbereich- und prozessübergreifende Integrationskomplexe

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	ANS101 Studienbrief Konzepte betrieblicher Anwendungssysteme mit Onlineübung ANS102 Studienbrief Geschäftsprozessmodellierung mit Onlineübung ANS201 Studienbrief Musterfallstudie Fachbuch Mertens: Integrierte Informationsverarbeitung mit ANS301-BH Begleitheft und Onlineübung Fallstudie Optimierung der innerbetrieblichen Logistik bei der Marcus Lang GmbH Online-Seminar (2 Std.)
-----------------------	---

Kompetenznachweis	Assignment
--------------------------	------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Andrea Herrmann
----------------------	-----------------



BWL26 BWL-Grundlagen

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	Zentrale strategische, organisatorische und rechtliche Fragen bei der Gründung von Unternehmen erläutern; betriebliche Funktionsbereiche (primäre und sekundäre) in Unternehmen und ihre grundlegenden Methoden erklären; für ein Beispielunternehmen den Leistungsprozess im engeren Sinne analysieren; Marktchancen und finanzielle Struktur des Unternehmens bewerten; grundlegende Aufgaben und Instrumente des Personalmanagements und des Rechnungswesens aufzählen und beurteilen.
-----------------------	---

Inhalt	<p>Grundlagen des Produktionsmanagements und der Materialwirtschaft</p> <p>Begriffliche Grundlagen Geschichte der industriellen Produktion – ein Überblick Produktionsmanagement Materialwirtschaft</p> <p>Marketing</p> <p>Wandel der Märkte und des Marketings Wie kommt es zu einer Kaufentscheidung? – Eine Analyse des Kaufverhaltens Informationsbeschaffung für das Marketing: die Marketingforschung Marketingkonzeption – Ergebnis eines systematischen Vorgehens im Marketing Ausgangspunkt für die Entwicklung einer Marketingkonzeption: die Umwelt- und Unternehmensanalyse Entwicklung von Marketingzielen und Marketingstrategie Marketinginstrumentarium und Marketingmix Marketingcontrolling Organisation der Marketingfunktion</p> <p>Personalmanagement</p> <p>Grundlagen des Personalmanagements Rechtliche Grundlagen des Personalmanagements Personalplanung Personalbeschaffung Personaleinsatz Personalentwicklung Betriebliche Anreizsysteme Personalbeurteilung Personalcontrolling und Personaldatenverwaltung Personalführung</p> <p>Rechnungswesen</p> <p>Grundlagen Finanzbuchhaltung Kosten- und Leistungsrechnung Spezialaufgaben des Rechnungswesens</p> <p>Grundlagen der Unternehmensführung</p> <p>Was ist Unternehmensführung</p>
---------------	--



St. Galler Managementkonzept
Normatives Management
Strategisches Management
Operatives Management

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	RAE101-EL Studienbrief mit Rechtsänderungen BWL103 Studienbrief Grundlagen des Produktionsmanagements und der Materialwirtschaft mit Onlineübung BWL104 Studienbrief Marketing mit Onlineübung BWL105 Studienbrief Personalmanagement mit Onlineübung BWL106 Studienbrief Rechnungswesen mit Onlineübung BWL107 Studienbrief Grundlagen der Unternehmensführung mit Onlineübung Onlinetutorium (1 Stunde)
-----------------------	---

Kompetenznachweis	Klausur (1 Stunde)
--------------------------	--------------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Beate Holze
----------------------	-------------



CPP23 Programmieren in C/C++ mit Labor

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Prinzipien der Programmierung in C und C++ verstehen; Unterschiede zwischen prozeduralem und projektorientierten Programmieransatz erläutern; grundlegende Sprachelemente von C++ kennen und anwenden; einfache funktions- und objektorientierte Programme in C++ erstellen und zum Ablauf bringen; mit einer Programm-Entwicklungsumgebung für C++ umgehen.</p> <p>Aufgabe und Zweck von Softwaredokumentation erläutern und die verschiedenen Dokumentationsarten beschreiben; wissen, was gute Dokumentation ausmacht.; Kosten und Nutzen von Softwaredokumentation bezogen auf verschiedene Einsatzgebiete gegenüberstellen; wissen, warum es Sinn macht, Softwaredokumentation automatisch zu erstellen; die Bedeutung einer guten Dokumentation für die Weiterentwicklung und Wartung von Software erkennen sowie die speziellen Formen der Software-Wartungsdokumentation erläutern.</p>
-----------------------	--

Inhalt	<p>Programmieren in C Einführung in das Programmieren Einführung in C Weiterführende Konzepte</p> <p>Programmieren in C++ Einführung in die prozedurale Programmierung in C++ Weiterführende Konzepte Objektorientierte Programmierung</p> <p>Einführung in die Programmierung mit C++ Das Arbeiten mit einer Entwicklungsumgebung Einstieg in die Programmierung Ausdrücke und Anweisungen Strukturierte Anweisungen Zusammengesetzte Datentypen Zeiger Funktionen Stack und Heap Klassen und Objekte Vererbung und Polymorphie Generische Programmierung Wichtige Bibliotheken Container und Iteratoren Unified Modelling Language</p> <p>Softwaredokumentation Systematisierung der Softwaredokumentation Vorgehensweise bei der Softwaredokumentation Online-Dokumentation Gedruckte Softwaredokumentation Das Projektmanagement bei der Softwaredokumentation Softwaredokumentationsqualität als Managementaufgabe</p>
---------------	---



Feldstudie zur Untersuchung der Dokumentationsqualität

Softwaredokumentationswerkzeuge

Klassifikation der Dokumentation und ihre Werkzeuge

Vertragsdokumente

Dokumentationswerkzeuge in der Anforderungsanalyse

Werkzeuge zur Source-Code Dokumentation

Autorensysteme

ITA und topicorientierte Dokumentation

Softwaredokumentation und Softwarewartung

Grundlagen und Begriffe

Arten, Aufwand und Risiken der Softwarewartung

Organisation der Softwarewartung

Management der Softwarewartung

Softwaredokumentation

Testen in der Softwarewartung

Metriken der Softwarewartung

Werkzeuge zur Unterstützung der Softwarewartung

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	<p>CPP109 Studienbrief Programmierung in C mit Onlineübung</p> <p>CPP110 Studienbrief Programmierung in C++ mit Onlineübung</p> <p>Fachbuch Kirch, C++ Lernen und professionell anwenden</p> <p>Fachbuch Kirch, C++ Uebungen 2018</p> <p>CPP101 Studienbrief Einführung in die Programmierung mit C++ mit Onlineübung</p> <p>MIP401 Studienbrief Softwaredokumentation mit Onlineübung</p> <p>MIP402 Studienbrief Softwaredokumentationswerkzeuge mit Onlineübung</p> <p>MIP403 Studienbrief Softwaredokumentation und Softwarewartung mit Onlineübung</p> <p>Präsenztutorium (2 Tage, Programmierübung)</p> <p>Labor (1 Tag, Programmierung mit automatischen Tools zur Softwaredokumentationserstellung. Definieren des Software-Projektes und Starten der Projektdurchführung in einer Gruppe)</p> <p>Onlineseminar (1 Stunde)</p>
-----------------------	--

Kompetenznachweis	Assignment (Laborbericht)
--------------------------	---------------------------

Lernaufwand	250 Stunden, 10 Leistungspunkte
--------------------	---------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Andrea Herrmann
----------------------	-----------------

DBA23 Datenbanken

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	<p>Grundkenntnisse in Dateiorganisation, Datenmodellierung und Datenbanken: Die Basistechniken der physischen und logischen Datenorganisation beschreiben, einfache ER-Modelle erstellen, daraus relationale Datenmodelle ableiten, diese über Normalformen optimieren sowie einen Überblick über die Aufgaben und den Aufbau von Datenbanksystemen geben.</p> <p>Architektur und Funktionsweise von Datenbanken sowie die Vorgehensweise beim Entwurf von Datenbanken beschreiben. ERM zum Datenbankentwurf anwenden. Die Befehle von SQL zur Datendefinition und zur Datenmanipulation kennen und verwenden. Konzepte zur Datenintegrität erläutern. Eine einfache relationale Datenbank aufbauen und benutzen. (Fach- und Methodenkompetenz.)</p>
Inhalt	<p>Vom Datenmodell zur Speicherung in Dateien</p> <p>Allgemeines zur Datenorganisation Entity-Relationship-Modelle Relationale Datenmodellierung Physische Datenorganisation</p> <p>Datenbanksysteme</p> <p>Structured Query Language Grundlagen von Datenbanksystemen Datenbanken in der Informationstechnologie Konzepte und Architekturen Logische Datenmodelle Einsatz von Datenbanksystemen im Unternehmen</p> <p>Datenbankentwurf</p> <p>Einführung: Prozess des Datenbankentwurfs im Überblick Konzeptuelle Modellierung Logische Modellierung: Umsetzung ins Relationenmodell Qualität des Datenbankentwurfs: Normalformen-Theorie Physische Modellierung</p> <p>SQL – Structured Query Language</p> <p>Datenbanksprachen und Datenbanksysteme Lebenszyklus einer Datenbankanwendung Datendefinition mit SQL Datenmanipulation mit SQL Einsatz von Sichten mit SQL Einbindung von SQL in andere Sprachen</p> <p>Erweiterte Konzepte von Datenbanksystemen</p> <p>Sicherheit und Zugriffskontrolle Ablaufsteuerung mit Transaktionen Wiederherstellung (Recovery) des DBS Performanz von Datenbanksystemen Weitere Datenbankkonzepte und Technologien</p>



Voraussetzungen

Grundlagen der Informatik

Modulbausteine

DAO101 Studienbrief Vom Datenmodell zur Speicherung in Dateien mit **Onlineübung**

DBA101 Studienbrief Grundlagen von Datenbanksystemen mit **Onlineübung**

DBA102 Studienbrief Datenbankentwurf mit **Onlineübung**

DBA103 Studienbrief SQL – Structured Query Language mit **Onlineübung**

DBA104 Studienbrief Erweiterte Konzepte von Datenbanksystemen mit **Onlineübung**

Onlineseminar (2 Stunden)

Onlinetutorium (1 Stunde)

Kompetenznachweis

Klausur (1,5 Stunden)

