

MODULHANDBUCH

Web Development Expert (IU)

Weiterbildung Web Development Expert (UPS-CPWDE)

n/a ECTS

Fernstudium

Klassifizierung: Career Path

Inhaltsverzeichnis

1. Semester

Modul DLBINGOPJ: Grundlagen der objektorientierten Programmierung mit Java

| | |
|--|---|
| Modulbeschreibung | 7 |
| Kurs IOBP01: Grundlagen der objektorientierten Programmierung mit Java | 9 |

Modul DLBCSDSJCL2_D: Datenstruktur und Java-Klassenbibliothek

| | |
|---|----|
| Modulbeschreibung | 15 |
| Kurs DLBCSDSJCL02_D: Datenstruktur und Java-Klassenbibliothek | 17 |

Modul DLBIADPS: Algorithmen, Datenstrukturen und Programmiersprachen

| | |
|---|----|
| Modulbeschreibung | 23 |
| Kurs DLBIADPS01: Algorithmen, Datenstrukturen und Programmiersprachen | 25 |

Modul DLBECGP: Grundlagen der Web-Programmierung

| | |
|---|----|
| Modulbeschreibung | 31 |
| Kurs DLBECGP01: Grundlagen der Web-Programmierung | 33 |

2. Semester

Modul IPWA1-01: Programmierung von Web-Anwendungsoberflächen

| | |
|---|----|
| Modulbeschreibung | 43 |
| Kurs IPWA01-01: Programmierung von Webanwendungsoberflächen | 45 |

Modul IPWA2-01: Programmierung von industriellen Informationssystemen mit Java EE

| | |
|---|----|
| Modulbeschreibung | 49 |
| Kurs IPWA02-01: Programmierung von industriellen Informationssystemen mit Java EE | 51 |

Modul IWNF1: Techniken und Methoden der agilen Softwareentwicklung

| | |
|--|----|
| Modulbeschreibung | 55 |
| Kurs IWNF01: Techniken und Methoden der agilen Softwareentwicklung | 57 |

Modul IWNF2: Projekt Agiles Software Engineering

| | |
|--|----|
| Modulbeschreibung | 61 |
| Kurs IWNF02: Projekt Agiles Software Engineering | 63 |

1. Semester

Grundlagen der objektorientierten Programmierung mit Java

Modulcode: DLBINGOPJ

| Modultyp | Zugangsvoraussetzungen | Niveau | ECTS | Zeitaufwand Studierende |
|---------------|------------------------|--------|------|-------------------------|
| s. Curriculum | keine | BA | n/a | 150 h |

| Semester | Dauer | Regulär angeboten im | Kurs- und Prüfungssprache |
|---------------|-----------------------------|----------------------|---------------------------|
| s. Curriculum | Minimaldauer: 1 Semester | WiSe/SoSe | Deutsch |

Modulverantwortliche(r)

Prof. Dr. Damir Ismailovic (Grundlagen der objektorientierten Programmierung mit Java)

Kurse im Modul

- Grundlagen der objektorientierten Programmierung mit Java (IOBP01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Kombistudium

Klausur, 90 Minuten

Studienformat: myStudium

Klausur, 90 Minuten

Studienformat: Fernstudium

Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

| | |
|--|---|
| <p>Lehrinhalt des Moduls</p> <ul style="list-style-type: none"> ▪ Einführung in die Sprache Java ▪ Java-Sprachkonstrukte ▪ Einführung in die objektorientierte Systementwicklung ▪ Vererbung ▪ Objektorientierte Konzepte ▪ Ausnahmebehandlung ▪ Interfaces | |
| <p>Qualifikationsziele des Moduls</p> <p>Grundlagen der objektorientierten Programmierung mit Java</p> <p>Nach erfolgreichem Abschluss sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> ▪ die Grundkonzepte der objektorientierten Modellierung und Programmierung zu erläutern und sie voneinander abzugrenzen. ▪ die Grundkonzepte und -elemente der Programmiersprache Java zu beschreiben und haben Erfahrungen in deren Verwendung. ▪ konkret beschriebene Probleme selbstständig zu lösen. | |
| <p>Bezüge zu anderen Modulen im Studiengang</p> <p>Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung</p> | <p>Bezüge zu anderen Studiengängen der Hochschule</p> <p>Alle Bachelor-Programme im Bereich IT & Technik</p> |