Lernaufwand

125 Stunden, 5 Leistungspunkte

Sprache

Deutsch

Studienleiter

Dr. Franz-Karl Schmatzer



DBA62 Nicht-Standard-Datenbanken

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Grundkenntnisse in der Dateioorganisation von Nicht-Standard-Datenbanken. Architektur und Funktionsweise von NoSQL-Datenbanken sowie die Befehle zur Datendefinition und zur Datenmanipulation kennen und verwenden können. Konzepte zur Datenintegrität und Transaktionen erläutern können. Die Probleme verteilter Datenbanken und deren Synchronisation kennen und beschreiben können. Eine einfache NoSQL-Datenbank aufbauen und nutzen können. (Fach- und Methodenkompetenz.)
Inhalt	Einführung in NoSQL-Systeme Ausgewählte Konzepte von NoSQL-Systemen Das Map/Reduce Framework CAP Theorem Verschiedene Konsistenzmodelle Zeitmessung in verteilten Systemen (Global Clock Problem) Concurrency-Control REST-Framework Ausgewählte NoSQL-Datenbanken Column Store Document Store Key/Value-Datenbanken Graphendatenbanken
Voraussetzungen	Grundlagen Datenbanken
Modulbausteine	Fachbuch Edlich/Friedland/Hampe/Brauer: NoSQL – Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken mit einem Begleitheft und einer Onlineübung
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Wolfgang Riggert



DBA67 Datenbank-Programmierung

Kompetenzzuordnung	Instrumentale Kompetenz
Kompetenzziele	<p>Datenbank Anbindung mittels JDBC kennenlernen und programmieren; Datenschemata anlegen, Daten speichern und Daten mittels JDBC abfragen.</p> <p>Den Persistence-Layer der EE-Architektur verstehen und nutzen; die Anbindung von Datenbanken mittels JDBC programmieren; die Java Transaktions-API kennenlernen; ein objektorientiertes Datenbanksystem kennenlernen; die Speicherung und das Abfragen in diesem System programmieren.</p>
Inhalt	<p>JDBC Einführung in JDBC Zugriff auf DBMS über JDBC</p> <p>Java Persistence API Einführung Objektrelationen Der Entity Manager Abfragen Vererbung Lebenszyklusmethoden</p> <p>Java Transaktion API Einführung Objektrelationen Queries Mapping-Strategien</p> <p>Objektdatenbank db4o Einführung CRUD-Operationen Objektidentität Native Abfragen Tiefe Objektgraphen Callbacks</p>
Voraussetzungen	<p>Einführung in Datenbanksysteme Grundlagen zur EE-Programmierung</p>
Modulbausteine	<p>JAV605 Studienbrief EE-Persistence Layer in Java mit Onlineübung</p> <p>JAV606 Studienbrief Fallbeispiele zur Programmierung von Datenbank-Anbindung mittels Java</p> <p>Fachbuch Abts: Grundkurs JAVA – Von den Grundlagen bis zu Datenbank- und Netzanwendungen</p> <p>Programmierlabor (1 Tag)</p>
Kompetenznachweis	Assignment (Laborbericht)



Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



EBS44 Mobile Computing

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Aufbau und technische Funktionsweise moderner mobiler Netzwerke kennenlernen und verstehen; Software-Architekturen für mobile Anwendungen im betrieblichen Umfeld kennenlernen und beurteilen können.
Inhalt	Grundlagen drahtloser Netze Einführung Physikalische Grundlagen Nachrichtentechnische Grundlagen Multiplex- und Medienzugriffsverfahren Das Mobilfunknetz – von GSM zu LTE Mobile Datenbanksysteme - Architektur, Implementierung, Konzepte WLAN-Systeme Einleitung Der IEEE 802.11 – Ein Überblick Die Bitübertragungsschicht Die Sicherungsschicht Sicherheit Verfahren zur Ortung und Navigation Motivation und Hintergrund Kennenlernen – Ortung und Sensoren Kennenlernen – Navigation Üben anhand eines Beispiels RFID Motivation Klassifizierung Grundlagen Normen Sicherheit Anwendungen Architektur mobiler Informationssysteme Architektur Software-intensiver Systeme Mobile Anwendungssysteme Mobile Anwendungen für Android
Voraussetzungen	Anwendungskennnisse im Bereich der Microcomputer-Systeme
Modulbausteine	EBS201 Studienbrief Grundlagen drahtloser Netze mit Onlineübung AB24-624 Fachbuch Mutschler; Specht: Mobile Datenbanksysteme – Architektur, Implementierung, Konzepte EBS203 Studienbrief WLAN-Systeme mit Onlineübung EBS204 Studienbrief Verfahren zur Ortung und Navigation mit Onlineübung EBS205 Studienbrief RFID mit Onlineübung



EBS206 Studienbrief Architektur mobiler Informationssysteme mit
Onlineübung

Kompetenznachweis	Assignment
--------------------------	------------

Lernaufwand	150 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Patrick Stepke
----------------------	----------------



EBS45 Embedded Software Entwicklungsprozess

Kompetenzzuordnung	Wissensvertiefung
---------------------------	-------------------

Kompetenzziele	<p>Vermittlung vertiefender Kenntnisse zum Design von Mikroprozessor- bzw. Mikrocontrollersystemen;</p> <p>Einsatz und Kopplung vorgefertigter Hardware-Funktionsmodule zu kompletten eingebetteten Systemen;</p> <p>Vermittlung von Fachkompetenzen für einen Systementwurf in Hard- und Software sowie einer nutzer-spezifischen Logik.</p>
-----------------------	---

Inhalt	<p>Modellierung eingebetteter Systeme</p> <p>Petrinetze</p> <p>VHDL</p> <p>Statechart</p> <p>SDL</p> <p>Fallstudie: Fahrstuhlsteuerung</p> <p>UML</p> <p>Hardware eingebetteter Systeme</p> <p>Prozessoren und Controller</p> <p>Geräte: Interne Kommunikation</p> <p>Geräte: Externe Kommunikation</p> <p>Anbindung analoger Sensoren und Aktoren</p> <p>Systemsoftware eingebetteter Systeme</p> <p>Scheduling</p> <p>Klassifikation</p> <p>Aperiodisches Scheduling</p> <p>Periodisches Scheduling</p> <p>Embedded Betriebssysteme</p> <p>Anforderungen, Übersicht</p> <p>Implementierung von eingebetteten Systemen</p> <p>Systemintegration</p> <p>Einrichten einer Testumgebung</p> <p>Systemintegration: Programmieren, Debuggen, Testen sowie Inbetriebnahme von eingebetteten Systemen</p> <p>Aufbau von Mehrprozessorsystemen</p> <p>Applikationsbeispiele</p>
---------------	--

Voraussetzungen	<p>Einführung in Mikrocontroller</p> <p>Einführung in eine Programmiersprache</p>
------------------------	---

Modulbausteine	<p>EBS301 Studienbrief Modellierung eingebetteter Systeme</p> <p>EBS302 Studienbrief Hardware eingebetteter Systeme</p> <p>EBS303 Studienbrief Systemsoftware eingebetteter Systeme</p> <p>EBS304 Studienbrief Implementierung von eingebetteten Systemen</p> <p>Labor (1 Tag)</p>
-----------------------	---



Kompetenznachweis	Assignment (Laborbericht)
Lernaufwand	250 Stunden, 10 Leistungspunkte
Sprache	Deutsch
Studienleiter	Patrick Stepke



EBS65 Echtzeitsysteme

Kompetenzzuordnung	Wissensvertiefung
Kompetenzziele	Grundlagen und Anwendungen von Echtzeitsystemen kennen; Anforderungen an Sicherheit und Zuverlässigkeit von Echtzeitsystemen einschätzen; Hardware Komponenten auf Echtzeitfähigkeit beurteilen und auswählen; Aufgaben und Funktionsweise von Echtzeit-Betriebssystemen kennen; Grundlagen für Entwurf und Programmierung von Microcomputer-Systemen für zeitkritische Anwendungen kennen und anwenden; die Prinzipien der digitalen Computerschnittstelle zur Außenwelt verstehen und beurteilen; den Einsatz und die Verwendung der seriellen und parallelen Standardschnittstellen sicher beherrschen; ausgewählte Bussysteme der Industrie im Bereich der Automatisierung und der Fahrzeugindustrie kennenlernen und beurteilen.
Inhalt	Grundlagen Echtzeitsysteme Einführung Realzeit-Scheduling Software in Echtzeitsystemen Echtzeit-Betriebssysteme Angewandtes Real Time Scheduling Programmiersprachen Verteilte Echtzeitanwendungen Verteilte Systeme Synchronisation Echtzeitkommunikation Standards Einführung in die industriellen Kommunikations-Bussysteme Vorbemerkungen Leitungen und Übertragungsmedien Impulse und Leitungen Serielle und parallele Schnittstellen Bussysteme Parallele Busse Serielle Busse Bussysteme im Bereich der Automatisierung und der Fahrzeugindustrie Vorbemerkungen Anforderungen an industrielle Bussysteme Fehlersicherung und Restfehlerrate Bussysteme in der Fahrzeugtechnik Bussysteme in der Automatisierungstechnik Ethernet-basierte Feldbussysteme
Voraussetzungen	Grundkenntnisse in mindestens einer Programmiersprache
Modulbausteine	SYS201 Studienbrief Grundlagen Echtzeitsysteme mit Onlineübung SYS202 Studienbrief Software in Echtzeitsystemen mit Onlineübung



SYS203 Studienbrief Verteilte Echtzeitanwendungen mit **Onlineübung**

IKB101 Studienbrief Einführung in die industriellen Kommunikations-Bussysteme mit **Onlineübung**

IKB102 Studienbrief Bussysteme im Bereich der Automatisierung und der Fahrzeugindustrie mit **Onlineübung**

Kompetenznachweis	Klausur (120 Minuten)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Patrick Stepke

EBS66 Ein- und Ausgaben auf mobilen Endgeräten

Kompetenzzuordnung	Instrumentale Kompetenz
Kompetenzziele	Prinzipien der Usability für mobile Endgeräte kennen und beurteilen; applikationsspezifische Anforderungen an kleine Benutzeroberflächen ableiten; Testverfahren für die Bewertung der Usability erläutern, die Anwendungsmöglichkeiten dieser Verfahren für die App-Entwicklung beurteilen; Typografien, Farbschemata für die App-Entwicklung kennenlernen und beurteilen; den Einsatz von Icons und Bildern einschätzen und beurteilen können; offene Forschungsfragen zur App-Usability kennen.
Inhalt	<p>Entwicklung einer mobilen App</p> <p>Übersicht der Systeme</p> <p>Was ist eine App?</p> <p>Entwicklungsstrategie einer App</p> <p>Szenariobasierte Entwicklung</p> <p>Contextual Design</p> <p>Testen von mobilen Apps</p> <p>Usability für Apps</p> <p>Übersicht: Stellenwert und Gestaltungsvorgaben</p> <p>Nutzerzentriertes Design</p> <p>Ausgabemöglichkeiten</p> <p>Bedienungstechniken für Smartphones</p> <p>Tablet-Bedienung</p> <p>Bedienung von Wearables</p> <p>UI-Prinzipien für Apps</p> <p>Controls and Views</p> <p>Standard Controls für iOS</p> <p>Standard Controls für Android</p> <p>Custom-Controls</p> <p>Typografie und Farbe</p> <p>Kategorien von Schriften</p> <p>Regeln für gute Typografie im App-Design</p> <p>Farbschemata</p> <p>Farben im App-Design</p> <p>Farbdarstellung auf Smartphones und Tablets</p> <p>Icons, Grafiken und Bilder</p> <p>Icons</p> <p>Grafiken und Fotografien</p> <p>Tools für den Export</p> <p>Innovative Ein- und Ausgaben</p> <p>Sprachein- und -ausgabe</p> <p>Neuere Trends</p>
Voraussetzungen	Keine.