Grundlagen der objektorientierten Programmierung mit Java

Kurscode: IOBP01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

Betriebliche Informationssysteme werden in der Regel objektorientiert geplant und programmiert. Daher werden in diesem Kurs grundlegende Kompetenzen der objektorientierten Programmierung vermittelt. Dabei werden die theoretischen Konzepte unmittelbar anhand der Programmiersprache Java gezeigt und geübt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die Grundkonzepte der objektorientierten Modellierung und Programmierung zu erläutern und sie voneinander abzugrenzen.
- die Grundkonzepte und -elemente der Programmiersprache Java zu beschreiben und haben Erfahrungen in deren Verwendung.
- konkret beschriebene Probleme selbstständig zu lösen.

Kursinhalt

1. Einführung in die objektorientierte Systementwicklung
 - 1.1 Objektorientierung als Sichtweise auf komplexe Systeme
 - 1.2 Das Objekt als Grundkonzept der Objektorientierung
 - 1.3 Phasen im objektorientierten Entwicklungsprozess
 - 1.4 Grundprinzip der objektorientierten Systementwicklung
2. Einführung in die objektorientierte Modellierung
 - 2.1 Strukturieren von Problemen mit Klassen
 - 2.2 Identifizieren von Klassen
 - 2.3 Attribute als Eigenschaften von Klassen
 - 2.4 Methoden als Funktionen von Klassen
 - 2.5 Beziehungen zwischen Klassen
 - 2.6 Unified Modeling Language (UML)

3. Programmieren von Klassen in Java
 - 3.1 Einführung in die Programmiersprache Java
 - 3.2 Grundelemente einer Klasse in Java
 - 3.3 Attribute in Java
 - 3.4 Methoden in Java
 - 3.5 main-Methode: Startpunkt eines Java-Programms
4. Java Sprachkonstrukte
 - 4.1 Primitive Datentypen
 - 4.2 Variablen
 - 4.3 Operatoren und Ausdrücke
 - 4.4 Kontrollstrukturen
 - 4.5 Pakete und Sichtbarkeitsmodifikatoren
5. Vererbung
 - 5.1 Modellierung von Vererbung im Klassendiagramm
 - 5.2 Programmieren von Vererbung in Java
6. Wichtige objektorientierte Konzepte
 - 6.1 Abstrakte Klassen
 - 6.2 Polymorphie
 - 6.3 Statische Attribute und Methoden
7. Konstruktoren zur Erzeugung von Objekten
 - 7.1 Der Standard-Konstruktor
 - 7.2 Überladen von Konstruktoren
8. Ausnahmebehandlung mit Exceptions
 - 8.1 Typische Szenarien der Ausnahmebehandlung
 - 8.2 Standard-Exceptions in Java
 - 8.3 Definieren eigener Exceptions
9. Programmierschnittstellen mit Interfaces
 - 9.1 Typische Szenarien für Programmierschnittstellen
 - 9.2 Interfaces als Programmierschnittstellen in Java

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Java (Hrsg.): Java Platform Standard Edition API Specification. (URL: <http://www.oracle.com/technetwork/java/api-141528.html> [letzter Zugriff: 21.11.2016]).
- Krüger G./Stark T. (2011): Handbuch der Java-Programmierung. 7. Auflage, Addison-Wesley, Salt Lake City.
- Lahres, B./Raýman, G. (2006): Praxisbuch Objektorientierung. Galileo Computing, Bonn.
- Oestereich B. (2012): Analyse und Design mit der UML 2.5. Objektorientierte Softwareentwicklung. 10. Auflage, Oldenbourg, München.
- Ratz, D. et al. (2011): Grundkurs Programmieren in Java. 6. Auflage, Carl Hanser Verlag, München.
- Ullenboom C. (2011): Java ist auch eine Insel. 10. Auflage, Galileo Computing, Bonn.

Studienformat Kombistudium

| | |
|------------------------------------|-----------------------------|
| Studienform Kombistudium | Kursart Vorlesung |
|------------------------------------|-----------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|--|--|
| Lehrmethoden | |
| <input checked="" type="checkbox"/> Learning Sprints® <input checked="" type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input checked="" type="checkbox"/> Shortcast <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Musterklausur | <input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input type="checkbox"/> Prüfungsleitfaden <input checked="" type="checkbox"/> Live Tutorium/Course Feed <input type="checkbox"/> Reader <input checked="" type="checkbox"/> Folien |

Studienformat myStudium

| | |
|---------------------------------|-----------------------------|
| Studienform myStudium | Kursart Vorlesung |
|---------------------------------|-----------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|---|---|
| Lehrmethoden | |
| <input checked="" type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien |

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|---|---|
| Lehrmethoden | |
| <input checked="" type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien |

Datenstruktur und Java-Klassenbibliothek

Modulcode: DLBCSDSJCL2_D

| | | | | |
|----------------------------------|--|---------------------|--------------------|---|
| Modultyp s. Curriculum | Zugangsvoraussetzungen keine | Niveau BA | ECTS n/a | Zeitaufwand Studierende 150 h |
|----------------------------------|--|---------------------|--------------------|---|

| | | | |
|----------------------------------|---|--|---|
| Semester s. Curriculum | Dauer Minimaldauer: 1 Semester | Regulär angeboten im WiSe/SoSe | Kurs- und Prüfungssprache Deutsch |
|----------------------------------|---|--|---|

Modulverantwortliche(r)

Prof. Dr. Damir Ismailovic (Datenstruktur und Java-Klassenbibliothek)

Kurse im Modul

- Datenstruktur und Java-Klassenbibliothek (DLBCSDSJCL02_D)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Advanced Workbook

Studienformat: Kombistudium
Advanced Workbook

Studienformat: myStudium
Advanced Workbook

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Programmierstil
- Arbeiten mit Objekten
- Externe Pakete und Bibliotheken
- Datenstrukturen
- Zeichenketten und Calendar
- Dateisystem und Datenströme

Qualifikationsziele des Moduls**Datenstruktur und Java-Klassenbibliothek**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- typische Datenstrukturen zu erläutern und diese voneinander abzugrenzen.
- in der Programmiersprache Java selbständig Lösungen unter Verwendung der Datenstrukturen zu erstellen.
- Szenarien und Strategien zum Vergleichen von Objekten zu skizzieren und diese in Java umzusetzen.
- Einsatzmöglichkeiten und Funktionen von Zeichenketten und Kalenderobjekten in Java zu erklären und zu verwenden.
- Einsatzmöglichkeiten und Funktionen von Streams in Java zu erläutern und in der Praxis anzuwenden.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Datenstruktur und Java-Klassenbibliothek

Kurscode: DLBCSDSJCL02_D

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

In diesem Kurs werden die Kenntnisse der objektorientierten Programmierung vertieft. Dabei werden insbesondere Datenstrukturen, deren Anwendungsfälle und deren Umsetzung in der Sprache Java betrachtet. Darüber hinaus werden Strategien und Szenarien von Objektvergleichen, die Verwendung von Funktionen des Datentyps „String“, der Einsatz von Kalenderobjekten sowie der Einsatz von Streams vermittelt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- typische Datenstrukturen zu erläutern und diese voneinander abzugrenzen.
- in der Programmiersprache Java selbständig Lösungen unter Verwendung der Datenstrukturen zu erstellen.
- Szenarien und Strategien zum Vergleichen von Objekten zu skizzieren und diese in Java umzusetzen.
- Einsatzmöglichkeiten und Funktionen von Zeichenketten und Kalenderobjekten in Java zu erklären und zu verwenden.
- Einsatzmöglichkeiten und Funktionen von Streams in Java zu erläutern und in der Praxis anzuwenden.