Modulbausteine	Fachbuch Semler; Tschierschke: App-Design – Das umfassende Handbuch Fachbuch Nielsen; Budiu: Mobile Usability – Für iPhone, Android, Kindle EBS602-BH Begleitheft zu den Fachbüchern EBS603-RG Research Guide
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Mark Harwardt



EBS67 App-Programmierung

Kompetenzzuordnung	Wissensverbreiterung, Instrumentale Kompetenz
Kompetenzziele	Die Prinzipien der App-Programmierung verstehen; den Ansatz der Plattformunabhängigkeit bei der App-Entwicklung anhand von LiveCode kennen und umsetzen; die Möglichkeiten von LiveCode kennen und umsetzen; Kennenlernen von Android Studio und der zugehörigen Entwicklungsumgebung; die Programmierung einer App-Anwendung durchführen und implementieren.
Inhalt	Programmieren mit LiveCode Installation der Entwicklungsumgebung Programmierkonzept von LiveCode Skript-Programmierung Systematische Code-Entwicklung mit LiveCode Fallbeispiele Stand Alone Entwicklung Portierung auf mobile Endgeräte App Programmieren mit Android Studio Installation der Entwicklungsumgebung Entwickeln der App Design Testen Fallbeispiele
Voraussetzungen	Einführung in die Programmierung
Modulbausteine	Fachbuch Fehr: Eigene Apps programmieren EBS604-BH Begleitheft zum Fachbuch EBS605 Studienbrief App-Programmierung mit Android Studio mit Onlineübung
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Andrea Herrmann



EBS68 Software-Anforderungen für mobile Endgeräte

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	<p>Das Anforderungsmanagement als integralen Bestandteil des Software-Entwicklungsprozesses verstehen; erkennen, dass Softwareentwicklung kostengünstiger und mit besseren qualitativen Ergebnissen erfolgt, wenn ein professionelles Anforderungsmanagement umgesetzt wird;</p> <p>Systematische Anforderungsanalyse beherrschen und anwenden; Werkzeuge für das Anforderungsmanagement kennen und beurteilen; die spezifischen Anforderungen mobiler Endgeräte kennenlernen und umsetzen; für einen ausgewählten Anwendungsfall eine Vorgehensweise für das Anforderungsmanagement für mobile Endgeräte entwickeln.</p>
Inhalt	<p>Grundlagen Requirements Engineering</p> <p>Requirements Engineering</p> <p>Tätigkeiten im Requirements Engineering</p> <p>Methoden des Requirements Engineering</p> <p>Anforderungsfeststellung</p> <p>Requirements ermitteln, analysieren und modellieren</p> <p>Requirements spezifizieren</p> <p>Requirements verifizieren und validieren</p> <p>Anforderungsbearbeitung</p> <p>Requirements vereinbaren</p> <p>Requirements verwalten</p> <p>Werkzeuge für das Requirements Engineering</p> <p>Anforderungen an mobile Endgeräte</p> <p>Übersicht der Systeme</p> <p>Anforderungen an die Software</p> <p>Spezifische Anforderungen für beschränkte Ressourcen</p> <p>Spezifische Werkzeuge für das Requirements Engineering mobiler Endgeräte</p> <p>Entwicklung eines Konzeptes zur Ermittlung der Anforderungen für ein mobiles Endgerät</p> <p>Entwicklung eines Vorgehensmodells für ein Fallbeispiel</p> <p>Aufzeigen der Arbeits- und Abstimmungsschritte</p> <p>Diskussion unterschiedlicher Möglichkeiten der Anforderungsdefinition</p>
Voraussetzungen	Kenntnisse im Bereich der Softwareentwicklung
Modulbausteine	<p>RER811 Studienbrief Grundlagen Requirements Engineering mit Onlineübung</p> <p>RER812 Studienbrief Anforderungsfeststellung mit Onlineübung</p> <p>RER813 Studienbrief Anforderungsbearbeitung mit Onlineübung</p> <p>RER817 Studienbrief Spezifische Anforderungen mobiler Endgeräte mit Onlineübung</p>



Kompetenznachweis Assignment

Lernaufwand 125 Stunden, 5 Leistungspunkte

Sprache Deutsch

Studienleiter Andrea Herrmann



EIT22 English for Computer Science - Introduction

Kompetenzzuordnung	Kommunikative Kompetenz
---------------------------	-------------------------

Kompetenzziele	<p>Englischsprachige E-Mails verstehen und selbst verfassen, englische Telefongespräche führen, an englischsprachigen Meetings teilnehmen können. Den wichtigsten Wortschatz und Grammatik für Besprechungen anwenden.</p> <p>Fachsprachliche Grundkenntnisse aus dem Technikbereich beim Sprechen, Lesen, Schreiben und Hören beherrschen.</p> <p>Aufgrund fachspezifischer Englischkenntnisse branchentypische kommunikative Situationen im IT-Bereich erfolgreich meistern.</p> <p>Fachspezifisches Vokabular im IT- und kaufmännischen Bereich anwenden.</p> <p>Englische Grundgrammatik beim Schreiben und Sprechen beherrschen, eine große Anzahl von Strukturen und Funktionen des Englischen sicher anwenden;</p>
-----------------------	---

Inhalt	<p>Interaktives Training</p> <p>Telefonate sicher führen verschiedene berufliche Gesprächssituationen Vorträge und Besprechungen Geschäftliche Dokumente wie z.B. Berichte, Besprechungsprotokolle, Briefe oder Broschüren Verhandlungen führen informelle Kommunikationssituationen Kontakt mit Kunden und Geschäftspartnern Wortschatz aus der Informatikindustrie und Telekommunikationsindustrie</p> <p>Exploring an IT company, dealing with customers, ordering, managing a networking project</p> <p>Exploring an IT company Looking after customers Handling orders and complaints Managing a networking project</p> <p>Installing hardware and software, troubleshooting</p> <p>Installing and configuring hardware Installing and customizing software Troubleshooting</p> <p>IT security, trade fairs, applying for a job</p> <p>IT security Trade fairs Applying for a job</p>
---------------	--

Voraussetzungen	Englischkenntnisse auf Niveau B2
------------------------	----------------------------------



Modulbausteine

Online-Content Rosetta Stone: B2: Situations: All Topics; Professions: IT; Areas of Expertise: Computers and IT, Telecommunications

MP3 English for IT

EIT101 Studienbrief Exploring an IT company, dealing with customers, ordering, managing a networking project

EIT102 Studienbrief Installing and configuring software, troubleshooting

EIT103 Studienbrief IT security, trade fairs, applying for a job

Einsendeaufgaben zu den Studienbriefen EIT101-103

Kompetenznachweis	Klausur (2 Stunden)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Englisch
Studienleiter	Verena Jung



FMI23 Formale Methoden der Informatik

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Die Begriffe Information und Codierung aus formaler Sicht beschreiben. Algorithmen definieren, ihre Prinzipien und Komplexität erkennen sowie den Ansatz der Rekursion erläutern. Graphen als anschauliche Darstellungen einsetzen. Das Erzeugen von formalen Sprachen durch Grammatiken sowie das Erkennen von Sprachen mittels Automaten beschreiben. Die Turingmaschine als minimales Automatenmodell von Computern erläutern. (Fach- und Methodenkompetenz.)
Inhalt	<p>Automaten und formale Sprachen, Teil I: Endliche Automaten und reguläre Sprachen</p> <p>Mathematische Notationen Deterministische Automaten Reguläre Sprachen Nichtdeterministische Automaten Minimierung deterministischer Automaten Reguläre Ausdrücke Grammatiken für reguläre Sprache Endliche Maschinen</p> <p>Automaten und formale Sprachen, Teil II: Kellerautomaten und kontextfreie Sprachen</p> <p>Kontextfreie Sprachen Nichtdeterministische Kellerautomaten Deterministische Kellerautomaten Berechenbarkeit, Entscheidbarkeit</p> <p>Komplexitätstheorie</p> <p>Grundlagen Nichtdeterministische Komplexität Die Komplexitätsklassen P und NP Zeit- und Platzhierarchien</p>
Voraussetzungen	Anwendungskennntnisse im Bereich der Linearen Algebra, Aussagenlogik und Booleschen Algebra
Modulbausteine	<p>FM101 Studienbrief Automaten und formale Sprachen, Teil I: Endliche Automaten und reguläre Sprachen mit Onlineübung</p> <p>FM102 Studienbrief Automaten und formale Sprachen, Teil II: Kellerautomaten und kontextfreie Sprachen mit Onlineübung</p> <p>FM104 Studienbrief Komplexitätstheorie mit Onlineübung</p> <p>Onlinetutorium</p>
Kompetenznachweis	Klausur (2 Stunden)



Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



INF20 Einführung in die Informatik

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Inhalte und Aufgaben der Informatik kennenlernen und verstehen; Aufbau und Arbeitsweise von Computersystemen kennen und verstehen; Grundbegriffe über Software und Programmierung wissen und anwenden können; Ansätze der Vernetzung von Rechnersystemen wissen und bewerten können (Fach- und Methodenkompetenz).</p> <p>Fachkenntnisse: Die Absolvierenden verfügen über breites und integrierendes Wissen der Informatik in der Informationsgesellschaft; sie verstehen die Wechselwirkungen des Fachgebietes in Bezug auf die historischen, sozialen, kulturellen, ökonomischen, politischen und ökologischen sowie ethischen Dimensionen von Gesellschaft.</p>
-----------------------	--