Kursinhalt

1. Programmierstil
 - 1.1 Code-Dokumentation
 - 1.2 Code-Annotationen
 - 1.3 Code-Konventionen
2. Arbeiten mit Objekten
 - 2.1 String-Darstellung von Objekten
 - 2.2 Vergleichen mit ==
 - 2.3 Vergleichen mit equals()
 - 2.4 Vergleichen per hashCode()
 - 2.5 compareTo()
 - 2.6 Klonen von Objekten

3. Externe Pakete und Bibliotheken
 - 3.1 Importieren von Paketen
 - 3.2 Die Java-Klassenbibliothek
4. Datenstrukturen
 - 4.1 Arrays
 - 4.2 Collections
 - 4.3 Mit Collections arbeiten
 - 4.4 Listen
 - 4.5 Mengen (Sets)
 - 4.6 Assoziativspeicher (Maps)
 - 4.7 Stacks (Keller)
 - 4.8 Queues (Schlangen)
5. Zeichenketten und Calendar
 - 5.1 Zeichenketten
 - 5.2 StringBuffer
 - 5.3 Aufteilen von Zeichenketten
 - 5.4 Datum und Uhrzeit
 - 5.5 Kalender
6. Dateisystem und Datenströme
 - 6.1 Arbeiten mit dem Dateisystem
 - 6.2 Arbeiten mit Dateien

Literatur

Pflichtliteratur

Weiterführende Literatur

- Java (Hrsg.). Java Platform Standard Edition API Specification. (URL: <http://www.oracle.com/technetwork/java/api-141528.html> [letzter Zugriff: 21.11.2016]).
- Krüger, G. & Stark, T. (2011). Handbuch der Java-Programmierung. 7. Auflage, Addison-Wesley, Salt Lake City.
- Lahres, B. & Raýman, G. (2006). Praxisbuch Objektorientierung. Galileo Computing, Bonn.
- Oestereich, B. (2012). Analyse und Design mit der UML 2.5. Objektorientierte Softwareentwicklung. 10. Auflage, Oldenbourg, München.
- Ratz, D. et al. (2011). Grundkurs Programmieren in Java. 6. Auflage, Carl Hanser Verlag, München.
- Ullenboom, C. (2011). Java ist auch eine Insel. 10. Auflage, Galileo Computing, Bonn.

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|-------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Advanced Workbook |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 110 h | 0 h | 20 h | 20 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

Studienformat Kombistudium

| | |
|------------------------------------|-----------------------------|
| Studienform Kombistudium | Kursart Vorlesung |
|------------------------------------|-----------------------------|

| | |
|--|-------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Advanced Workbook |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 110 h | 0 h | 20 h | 20 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

Studienformat myStudium

| | |
|---------------------------------|-----------------------------|
| Studienform myStudium | Kursart Vorlesung |
|---------------------------------|-----------------------------|

| | |
|--|-------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Advanced Workbook |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 110 h | 0 h | 20 h | 20 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

DLBCSDSJCL02_D

Algorithmen, Datenstrukturen und Programmiersprachen

Modulcode: DLBIADPS

| Modultyp | Zugangsvoraussetzungen | Niveau | ECTS | Zeitaufwand Studierende |
|---------------|------------------------|--------|------|-------------------------|
| s. Curriculum | IOBP01, DLBCSDSJCL02_D | BA | n/a | 150 h |

| Semester | Dauer | Regulär angeboten im | Kurs- und Prüfungssprache |
|---------------|-----------------------------|----------------------|---------------------------|
| s. Curriculum | Minimaldauer: 1 Semester | WiSe/SoSe | Deutsch |

Modulverantwortliche(r)

Prof. Dr. Paul Libbrecht (Algorithmen, Datenstrukturen und Programmiersprachen)

Kurse im Modul

- Algorithmen, Datenstrukturen und Programmiersprachen (DLBIADPS01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium

Klausur, 90 Minuten

Studienformat: myStudium

Klausur, 90 Minuten

Studienformat: Kombistudium

Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Datenstrukturen
- Algorithmen-Entwurf
- Wichtige Algorithmen
- Grundbegriffe XML
- Programmierparadigmen und Grundbegriffe von Programmiersprachen
- Überblick über verbreitete Programmiersprachen

Qualifikationsziele des Moduls**Algorithmen, Datenstrukturen und Programmiersprachen**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- grundlegende Datenstrukturen zu erklären und in konkreten Anwendungsfällen zu vergleichen und anzuwenden.
- grundlegende Algorithmen zu erklären.
- in konkreten Anwendungsfällen geeignete Algorithmen zu entwerfen oder auszuwählen sowie anzuwenden.
- den Einsatz XML als Datenstruktur sowie die wichtigsten Algorithmen und Konzepte zur Verarbeitung von XML-Dokumenten (DOM, SAX, XLS) zu erklären und in einfachen Anwendungsfällen anzuwenden.
- die verbreiteten Programmierparadigmen und Programmiersprachen zu erläutern und zu vergleichen.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für alle weiteren Module aus dem Bereich Informatik & Software-Entwicklung.

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Algorithmen, Datenstrukturen und Programmiersprachen

Kurscode: DLBIADPS01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | IOBP01, DLBCSDSJCL02_D |

Beschreibung des Kurses

Programmierung besteht im Wesentlichen daraus, für eine konkrete Aufgabenstellung geeignete Algorithmen und Datenstrukturen auszuwählen und diese in Programmcode umzusetzen. Dabei gibt es eine Vielzahl unterschiedlicher Programmiersprachen, die auf unterschiedlichen Vorgehensweisen beruhen und in denen Algorithmen und Datenstrukturen daher unterschiedlich umgesetzt werden. In diesem Modul werden diese bisher an konkreten Beispielen behandelten Konzepte systematisch aufbereitet und auf eine breitere Grundlage gestellt, um den Studierenden das notwendige Handwerkszeug für ein systematisches Vorgehen bei der Programmierung zu geben.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- grundlegende Datenstrukturen zu erklären und in konkreten Anwendungsfällen zu vergleichen und anzuwenden.
- grundlegende Algorithmen zu erklären.
- in konkreten Anwendungsfällen geeignete Algorithmen zu entwerfen oder auszuwählen sowie anzuwenden.
- den Einsatz XML als Datenstruktur sowie die wichtigsten Algorithmen und Konzepte zur Verarbeitung von XML-Dokumenten (DOM, SAX, XLS) zu erklären und in einfachen Anwendungsfällen anzuwenden.
- die verbreiteten Programmierparadigmen und Programmiersprachen zu erläutern und zu vergleichen.

Kursinhalt

1. Grundbegriffe
 - 1.1 Geschichte der Algorithmik
 - 1.2 Detaillierung und Abstraktion
 - 1.3 Kontrollstrukturen
 - 1.4 Datentypen
 - 1.5 Grundlegende Datenstrukturen

2. Datenstrukturen
 - 2.1 Weiterführende Datenstrukturen
 - 2.2 Blockchain
 - 2.3 Abstrakte Datentypen, Objekte und Klassen
3. Algorithmenentwurf
 - 3.1 Induktion, Iteration und Rekursion
 - 3.2 Methoden des Algorithmen-Entwurfs
 - 3.3 Korrektheit und Verifikation von Algorithmen
 - 3.4 Effizienz und Komplexität von Algorithmen
4. Grundlegende Algorithmen
 - 4.1 Traversieren und Linearisieren von Bäumen
 - 4.2 Sortieralgorithmen
 - 4.3 Suche in Zeichenketten
 - 4.4 Hash-Algorithmen
 - 4.5 Mustererkennung
5. XML
 - 5.1 Aufbau von XML-Dokumenten
 - 5.2 Zugriff auf XML-Dokumente mit DOM und SAX
 - 5.3 Transformation von XML-Dokumenten mittels XSL
 - 5.4 JSON als Alternative zu XML
6. Programmiersprachen
 - 6.1 Programmierparadigmen
 - 6.2 Ausführung von Programmen
 - 6.3 Typen von Programmiersprachen
 - 6.4 Syntax, Semantik und Pragmatik
 - 6.5 Variablen und Typsysteme
7. Überblick über wichtige Programmiersprachen
 - 7.1 Die C-Familie
 - 7.2 Java
 - 7.3 Matlab
 - 7.4 COBOL
 - 7.5 PHP und HTML
 - 7.6 Weitere Programmiersprachen