Inhalt	<p>Grundlagen der Informationsverarbeitung Einstieg ins Thema Was ist Informatik? Information und Daten: Was wird verarbeitet? Informationsdarstellung: Das Abbild der Welt im Computer Informationsverarbeitung: Der Computer als Werkzeug Computergestützte Parallelwelt</p> <p>Rechnerarchitektur Hardware-Grundlagen Rechnerarchitektur und Universalrechner Moderne Prozessor-Mikroarchitekturen Multicore- und Multiprozessorarchitekturen Speichersystem- und Kommunikationsstrukturen Alternative Rechnerarchitekturen für die Zukunft</p> <p>Software Software als Element der Wirtschaftsinformatik Die unterschiedlichen Softwarearten Der Weg zur Softwarelösung Entwurf einer Lösungsvorschrift: Algorithmen und Datenstrukturen Programmierung: Umsetzen der Lösung</p> <p>Computernetze und Internet Netzwerkgrundlagen Netzwerkarchitekturen Übertragungsmedien Ethernet Internetprotokoll (IP) Transmission Control Protocol (TCP)/User Datagram Protocol (UDP) Layer 2 - Konzepte und Geräte Layer 3 - Konzepte und Geräte Internetdienste Aspekte zukünftiger Netzplanung</p>
---------------	--



Informatik und Gesellschaft

Das Selbst- und Fremdbild von Informatikerinnen und Informatik
Informatik in der Informationsgesellschaft

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	WIN102 Studienbrief Grundlagen der Informationsverarbeitung mit Onlineübung WIN103 Studienbrief Rechnerarchitektur mit Onlineübung WIN104 Studienbrief Software mit Onlineübung WIN106 Studienbrief Computernetze und Internet mit Onlineübung WIN205 Research Guide
-----------------------	---

Kompetenznachweis	Klausur (120 Minuten)
--------------------------	-----------------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Andrea Herrmann
----------------------	-----------------



IUK20 Grundlagen zu Betriebssystemen und Netzwerken

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	Aufbau und Funktionsweise moderner Betriebssysteme erläutern; die Lösungsansätze in Betriebssystemen bewerten; die konkrete Realisierung in UNIX/Linux beschreiben. Die technischen Grundlagen und Mechanismen von Datenkommunikation und Computernetzwerken erläutern. Die Funktionsweise von Rechnernetzen im Allgemeinen und des Internets im Besonderen erläutern. Die Konzepte zur Netzwerksicherheit bewerten. Die Aufgaben und Hilfsmittel der Netzverwaltung beschreiben. (Fach- und Methodenkompetenz.)
-----------------------	--

Inhalt	<p>Betriebssysteme I: Architektur und Funktionsprinzipien Überblick und Einordnung Architektur von Betriebssystemen Prozesse Koordination paralleler (nebenläufiger) Prozesse Betriebsmittel (Ressourcen) Speicherverwaltung Ein-/Ausgabe-System Dateiverwaltung Praktischer Einsatz von Betriebssystemen</p> <p>Netzwerke I: Netzwerktechnik Grundlagen der Datenkommunikation Das OSI-Referenzmodell Aufbau und Funktionsweise von Netzwerken Netzverbund, Internetworking</p> <p>Netzwerke II: Internet-Technik Protokolle und Dienste (Einführung) Die Vermittlungsschicht: Internet Layer Protokolle der Transportschicht: Host-to-Host-Layer Der Netzzugang: Network Access Layer Die Anwendungsschicht: Application Layer</p> <p>Netzwerke III: Inhouse-Netzwerke LAN LAN-Basisverfahren und Standards Intranets und Extranets Das Arbeiten in LANs</p> <p>Netzverwaltung und Netzwerksicherheit Netzwerkmanagement Integrität, Funktionsfähigkeit und Auslastung des Netzes Benutzerverwaltung, Zugriffsrechte</p>
---------------	---



Anwendungsverwaltung
Netzwerkmanagement-Protokolle
Sicherheit im Netz
Kryptologie
Sicherheitsprotokolle

Voraussetzungen	Grundlagen der Wirtschaftsinformatik
------------------------	--------------------------------------

Modulbausteine	IUK101 Studienbrief Betriebssysteme I: Architektur und Funktionsprinzipien mit Onlineübung IUK103 Studienbrief Netzwerke I: Netzwerktechnik mit Onlineübung IUK104 Studienbrief Netzwerke II: Internet-Technik mit Onlineübung IUK105 Studienbrief Netzwerke III: Inhouse-Netzwerke mit Onlineübung IUK106 Studienbrief Netzverwaltung und Netzwerksicherheit mit Onlineübung Onlinetutorium (1 Stunde)
-----------------------	---

Kompetenznachweis	Klausur (2 Stunden)
--------------------------	---------------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Dr. Franz-Karl Schmatzer
----------------------	--------------------------



IUK21 Internet der Dinge und Embedded Systems

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Kenntnis der wesentlichen Grundlagen des Internets der Dinge (Internet of Things (IoT)). Gängige Kommunikationsstandards und Netzwerktopologien kennen. Kenntnis der wesentlichen Konzepte zu eingebetteten Systemen. Typische Anwendungen von Embedded Systems einordnen können. Überblick über Anwendungsgebiete vom Smart Home bis zur Smart Factory. Das wirtschaftliche Potential solcher Anwendungen einordnen können. Diskussion gesellschaftlicher, rechtlicher und sozialer Aspekte der aktuellen Entwicklungen.
Inhalt	Grundlagen für das Internet der Dinge Industrie 1.0 bis Industrie 4.0 Ursprünge und Entwicklung des Internets (Web 1.0 bis Web 4.0) Aspekte für Arbeitsmarkt und Weiterbildung Aspekt Datenschutz Netzwerktopologien und -protokolle SMART Anwendungen des IoT Einführung in die Smart Services Prinzipien des IoT und der Smart Services Der Faktor Mensch Umsetzung und Best Practices
Voraussetzungen	Keine.
Modulbausteine	Fachbuch Borgmeier: Smart Services und Internet der Dinge IUK201-Begleitheft zum Fachbuch IUK202 Studienbrief SMART Anwendungen des IoT mit Onlineübungen
Kompetenznachweis	Klausur (2 Stunden)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Günther Würtz



JAV41 Programmieren in Java 1

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Die Prinzipien der Programmierung in Java verstehen. Den Ansatz der Plattformunabhängigkeit (Java Virtual Machine) erläutern. Die grundlegenden Sprachelemente von Java kennen und anwenden. Einfache grafische Anwendungen und Programme mit Datenbankbindung erstellen und zum Ablauf bringen. Die Möglichkeiten von Java zur Programmierung von verteilten Anwendungen in Netzwerkumgebungen kennen. Mit einer Programm-Entwicklungsumgebung für Java umgehen. (Fach- und Methodenkompetenz.)
Inhalt	Programmieren in Java Objektorientierung und erstes Programmieren in Java Die Programmiersprache Java Grundlegende Java-Bibliotheken
Voraussetzungen	Grundlagen der Programmierung
Modulbausteine	ABTE003-EL Fachbuch Ratz/Scheffler/Seese/Wiesenberger: Grundkurs Programmieren in Java JAV101 Studienbrief Programmieren in Java mit Onlineübung
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



JAV60 JAVA-EE Entwicklung

Kompetenzzuordnung	Instrumentale Kompetenz
---------------------------	-------------------------

Kompetenzziele	Die Java EE-Architektur verstehen und anwenden; Servlets, JavaServer Pages und JavaServer Faces entwerfen, programmieren und das Deployment durchführen; Enterprise JavaBeans entwerfen, entwickeln und einsetzen; RESTful Webservices entwickeln und ein Deployment durchführen.
-----------------------	--

Inhalt	<p>Einführung in die Architektur von Java EE Schichtenmodelle für verteilte Systeme in Java Das Java EE Anwendungsmodell Die Java EE Container Die Web- und die Verarbeitungsschicht Komponenten von Java EE</p> <p>Servlets Webseiten erstellen Die Servlet-Architektur Deployment-Deskriptoren für Servlets</p> <p>JavaServer Pages Scriptelemente Direktiven Aktionen Verwendung von JavaBeans</p> <p>JavaServer Faces Das JSF-Framework Konfiguration von JSF Anwendungen</p> <p>Enterprise Javabeans Verteilte Systeme Aufbau von JavaBeans EJB Deployment Remote Zugriff Transaktionen</p> <p>RESTful Web-Services REST Prinzipien JAX-RS</p>
---------------	---

Voraussetzungen	Programmierkenntnisse in JAVA
------------------------	-------------------------------

Modulbausteine	JAV601 Studienbrief EE-Programmierkonzepte in Java mit Onlineübung JAV602 Studienbrief Software und Tools zur EE-Programmierung mit Onlineübung JAV603 Studienbrief Fallbeispiele zur EE-Programmierung
-----------------------	--



Fachbuch Müller-Hofmann; Hiller; Wanner: Programmierung von verteilten Systemen und Webanwendungen mit Java EE

JAV604-BH Begleitheft zum Fachbuch

Programmierlabor (1 Tag)

Kompetenznachweis	Assignment (Laborbericht)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



MAT29 Grundlagen der Mathematik und Lineare Algebra

Kompetenzzuordnung

Wissensverbreiterung

Kompetenzziele

Eigenschaften (Grenzwert, Bildungsgesetz) von Folgen und Reihen kennen und Grenzwerte berechnen; vollständige Induktion anwenden; mit unterschiedlichen Zahlensystemen umgehen;

lineare Gleichungssysteme lösen; die Vektorrechnung sicher einsetzen und anwenden; mathematische Lösungsverfahren anwenden und Lösungen darstellen; Ergebnisse und ihre Genauigkeit und Grenzen interpretieren; Grundlagen der komplexen Zahlen kennen und Rechenregeln sicher anwenden; Geraden und Ebenen in berufsspezifischer Darstellungsweise abbilden und typische Merkmale berechnen; die Fähigkeit entwickeln, ingenieurtechnische Probleme mit mathematischen Modellen zu beschreiben und zu lösen.