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Gumm H. P. /Sommer M. (2013): Einführung in die Informatik. 10. Auflage. Oldenbourg, München.
- Harel, D. (2006): Algorithmik. Die Kunst des Rechnens. Springer, Berlin/Heidelberg/New York.
- Vonhoegen, H. (2015): Einstieg in XML. Grundlagen, Praxis, Referenz. Rheinwerk Computing, Bonn.

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|---|---|
| Lehrmethoden | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien |

Studienformat myStudium

| | |
|---------------------------------|-----------------------------|
| Studienform myStudium | Kursart Vorlesung |
|---------------------------------|-----------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|---|---|
| Lehrmethoden | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien |

Studienformat Kombistudium

| | |
|------------------------------------|-----------------------------|
| Studienform Kombistudium | Kursart Vorlesung |
|------------------------------------|-----------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

Grundlagen der Web-Programmierung

Modulcode: DLBECGP

| | | | | |
|----------------------------------|--|---------------------|--------------------|---|
| Modultyp s. Curriculum | Zugangsvoraussetzungen keine | Niveau BA | ECTS n/a | Zeitaufwand Studierende 150 h |
|----------------------------------|--|---------------------|--------------------|---|

| | | | |
|----------------------------------|---|--|---|
| Semester s. Curriculum | Dauer Minimaldauer: 1 Semester | Regulär angeboten im WiSe/SoSe | Kurs- und Prüfungssprache Deutsch |
|----------------------------------|---|--|---|

Modulverantwortliche(r)

Prof. Dr. Maik Günther (Grundlagen der Web-Programmierung)

Kurse im Modul

- Grundlagen der Web-Programmierung (DLBECGP01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Kombistudium
Schriftliche Ausarbeitung: Fallstudie

Studienformat: Duales Studium
Klausur oder Schriftliche Ausarbeitung:
Fallstudie

Studienformat: Fernstudium
Schriftliche Ausarbeitung: Fallstudie

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Architektur und Grundkonzepte von Web-Anwendungen
- Statische Webseiten
- Server-seitige dynamische Web-Anwendungen
- Client-seitige dynamische Web-Anwendungen
- Entwicklungsvorgehensweisen und -werkzeuge

Qualifikationsziele des Moduls**Grundlagen der Web-Programmierung**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die notwendige Infrastruktur und die Vorgehensweise für die Entwicklung von Web-Anwendungen zu verstehen.
- die wichtigsten Architekturen für Web-Anwendungen zu erläutern.
- die wesentlichen Technologien für Client-basierte Web-Anwendungen zu bewerten.
- einfache statische Web-Anwendungen zu entwickeln.
- einfache PHP-Anwendungen zu kreieren.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Grundlagen der Web-Programmierung

Kurscode: DLBECGP01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

Ziel ist es, die grundlegenden Konzepte der Programmierung von Web-Anwendungen zu vermitteln und die Studierenden in die Lage zu versetzen, zumindest einfache Web-Anwendungen selbst zu entwickeln. Nach einem Überblick über die grundlegenden Konzepte von Web-Anwendungen, insbesondere die unterschiedlichen dafür verwendeten Architekturen, werden statische Webseiten vorgestellt. Darauf baut eine Einführung in die Entwicklung von dynamischen Webseiten auf, sowohl auf Server- als auch auf Client-Seite. Dabei wird insbesondere auf die Entwicklung mit PHP eingegangen. In diesem Kurs wird besonders darauf geachtet, nicht nur Technologien und Programmiersprachen vorzustellen, sondern diese in einen Entwicklungszyklus einzubetten, der die Qualität und Sicherheit der entwickelten Anwendungen sicherstellt und durch geeignete Werkzeuge unterstützt wird.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- die notwendige Infrastruktur und die Vorgehensweise für die Entwicklung von Web-Anwendungen zu verstehen.
- die wichtigsten Architekturen für Web-Anwendungen zu erläutern.
- die wesentlichen Technologien für Client-basierte Web-Anwendungen zu bewerten.
- einfache statische Web-Anwendungen zu entwickeln.
- einfache PHP-Anwendungen zu kreieren.