Inhalt**Zahlenmengen und Zahlensysteme**

Zahlenmengen

Summen und Produkte

Vollständige Induktion

Stellenwertsysteme

Teilbarkeit

Relationen und Funktionen

Relationen und Anwendungen

Funktionen

Folgen und Reihen

Der Begriff der Folgen und Reihen

Vollständige Induktion

Arithmetische Folgen und Reihen

Geometrische Folgen und Reihen

Grenzwerte von Folgen und Reihen

Vektoralgebra

Vektorbegriff und einfache Rechenarten

Skalarprodukt und Anwendungen

Vektorprodukt und Anwendungen

Linearkombination und lineare Unabhängigkeit

Analytische Geometrie

Vektorielle Darstellung einer Geraden

Vektorielle Darstellung einer Ebene

Lineare Algebra

Matrixalgebra

Lineare Gleichungssysteme (LGS) und ihre Lösung

Der lineare Raum

Vom Gleichungssystem zur Determinante

**Voraussetzungen**

Mathematische Grundlagen

Modulbausteine

WM107 Studienbrief Folgen und Reihen mit **Einsendeaufgabe**
IMA201 Studienbrief Vektoralgebra
IMA203 Studienbrief Analytische Geometrie
WM111 Studienbrief Lineare Algebra mit **Einsendeaufgaben**
Fachbuch Teschl: Mathematik für Informatiker – Band 1: Diskrete
Mathematik und lineare Algebra
2 Präsenztutorien (je 1 Tag)

Kompetenznachweis

Klausur (120 Minuten)

Lernaufwand

125 Stunden, 5 Leistungspunkte

Sprache

Deutsch

Studienleiter

Dr. Rainer Berkemer



MCS41 Microcomputer-Systeme mit Labor

Kompetenzzuordnung	Instrumentale Kompetenz
---------------------------	-------------------------

Kompetenzziele	Architektur, Funktionsweise und Programmierung von Mikrocomputern sowie Grundlagen eingebetteter Systeme (Embedded Systems) kennen; Methoden und Werkzeuge für Softwareentwurf beherrschen; einfache Aufgaben zur Ansteuerung von Peripherie und zur Messwerterfassung mithilfe von Mikrocomputern lösen; einen handelsüblichen Mikrocontroller im Detail kennen. (Fach- und Methodenkompetenz.)
-----------------------	--

Inhalt	<p>Grundlagen der Mikrocomputersysteme Grundbegriffe Rechnerarchitektur Darstellung von Zahlen und Zeichen im Mikrocomputer Innerer Aufbau eines Mikrocomputers Speicher und Ein-/Ausgabe</p> <p>Mikrocontroller und Schnittstellen Typische Mikrocontroller Timer und Wandler Chipschnittstellen Standardschnittstellen Digitale Interface-Schaltungen Anzeigen und Displays</p> <p>Programmierung von Mikrocomputersystemen Programmentwicklung – Vom Problem zur Lösung Programmierung in Assembler Den Mikrocontroller in C programmieren</p> <p>Anwendungen von Mikrocomputersystemen Vertiefende Assemblerprogrammierung mit dem 68HC11 Arduino Statemachine Motorsteuerung Analoge Daten verarbeiten Datenauswertung</p> <p>Mikrocomputerpraktikum mit dem Arduino Die Arduino-Entwicklungssoftware Das Arbeitsbuch "Die elektronische Welt mit Arduino entdecken" Ablauf des Labors zu MCS41</p>
---------------	---

Voraussetzungen	Kenntnisse der Digitaltechnik und im Programmieren in C; Grundlagen der Elektronik
------------------------	---



Modulbausteine	ABTE010-EL Fachbuch Brühlmann: Arduino Praxiseinstieg Bausatz mit Arduino Mikrocontroller und Zubehör mit Software (Entwicklungsumgebung Arduino) MCS401-BH Begleitheft zum ABTE022-EL Fachbuch Bartmann: Mit Arduino die elektronische Welt entdecken ABTE079-EL Fachbuch Bernstein: Microcontroller Labor (2 x 1 Tag im Abstand von ca. 5 - 6 Wochen; 1. Tag: Inbetriebnahme und erste Übungen; 2. Tag: praktische Übungen mit einem Mikrocontroller)
Kompetenznachweis	Assignment (Laborbericht)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



MCS60 Projekt Mikrocomputer Programmierung und IOT

Kompetenzzuordnung	Instrumentale Kompetenz
Kompetenzziele	Embedded Systeme auf IOT-Systeme anwenden; typische Aufbauformen und verwendete Controller von Embedded Systemen kennen sowie Einsatzbereiche beurteilen; Embedded Systeme hard- und softwaremäßig entwerfen, aufbauen und programmieren; Integration von Sensoren und Aktoren in Embedded Systeme verstehen und Anbinden an ein Netzwerk und die Cloud.
Inhalt	Einführung Tastatureingabe, prellen und entprellen Laborübung Tastatureingabe Aufbau einer Matrix-Tastatur Laborübung digitale Sensoren mit 1-Wire Anbindung Laborübung I2C-Bus (2-Wire) Anbindung Laborübung digitale Port-Erweiterung durch einen Port Expander Laborübung SPI-Bus (2-Wire) Anbindung Laborübung Bluetooth Anbindung Laborübung WiFi Anbindung Laborübung Cloud-Anbindung und Verwalten der Daten in der Cloud
Voraussetzungen	Kenntnisse der Digitaltechnik Kenntnisse im Programmieren mit C Grundlagen der Elektronik Programmiererfahrung mit dem Mikrocontroller Arduino
Modulbausteine	Bausatz II mit Arduino Mikrocontroller und Zubehör mit Software (Entwicklungsumgebung Arduino) MCS601 SB Praktikum und erweiterte Übungsaufgaben mit dem Arduino Labor (1 Tag)
Kompetenznachweis	Assignment (Laborbericht)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



PRG23 Grundlagen der Programmierung und Digitaltechnik

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Die Grundbegriffe und grundlegenden Ansätze der Programmierung definieren bzw. beschreiben; die wesentlichen Kontrollstrukturen in Programmiersprachen beschreiben; die grundlegenden Datentypen und -strukturen und ihre Abbildung in Computern erläutern; die Komponenten zur Programmentwicklung abgrenzen (am Beispiel C/C++); logische Funktionen und wesentliche Eigenschaften digitaler Schaltkreisfamilien, sowie Typen und Struktur von Halbleiterspeichern kennen und verstehen; digitale Schaltungen miteinander kombinieren, programmierbare Logik kennen; Grundlagen des Programmierens von Logikbausteinen kennen und anwenden; einfache Steuerungen anhand von ausgewählten Anwendungen entwerfen und simulieren; Grundlagen von Mikrocontrollern und SPS verstehen.</p>
-----------------------	--

Inhalt	<p>Grundlagen der Programmierung Informationen und Daten Verarbeitung von Daten in Rechnern Programmiersprachen Datentypen und Datenstrukturen Programmierung im Kleinen Programmierung im Großen Ein- und Ausgabe in Programmen Softwareentwicklung Zahlensysteme und Codes Geschichte der Digitaltechnik Signale und Nachricht Zahlensysteme Fest- und Gleitkommadarstellung Informationstheorie Codes Numerische und alphanumerische Codes Gesicherte Codes und Codeeffizienz Boolesche Algebra und kombinatorische Schaltkreise Boolesche Logik Grundlagen der Aussagenlogik Optimierung von Logikfunktionen Kombinatorische Schaltkreise Rechenschaltungen Sequenzielle Schaltungen, Schaltwerke und Simulationssoftware</p>
---------------	---



Automatentheorie
Flipflop
Realisierung eines synchronen Automaten
Register und Zähler
Ein einfacher Rechner
Programmierbare Logikhardware

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	PRG101 Studienbrief Grundlagen der Programmierung mit Onlineübung ELT301 Studienbrief Zahlensysteme und Codes mit Onlineübung ELT302 Studienbrief Boolesche Algebra und kombinatorische Schaltkreise mit Onlineübung ELT303 Studienbrief Sequenzielle Schaltungen, Schaltwerke und Simulationssoftware mit Onlineübung Onlinetutorium (1 Stunde)
-----------------------	---

Kompetenznachweis	Klausur (120 Minuten)
--------------------------	-----------------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Matthias Riege
----------------------	----------------



PRG24 Programmierparadigmen

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Die Grundbegriffe und grundlegenden Ansätze der Programmierung kennen; die wesentlichen Kontrollstrukturen in Programmiersprachen beschreiben und anwenden; die grundlegenden Datentypen und -strukturen und ihre Abbildung in Computern kennen; die wesentlichen Programmiermodelle kennen und verstehen (am Beispiel von Racket; Fachkompetenz).
Inhalt	Einführung in die Programmierung mit Racket Variablen Datentypen Prozeduren Benannte Prozeduren Bedingte Ausdrücke Rekursive Prozeduren Funktionsorientierte Programmierung Charakteristik Werte erster Klasse Transformation echt rekursiver Prozeduren in endständige Evaluation von Ausdrücken Der Lambda-Kalkül Umgebungsmodell Objektorientierte Programmierung Konkrete und abstrakte Datentypen Das 4-Stufen-Modell Generische Operationen Objekte, Zustand und Methode Klassenvariablen Vererbung und Polymorphie Imperative Programmierung Wertzweisung, Befehle, Sprünge und Zyklen L-Wert und R-Wert Identität und Gleichheit Parametervermittlung
Voraussetzungen	Grundlagenkenntnisse der Informatik
Modulbausteine	Fachbuch Wagenknecht: Programmierparadigmen Labor (1 Tag, praktische Übung)
Kompetenznachweis	Assignment (Laborbericht)
Lernaufwand	125 Stunden, 5 Leistungspunkte



Sprache Deutsch

Studienleiter Andrea Herrmann



PWS40 Projektwerkstatt

Kompetenzzuordnung	Wissensvertiefung
Kompetenzziele	Aufgabenstellungen mit einem wissenschaftlichen Anspruch auf Bachelorniveau und im Kontext der Themenfelder und Schwerpunkte des Studiengangs problem- und zielorientiert im Team und nach den Methoden eines modernen Projektmanagements bearbeiten und lösen; das erworbene – auch interdisziplinäre – Fachwissen umsetzen und anwenden; über die Fähigkeit verfügen, geeignete Werkzeuge der Kooperation und Kommunikation einzusetzen; Ergebnisse zielorientiert und nach den Regeln der Wissenschaftlichkeit dokumentieren und präsentieren.
Inhalt	Bearbeitung einer Projektaufgabe Selbstständig sowie in Gruppen unter Verwendung verschiedener Methoden und Diskurse; Beispiele: Modell- oder Konzeptentwicklung, Optimierungsempfehlungen, Untersuchungen, empirische Forschungsarbeit, Gestaltungsempfehlungen usw. Gegenstand der Projektarbeiten: Analyse, Planung, Konzeption, Gestaltung, Entwicklung, Einsatz und Bewertung von Lösungen für den Praxiseinsatz unter Berücksichtigung der Kompetenzfelder der Studiengangsschwerpunkte.
Voraussetzungen	Keine.
Modulbausteine	Keine.
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Ulrich Kreutle