Kursinhalt

1. Internet und Web-Anwendungen
 - 1.1 Historie und Aufbau des Internets
 - 1.2 Architektur von Web-Anwendungen
 - 1.3 Internet-Protokolle und URIs
 - 1.4 Qualität von Web-Anwendungen
2. Statische Web-Seiten
 - 2.1 HTML
 - 2.2 CSS
 - 2.3 XML

3. Server-seitige Web-Programmierung
 - 3.1 Einführung in die Server-seitige Web-Programmierung
 - 3.2 JAVA
 - 3.3 Common Gateway Interface (CGI)
 - 3.4 Datenbank-Anbindung
4. Einführung in PHP
 - 4.1 Aufbau von PHP und Integration in HTML
 - 4.2 Kontrollstrukturen von PHP
 - 4.3 Funktionen in PHP
 - 4.4 Formulare mit PHP
5. Client-seitige Web-Programmierung
 - 5.1 Einführung in die Client-seitige Web-Programmierung
 - 5.2 JavaScript und DOM
 - 5.3 AJAX
 - 5.4 JSON
6. Entwicklung von Web-Anwendungen
 - 6.1 Vorgehensweisen für die Entwicklung von Web-Anwendungen
 - 6.2 Prüfung und Test von Web-Anwendungen
 - 6.3 Entwicklungswerkzeuge und -umgebungen
 - 6.4 Grundbegriffe des Webdesigns
 - 6.5 Sicherheit von Web-Anwendungen

Literatur

Pflichtliteratur

Weiterführende Literatur

- Avci, O./Trittmann, R./Mellis, W. (Hrsg.) (2003): Web-Programmierung, Vieweg, Wiesbaden.
- Balzert, H. (2017): Basiswissen Web-Programmierung. 2. Auflage, Springer Campus, Wiesbaden.
- Meinel, C./Sack, H. (2012): WWW: Kommunikation, Internetworking, Web-Technologien. Springer-Verlag Berlin, Heidelberg/New York.
- SELFHTML e. V. (Hrsg.) (2018): SELFHTML Wiki. (URL: <https://wiki.selfhtml.org/wiki/> [letzter Zugriff: 21.01.2019]).
- The PHP Group (2019): PHP-Handbuch (URL: <http://php.net/manual/de/index.php> [letzter Zugriff: 21.01.2019]).

Studienformat Kombistudium

| | |
|------------------------------------|-----------------------------|
| Studienform Kombistudium | Kursart Vorlesung |
|------------------------------------|-----------------------------|

| | |
|--|---------------------------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Schriftliche Ausarbeitung: Fallstudie |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 110 h | 0 h | 20 h | 20 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

Studienformat Duales Studium

| | |
|--------------------------------------|---|
| Studienform Duales Studium | Kursart Integrierte Vorlesung |
|--------------------------------------|---|

| | |
|--|---|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur oder Schriftliche Ausarbeitung: Fallstudie |

| | | | | | |
|----------------------------------|---------------------------------|---------------------------|---------------------------------|----------------------------|------------------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium 129,75 h | Präsenzstudium 13,5 h | Tutorium 6,75 h | Selbstüberprüfung 0 h | Praxisanteil 0 h | Gesamt 150 h |

| |
|--|
| Lehrmethoden |
| Der Kurs verbindet die interaktive Präsenzlehre mit einer online unterstützten Selbstlernphase. Während der Präsenzphase werden Studierende gezielt bei der Übung und Vertiefung der vermittelten Inhalte begleitet. |