SQF24 Schlüsselqualifikationen für Studium und Beruf

Kompetenzzuordnung	Systemische Kompetenz
Kompetenzziele	<p>Grundlagenkenntnisse: die eigene Persönlichkeit und den eigenen Arbeitsstil einschätzen und Ansätze zu deren Verbesserung finden. Arbeits- und Kreativitätstechniken beschreiben und einfache Techniken anwenden.</p> <p>Moderne Methoden des Zeitmanagements anwenden.</p> <p>Grundlagenkenntnisse: Präsentationen didaktisch-methodisch planen, organisatorisch vorbereiten, selbst durchführen und nachbereiten können.</p> <p>Präsentationen beurteilen und Verbesserungsansätze für Rhetorik und Körpersprache erkennen (Methoden-, Medien-, persönliche, kommunikative, soziale Kompetenz).</p> <p>Anforderungen an wissenschaftliche Einsendeaufgaben, Referate und Abschlussarbeiten beschreiben und erläutern Möglichkeiten der wissenschaftlichen Recherche beschreiben und unterscheiden</p> <p>Korrekt zitieren (Methodenkompetenz)</p>
Inhalt	<p>Selbstmanagement</p> <p>Die Vielfalt des Lebens</p> <p>Lebenshaltungen</p> <p>Ziele</p> <p>Entscheidungs- und Handlungskompetenz</p> <p>Ziel- und Zeitmanagement</p> <p>Zeit braucht Ziele</p> <p>Methoden des Ziel- und Zeitmanagements</p> <p>Instrumente des Ziel- und Zeitmanagements</p> <p>Kreative Kompetenz</p> <p>Was ist kreative Kompetenz?</p> <p>Einflüsse auf die Kreativität</p> <p>Techniken der Kreativität</p> <p>Vom Lesen zum Schreiben</p> <p>Zielsicher Präsentieren</p> <p>Ist Präsentieren schwierig?</p> <p>Wege zu einer guten Präsentation</p> <p>Medieneinsatz</p> <p>Wissenschaftliches Arbeiten</p> <p>Wissenschaftliche Vorarbeit</p> <p>Wissenschaftliche Hauptarbeit</p> <p>Wissenschaftliche Nacharbeit</p>
Voraussetzungen	Keine.
Modulbausteine	<p>Orientierungswerkstatt (1 Tag + 2 x 0,5 Tage Präsenzseminar + 2 Stunden Onlineseminar)</p> <p>SQF232 Studienbrief Selbstmanagement</p>



SQF233 Studienbrief Ziel- und Zeitmanagement

SQF234 Studienbrief Kreative Kompetenz

SQF235 Studienbrief Zielsicher Präsentieren

SQL301 Studienbrief Wissenschaftliches Arbeiten mit **Onlineübung**

SQLD302-VH Download Vorgaben für wissenschaftliche Studien- und Abschlussarbeiten bei AKAD

Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Prof. Dr. Marianne Blumentritt



SQF42 Klassisches und agiles Projektmanagement

Kompetenzzuordnung	Instrumentale Kompetenz
---------------------------	-------------------------

Kompetenzziele	<p>Einsatzmöglichkeiten und Elemente der Projektorganisation kennen; wissen, wie Projekte initialisiert werden (Analyse des Projektumfeldes und der Stakeholder), und wie sich Ziele, Anforderungen und Erfolgsfaktoren definieren lassen;</p> <p>das Projekt strukturieren, den Aufwand schätzen und die Mittel planen können;</p> <p>Einblick in die Führungsaufgaben innerhalb von Projekten erhalten;</p> <p>Bedeutung von Kommunikation, Teamentwicklungsprozessen und Konfliktmanagement erkennen;</p> <p>begleitende Aufgaben wie Projektmarketing, Changemanagement, Konfigurationsmanagement, QM erläutern und einschätzen können;</p> <p>Erkennen und Überwinden von Widerständen. Agile Projektmanagement-Methoden kennen, ihre Vor- und Nachteile bewerten und ihre Einsatzmöglichkeiten erläutern können. (Fach- und Methodenkompetenz.)</p>
-----------------------	---

Inhalt	<p>Projektaufbau, Funktionen und Managementtechniken</p> <p>Begriffe Projektaufbau Funktionen im Projekt Managementtechniken</p> <p>Projekte initialisieren und planen</p> <p>Projekte initialisieren Projekte planen</p> <p>Projekte abwickeln und abschließen</p> <p>Projekte leiten und steuern Risikomanagement Problemmanagement Projektberichte Projektabschluss Projektsitzungen und Workshops</p> <p>Führen in Projekten und begleitende Aufgaben</p> <p>Die Projektführung Das Projektteam Kommunikation Widerstand Konflikte Projektmarketing Änderungs- und Konfigurationsmanagement Qualität im Projekt Lieferantenmanagement</p> <p>Multiprojektmanagement</p> <p>Multiprojektmanagement: Stellenwert und Standort Multiprojektmanagement-Prozess Multiprojektmanagement-Methoden</p>
---------------	---



Multiprojektmanagement-Organisation
 Multiprojektmanagement-Qualifikation
 Implementierung des Multiprojektmanagements
 Multiprojektmanagement-Organisation
Historische Entwicklung der Vorgehensmodelle
 Spezifikationsorientierte Entwicklung kompletter Systeme
 Prototyorientierte Entwicklung kompletter Systeme
 Spiralmodell für komplette Systeme
 Agile, inkrementelle Softwareentwicklung
 Fortschritte durch die verschiedenen Vorgehensmodelle
 Auswahl eines Vorgehensmodells
Das agile Rahmenwerk Scrum
 Historie von Scrum
 Charakteristika von Scrum
 Übersicht über den Scrum-Prozess
 Rollen in Scrum Teams
 Projektumsetzung mit Scrum
 Vor- und Nachteile von Scrum
 Hybride Verwendung von Scrum
 Unterschiede zwischen Scrum und Extreme Programming
Die Change Management-Methode von Kanban
 Historie von Kanban
 Begriffswelt der Kanban-Methode
 Vergleich von Kanban mit Scrum

Voraussetzungen	Keine.
Modulbausteine	SQF201 Studienbrief Projektaufbau, Funktionen und Managementtechniken mit Onlineübungen SQF401 Studienbrief Projekte initialisieren und planen mit Onlineübungen SQF402 Studienbrief Projekte abwickeln und abschließen mit Onlineübungen SQF403 Studienbrief Führen in Projekten und begleitende Aufgaben mit Onlineübungen SQF404 Studienbrief Multiprojektmanagement E-Book Goll/Hommel: Mit Scrum zum gewünschten System SQF405-BH Begleitheft mit Online-Übung
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Ulrich Kreutle



SWE40 Design Pattern

Kompetenzzuordnung	Wissensvertiefung
Kompetenzziele	Grundkenntnisse in Softwareentwicklung; Grundsätze und Prinzipien der Softwareentwicklung sowie Vorgehensweisen beschreiben; Design-Muster als wesentliche Bestandteile der Softwareentwicklung kennenlernen und bewerten können.
Inhalt	Einführung in die Systementwicklung Einführung: Softwareentwicklung als Problem Grundlegende Entwicklungsstrategien und Prinzipien Vorgehensmodelle: Softwareentwicklung als Prozess Die Phasen der Softwareentwicklung Phasenunabhängige Aufgaben Objektorientierte Softwareentwicklung Agile Softwareentwicklung Objektorientierte Softwareentwicklung Objektorientierung Objektorientierte Modellierung: UML Objektorientierter Entwicklungsprozess Komponentenbasierte Softwareentwicklung Serviceorientierte Softwareentwicklung Werkzeuge und Entwicklungsumgebungen Serviceorientierte Architektur (SOA) Design Pattern in der Softwareentwicklung Singleton Pattern Observer Pattern Mediator Pattern Chain of Responsibility State Pattern Command Pattern Strategy Pattern Iterator Pattern
Voraussetzungen	Programmierkenntnisse in Java
Modulbausteine	SWE101 Studienbrief Einführung in die Systementwicklung mit Onlineübung SWE204 Studienbrief Objektorientierte Softwareentwicklung mit Onlineübung SWE401 Studienbrief wichtige Design Pattern mit Onlineübung Fachbuch Siebler: Design Patterns mit Java: Eine Einführung in Entwurfsmuster



Kompetenznachweis Klausur (2 Stunden)

Lernaufwand 125 Stunden, 5 Leistungspunkte

Sprache Deutsch

Studienleiter Andrea Herrmann



SWE41 Software Architekturen

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Kennenlernen von Softwarearchitekturen, die für den Entwurf verschiedener Softwaresysteme genutzt werden; Analyse von Softwarearchitekturen; Bewerten und Beurteilen von Softwarearchitekturen; Kennenlernen verschiedener Softwarearchitekturen.
Inhalt	Einführung in Software-Architekturen Die Bedeutung von Softwarearchitekturen Softwarearchitekturen im Web-Umfeld Softwarearchitekturen im Desktop-Umfeld Softwarearchitekturen im Server-Umfeld Referenzarchitekturen Typen von Referenzarchitekturen Beschreiben von Referenzarchitekturen Nutzen von Referenzarchitekturen Pflege und Wartung Analyse Ziele der Analyse Umfang der Analyse Analysearten Analysetechniken Beispiele von Architekturmustern Multimedia-Architekturen Peer-to-Peer-Architekturen Grid-Architekturen Serviceorientierte Architekturen
Voraussetzungen	Keine.
Modulbausteine	SWE402 Studienbrief Einführung in Software-Architekturen mit Onlineübung SWE403 Studienbrief Referenzarchitekturen mit Onlineübung SWE404 Studienbrief Analyse von Software-Architekturen mit Onlineübung SWE405-RGResearch-Guide
Kompetenznachweis	Klausur (120 Minuten)
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch



Studienleiter

Andrea Herrmann



SWE63 Software Qualitätsmanagement

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	Techniken, Methoden und Strategien zur Umsetzung des Software-Qualitätsmanagements kennen und anwenden; Maßnahmen zur Qualitätsverbesserung gezielt aufgabenorientiert auswählen und umsetzen; Werkzeuge zur Qualitätssicherung kennenlernen; Qualitätssicherung durch Unit-Tests in Java.
-----------------------	---

Inhalt	Methoden im Software Qualitätsmanagement Geschichte des Qualitätswesens Qualitätsmerkmale Messen der Qualität durch Metriken William Edward Deming und seine Qualitätsphilosophie Ausgewählte Methoden und Werkzeuge des Qualitätsmanagements Statistische Methoden im Qualitätsmanagement Statistische Grundlagen Datensammlung im Qualitätswesen Verteilungen und Vertrauensbereiche Wichtige Verteilungsformen und deren Regelkarten Test auf Normalverteilung Fähigkeitsbetrachtung Stichproben Software-Qualitätssicherung und Software-Wirtschaft Softwarequalität Qualitätssicherung und Qualitätsmanagement Produktorientiertes Qualitätsmanagement Prozessorientiertes Qualitätsmanagement Wirtschaftlichkeit der Softwareentwicklung Software-Qualitätssicherungs-Werkzeuge Qualitätssicherung Testfälle Testebenen Testautomation Testarchitektur und Mocking Manuelle Mock-Erstellung Mock-Werkzeuge Test von Nutzeroberflächen GUI-Steuerung Capture & Replay Bildererkennung
---------------	---

Voraussetzungen	Kenntnisse im Bereich der Softwareentwicklung
------------------------	---



Modulbausteine

SWE206 Studienbrief Software-Qualitätssicherung und Software-Wirtschaft mit **Onlineübung**

QUM601 Studienbrief Qualitätsphilosophien und Methoden im Software Qualitätsmanagement mit **Onlineübung**

QUM102 Studienbrief Statistische Methoden im Qualitätsmanagement mit **Onlineübung**

Fachbuch Kleuker: Qualitätssicherung durch Softwaretests – Vorgehensweisen und Werkzeuge zum Testen von Java-Programmen

SWE601-BH Begleitheft zum Fachbuch

Kompetenznachweis	Klausur (120 Minuten)
--------------------------	-----------------------

Lernaufwand	125 Stunden, 5 Leistungspunkte
--------------------	--------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Andrea Herrmann
----------------------	-----------------



SWE64 Softwareentwicklung

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Grundkenntnisse in Softwareentwicklung; Grundsätze und Prinzipien der Softwareentwicklung sowie Vorgehensweisen beschreiben; Bedeutung, Inhalt und Ergebnisse der Phasen der Softwareentwicklung erläutern; grundlegende Hilfsmittel und Methoden nennen; Erkennen, wie sich große Systeme in Komponenten zerlegen lassen; die Schritte im komponentenbasierten Entwicklungsprozess erläutern; Komponenten entwerfen und beschreiben; die Rolle von Softwarearchitekturen bei der Bewältigung komplexer Entwurfs- und Entwicklungsvorhaben verstehen; erkennen, was Softwarearchitektur leisten kann; Softwarearchitekturen darstellen und bewerten; Softwarearchitekturen praktisch entwerfen; beurteilen, welche Ansätze, Software ingenieurmäßig zu entwickeln, für welche Einsatzzwecke geeignet sind; dabei die Ansätze der Wiederverwendung erläutern; die Web-Services-Standards kennen und wissen, wie diese die organisationsübergreifende Verarbeitung unterstützen können; Geschäftsprozessmodelle als Workflows darstellen und für den Entwurf serviceorientierter Systeme einsetzen.</p>
-----------------------	--

Inhalt	<p>Einführung in die Softwareentwicklung Einführung und Überblick Prinzipien der Softwareentwicklung Allgemeine Vorgehensweisen Vorgehensmodelle Software-Management Software-Management Projektmanagement Vorstudie und Lastenheft Produktivität und Aufwandsschätzung Innovations- und Risikomanagement Software-Ergonomie und Interaktionsdesign mit Internet Grundlagen der Mensch-Computer-Kommunikation (MCK) Benutzer- und Anwendungsklassen Allgemeine Grundsätze der Softwareergonomie Gestaltungskriterien für Computer-Arbeitsplätze Entwicklung von Dialogschnittstellen Benutzerunterstützung Interaktionsdesign im Internet Entwicklung der Architekturen arbeitsteiliger Softwareentwicklung Einführung: Architekturen, Anwendungslandschaft, Entwurf, Anwendungssoftware, Softwarearchitekturen Architekturbeschreibungen Architekturbeschreibungen mit der UML</p>
---------------	---



Architektur- und Komponentenentwicklung
Ausgewählte arbeitsteilige Architekturen
Wiederverwendung
Bewertung von Softwarearchitekturen
Trends, zukünftige Entwicklungen

Voraussetzungen	Keine.
------------------------	--------

Modulbausteine	SWE201 Studienbrief Einführung in die Softwareentwicklung mit Onlineübung SWE202 Studienbrief Softwaremanagement mit Onlineübung SWE205 Studienbrief Software-Ergonomie und Interaktionsdesign im Internet mit Onlineübung AB83-383 Fachbuch Reussner; Hasselbring (Hrsg.): Handbuch der Software-Architektur ABTE011-EL Fachbuch Staud: Unternehmensmodellierung – objektorientierte Theorie und Praxis mit UML 2.0 AB84-384 Fachbuch Dostal (u. a.): Service-orientierte Architekturen mit Web Services – Konzepte – Standards – Praxis MIP301-RG Research Guide
-----------------------	--

Kompetenznachweis	Assignment
--------------------------	------------

Lernaufwand	250 Stunden, 10 Leistungspunkte
--------------------	---------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Andrea Herrmann
----------------------	-----------------



VS42 Verteilte Systeme mit Labor, Verteilte Programmierung

Kompetenzzuordnung	Wissensvertiefung
---------------------------	-------------------

Kompetenzziele	<p>Charakteristische Eigenschaften verteilter Systeme benennen; verschiedene Systemmodelle einordnen (Client-/Server-, Multitier- Architektur, Mobiler Code, Mobile Agenten); die Grundlagen von Kommunikationsmechanismen und Middleware in verteilten Systemen beherrschen (Sockets, RPC, RMI, CORBA); die Ansätze für den Entwurf und die Realisierung von verteilten Anwendungen beurteilen (verteilte Objekte, Web-Services); die Herausforderungen und Lösungsmöglichkeiten für die verteilte Dateihaltung (verteilte Dateisysteme, Namensdienste, Datenbanken, Transaktionen) beschreiben.</p> <p>Konzepte und Programmiermodelle für Multicore-Systeme kennenlernen und mit dem Threadmodell von Java umsetzen; verschiedene Synchronisierungsmechanismen kennenlernen und einsetzen; die Programmierung von verteilten Systemen mit Sockets und RMI kennenlernen und umsetzen.</p>
-----------------------	---

Inhalt	<p>Grundlagen verteilter Systeme Hin zu verteilten Systemen Eigenschaften verteilter Systeme Architekturmodelle verteilter Systeme Interprozesskommunikation in verteilten Systemen Technologieplattformen für verteilte Systeme Cluster-Systeme</p> <p>Entwicklung verteilter Anwendungen Objektorientierte Programmierung und verteilte Systeme Technologien in verteilten Systemen REST-konforme Architektur Technologieplattformen für verteilte Systeme</p> <p>Verteilte Dateihaltung RAID-Konzepte Organisationsformen für Speichersysteme Verteilte Dateisysteme Cluster-Dateisysteme Verteilte Dateihaltung im Internet WEB 2.0 Ansätze für verteilte Systeme Zwischenbilanz Verteilte Dateihaltung in verteilten Applikationen Verteilte Dateihaltung in der Java-Technologie Namens- und Verzeichnisdienste</p> <p>Das Thread-Konzept von Java Main-Thread Erzeugen und Starten von Threads Lebenszyklus von Threads Exception-Handler</p>
---------------	--



Konkurrierende Zugriffe auf Daten
Synchronisierungskonzepte in Java
Semaphore
Message Queues
Pipes
Verklemmung
Verteilte Anwendung mit Sockets
Socket Schnittstelle
Kommunikation über UDP
Kommunikation über TCP
Sequentielle und parallele Server
Verteilte Anwendung mit RMI
Prinzip von RMI
Parallelität bei RMI-Methodenaufrufen
Kommunikation über TCP
Sequenzielle und parallele Server

Voraussetzungen	Einführung Informatik Kenntnisse in der Java Programmierung
------------------------	--

Modulbausteine	VS101 Studienbrief Grundlagen verteilter Systeme mit Onlineübung VS102 Studienbrief Entwicklung verteilter Anwendungen mit Onlineübung VS103 Studienbrief Verteilte Datenhaltung mit Onlineübung Fachbuch Oechsle: Parallele und verteilte Anwendungen in Java VS402-BH Begleitheft zum Lehrbuch mit Onlineübung Programmierlabor (2 Tage)
-----------------------	---

Kompetenznachweis	Assignment (Laborbericht)
--------------------------	---------------------------

Lernaufwand	250 Stunden, 10 Leistungspunkte
--------------------	---------------------------------

Sprache	Deutsch
----------------	---------

Studienleiter	Mark Harwardt
----------------------	---------------



WEB42 Webprojekt I

Kompetenzzuordnung	Wissensverbreiterung
---------------------------	----------------------

Kompetenzziele	<p>Die grundlegenden Konzepte und Sprachen zur Internet-Programmierung beschreiben;</p> <p>die wesentlichen Sprachkonstrukte von HTML und CSS kennen und einfache Internetauftritte (Webseiten) realisieren;</p> <p>die grundlegenden Eigenschaften von JavaScript und XML beschreiben;</p> <p>PHP als serverseitige Programmiersprache grundlegend beherrschen und kleinere Logiken abbilden können;</p> <p>Grundlagen zum Aufbau und zur Funktion von Datenbanken kennen;</p> <p>Eigenschaften eines Datenbankmanagementsystems (DBMS) kennen; ein DBMS einsetzen und anwenden.</p> <p>(Methoden-, Sozial- und Medienkompetenz)</p>
-----------------------	---

Inhalt	<p>Einführung in die Internet-Programmierung</p> <p>Internet und Browser</p> <p>Erstellen von Webseiten</p> <p>HTML</p> <p>CSS Style-Sheets</p> <p>JavaScript</p> <p>Grundlagen der CGI-Programmierung</p> <p>XML – Extensible Markup Language</p> <p>Grundlagen der Programmierung mit PHP</p> <p>Voraussetzungen</p> <p>PHP-Grundlagen</p> <p>Formularelemente</p> <p>Weitere Funktionen von PHP</p> <p>Einführung in die Datenbank-Programmierung</p> <p>MySQL und PHP</p> <p>Einführung und Bedeutung, Einsatzgebiete und Merkmale von MySQL und PHP</p> <p>MySQL</p> <p>Arbeiten mit MySQL</p>
---------------	--

Voraussetzungen	Grundkenntnisse zu Datenbanken
------------------------	--------------------------------

Modulbausteine	<p>INT102 Studienbrief Einführung in die Internet-Programmierung mit Onlineübung</p> <p>WEB602 Studienbrief Grundlagen der Programmierung mit PHP</p> <p>DABA201 Studienbrief Einführung in die Datenbank-Programmierung</p> <p>Labor (1 Tag, praktische Übung)</p>
-----------------------	--

Kompetenznachweis	Assignment (Gruppenarbeit/Laborbericht)
--------------------------	---



Lernaufwand 125 Stunden, 5 Leistungspunkte

Sprache Deutsch

Studienleiter Dr. Franz-Karl Schmatzer



WEB43 Webprojekt II

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	Eigenschaften eines Datenbankmanagementsystems (DBMS) kennen; ein DBMS einsetzen und anwenden; eine konkrete betriebliche Aufgabenstellung zum Aufbau einer vollständigen Datenbank mithilfe des DBMS lösen.
Inhalt	Fallstudie Fallsituation Aufgabenbeschreibung Beispielhafte Lösung zu der Fallstudie
Voraussetzungen	Webprojekt I
Modulbausteine	WEB603-FS Fallstudie
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer

WEB70 Navigationskonzepte

Kompetenzzuordnung	Wissensverbreiterung
Kompetenzziele	<p>Aufgabenbereiche zur Konzeption und Planung einer Web-Business-Lösung kennen;</p> <p>Bedeutung der Oberflächengestaltung für den Erfolg von Web-Anwendungen erkennen;</p> <p>Navigationskonzepte für Web-Anwendungen systematisieren; ausgehend von den Anforderungen von Anwendergruppen und den funktionalen Anforderungen an Web-Anwendungen geeignete Navigationskonzepte auswählen;</p> <p>Technologie-Konzepte zur Realisierung von Navigationskonzepten erläutern und bewerten können. (Fach- und Methodenkompetenz.)</p>
Inhalt	<p>Konzeption und Design von Web-Business-Lösungen</p> <p>Beurteilungskriterien für Web-Business-Lösungen</p> <p>Systemkonzeption</p> <p>Systemrealisierung und laufender Betrieb</p> <p>Systemoptimierung</p> <p>Wahrnehmung und Akzeptanz von Applikationen</p> <p>Begriffsbestimmung und Überblickswissen zur Usability</p> <p>Differenzierung von Anwendungssystemen</p> <p>Prinzipien des Responsiven Designs</p> <p>Erfolgsfaktoren von Web-Applikationen</p> <p>Interaktions- und Navigations-Konzepte</p> <p>Vorstellung und Bewertung von Interaktions- und Navigationskonzepten</p> <p>Empfehlungen für den Einsatz ausgewählter Interaktions- und Navigationskonzepte – abgestimmt auf die jeweiligen Anwendungsszenarien</p> <p>Technologie-Konzepte</p> <p>Darstellung und Bewertung etablierter Technologien zur Umsetzung von Interaktions- und Navigations-Konzepten</p> <p>Vorstellung und Bewertung ausgewählter Frameworks</p> <p>Umsetzungs-Konzepte</p> <p>Arbeitsschritte der Web-Engineering-Konzepte</p> <p>Agile Vorgehensmodelle</p>
Voraussetzungen	Kenntnisse der Bewertung webbasierter Front-Ends
Modulbausteine	<p>WEB102 Studienbrief Konzeption und Design von Web-Business-Lösungen mit Onlineübung</p> <p>Fachbuch Richter, Michael; Flückiger, Markus: Usability und UX kompakt. Produkte für Menschen</p> <p>Fachbuch Stapelkamp, Torsten: Informationsvisualisierung</p> <p>Fachbuch Thesmann, Stephan: Interface Design: Usability, User Experience und Accessibility im Web gestalten</p> <p>Fachbuch Weichert, Steffen; Quint, Gesine; Bartel, Torsten: Quick Guide UX Management</p>



Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer



WEB71 Interface Qualität

Kompetenzzuordnung	Instrumentale Kompetenz
Kompetenzziele	<p>Allgemeine Grundsätze der Softwareergonomie kennen und bei der Entwicklung von Web-Anwendungen konzeptionell berücksichtigen können;</p> <p>Dialogschnittstellen in Abhängigkeit der Nutzergruppen und der Anwendungszwecke entwickeln können.</p> <p>Die spezifischen Anforderungen von Kindern an Softwareprogramme und insbesondere an Benutzeroberflächen erläutern und für eine selbstgewählte Aufgabenstellung ein Oberflächenkonzept für eine Web-Anwendung für Kinder entwerfen können.</p> <p>(Fach-, Methoden- und konzeptionelle Kompetenz.)</p>
Inhalt	<p>Software-Ergonomie und Interaktionsdesign</p> <p>Mensch-Computer-Kommunikation</p> <p>Benutzer- und Anwendungsklassen</p> <p>Allgemeine Grundsätze der Softwareergonomie</p> <p>Gestaltungskriterien für Computer-Arbeitsplätze</p> <p>Entwicklung von Dialogschnittstellen</p> <p>Benutzerunterstützung</p> <p>Interaktionsdesign im Internet</p> <p>Analyse der Anwendergruppe "Kind"</p> <p>Theorien zur menschlichen Entwicklung</p> <p>Verlauf der menschlichen Entwicklung</p> <p>Kognitive Entwicklung</p> <p>Körperliche und motorische Entwicklung</p> <p>Emotionale und soziale Entwicklung</p> <p>Computernutzung und -erfahrung</p> <p>Software für Kinder</p> <p>Arten von Kindersoftware</p> <p>Qualitätsmerkmale guter Kindersoftware</p> <p>Methode der Softwareentwicklung für Kinder</p> <p>Child-Centered Design</p> <p>Usability Engineering Lifecycle für Kindersoftware</p> <p>Anforderungsanalyse – Kinder als Informanten</p> <p>Ermittlungstechniken</p> <p>Nutzerspezifische Ermittlungstechniken</p> <p>Konzept – Kinder als Nutzer</p> <p>Gestaltungsempfehlungen</p> <p>Screen-Design</p> <p>Konzeption einer einfachen Anwendung für Kinder auf Basis einer selbstgewählten Fragestellung unter Einsatz eines Grafik-Programms</p>
Voraussetzungen	Kenntnisse der Bewertung webbasierter Front-Ends
Modulbausteine	SWE205 Studienbrief Software-Ergonomie und Interaktionsdesign im Internet mit Onlineübung



Fachbuch Liebal; Exner: Usability für Kids. Ein Handbuch zur ergonomischen Gestaltung von Software und Websites für Kinder

Fachbuch Richter, Michael; Flückiger, Markus: Usability und UX kompakt. Produkte für Menschen

Fachbuch Thesmann, Stephan: Interface Design: Usability, User Experience und Accessibility im Web gestalten

Fachbuch Weichert, Steffen; Quint, Gesine; Bartel, Torsten: Quick Guide UX Management

Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Dr. Franz-Karl Schmatzer

WEB75 Konzeption und Weiterentwicklung komplexer Web-Anwendungen 1

Kompetenzzuordnung	Instrumentale Kompetenz
Kompetenzziele	<p>Nach erfolgreicher Teilnahme am Modul WEB75 kennen die Studierenden die Prinzipien der Software-Ergonomie und des User Interface Engineerings.</p> <p>Sie können daraus applikationsspezifische Anforderungen an Benutzeroberflächen ableiten, einen Styleguide selbst entwickeln, Testverfahren für die Bewertung der Usability planen und die Anwendungsmöglichkeiten dieser Verfahren beurteilen.</p> <p>Sie können den Zusammenhang zwischen funktionalen Merkmalen einer Applikation und Anforderungen an die Usability charakterisieren.</p> <p>Sie können offene Forschungsfragen zur Usability ermitteln und ihre Bedeutung für Anwendungsbereiche in der Praxis beurteilen und begründen.</p>
Inhalt	<p>Usability</p> <p>Übersicht: Stellenwert und Gestaltungsvorgaben</p> <p>Nutzer: Wahrnehmungsprozesse und Zielgruppen</p> <p>Gestaltungselemente: Bausteine für konsistente Applikationen</p> <p>Usability: Regeln für gute Software-Ergonomie</p> <p>User Interface Engineering</p> <p>Historische Entwicklung</p> <p>Kontext der Softwareentwicklung</p> <p>Phasen der Entwicklung</p> <p>User Experience Design</p> <p>Szenariobasierte Entwicklung</p> <p>Contextual Design</p> <p>Partizipative Entwicklung und Living Labs</p> <p>Beobachtungen, Befragungen und Workshops</p> <p>Werkzeuge für das User Interface Engineering</p> <p>Styleguides im User Interface Engineering</p> <p>Firmen- und plattformspezifische Richtlinien</p> <p>Erstellung von Styleguides</p> <p>Inhalt und Struktur von Styleguides</p> <p>Validierung und Verifikation</p> <p>Einführung von Styleguides</p> <p>Veränderungen von Styleguides</p> <p>Beispiele für Styleguides</p> <p>UX Styleguides</p> <p>Entwicklung eines Styleguides</p> <p>Entwicklung eines Styleguides für eine ausgewählte Web-Applikation</p>
Voraussetzungen	<p>Bewertungskriterien für Web-Applikationen</p> <p>Mehrdimensionale Scoringmodelle</p>

Modulbausteine	ABTE039-EL Fachbuch Preim; Dachsel: Interaktive Systeme – Band 2 – User Interface Engineering, 3D-Interaktion, Natural User Interfaces ABTE017-EL Fachbuch Stapelkamp: Informationsvisualisierung – Web – Print – Signalethik – Erfolgreiches Informationsdesign: Leitsysteme, Wissensvermittlung und Informationsarchitektur mit WEB901-RG Research Guide ABTE040-EL Fachbuch Thesmann: Interface Design – Usability, User Experience und Accessibility im Web gestalten
Kompetenznachweis	Assignment
Lernaufwand	125 Stunden, 5 Leistungspunkte
Sprache	Deutsch
Studienleiter	Mark Harwardt