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---------------------------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Schriftliche Ausarbeitung: Fallstudie |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 110 h | 0 h | 20 h | 20 h | 0 h | 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

DLBECGP01

2. Semester

Programmierung von Web-Anwendungsoberflächen

Modulcode: IPWA1-01

| Modultyp | Zugangsvoraussetzungen | Niveau | ECTS | Zeitaufwand Studierende |
|---------------|------------------------|--------|------|-------------------------|
| s. Curriculum | keine | BA | n/a | 150 h |

| Semester | Dauer | Regulär angeboten im | Kurs- und Prüfungssprache |
|---------------|-----------------------------|----------------------|---------------------------|
| s. Curriculum | Minimaldauer: 1 Semester | WiSe/SoSe | Deutsch |

Modulverantwortliche(r)

Prof. Dr. Marian Benner-Wickner (Programmierung von Webanwendungsoberflächen)

Kurse im Modul

- Programmierung von Webanwendungsoberflächen (IPWA01-01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur oder Schriftliche Ausarbeitung:
Fallstudie, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Aufbau und Überblick von Web-Architekturen
- Statische Webseiten
- Grundlegende Java-Web-Technologien
- Komponentenbasierte Web-Benutzerschnittstellen
- Verknüpfung von View und Model
- Komponentenbibliotheken

Qualifikationsziele des Moduls**Programmierung von Webanwendungsoberflächen**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- wichtige Elemente zu identifizieren und den Aufbau aktueller Web-Anwendungsarchitekturen zu beschreiben.
- einfache statische Webseiten mit HTML zu realisieren.
- Webseiten mit CSS und CSS-Frameworks zu gestalten.
- dynamische Webseiten mit JavaScript zu entwickeln.
- den Umgang mit relevanten Tools zur Entwicklung und Quellcode-Verwaltung von Webseiten zu beherrschen.
- Konzepte zum Testen von Webanwendungen zu verstehen.
- typische Sicherheitsprobleme von Webseiten zu erkennen und zu wissen, wie sie überwunden werden können.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Programmierung von Webanwendungsoberflächen

Kurscode: IPWA01-01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

Ziel dieses Kurses ist es, die Studierenden in die Lage zu versetzen, einfache Webanwendungen mit etablierten Technologien zu programmieren. Zunächst erhalten die Studierenden wichtige Einblicke in den typischen Aufbau aktueller Web-Anwendungsarchitekturen. Darauf aufbauend erlernen die Studierenden die Hypertext Markup Language (HTML), um einfache Webseiten zu entwickeln. In Ergänzung sollen die Studierenden sich mit den wichtigsten und gängigsten Elementen des Cascading Stylesheet (CSS) Standards vertraut machen, um die in HTML geschriebenen Inhalte zu layouten. Um einfache dynamische Webseiten zu realisieren, lernen die Studierenden den Einsatz von JavaScript und exemplarisch unterstützende Frameworks. Damit verbunden ist der Einsatz einschlägiger Tools zur Entwicklung- und Quellcode-Verwaltung von Webseiten. Schließlich lernen sie die Grundlagen des Testens und der Sicherheit von Webseiten kennen und zu bewerten.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- wichtige Elemente zu identifizieren und den Aufbau aktueller Web-Anwendungsarchitekturen zu beschreiben.
- einfache statische Webseiten mit HTML zu realisieren.
- Webseiten mit CSS und CSS-Frameworks zu gestalten.
- dynamische Webseiten mit JavaScript zu entwickeln.
- den Umgang mit relevanten Tools zur Entwicklung und Quellcode-Verwaltung von Webseiten zu beherrschen.
- Konzepte zum Testen von Webanwendungen zu verstehen.
- typische Sicherheitsprobleme von Webseiten zu erkennen und zu wissen, wie sie überwunden werden können.

Kursinhalt

1. Architektonische Fundamente
 - 1.1 Struktur und Geschichte des Internets
 - 1.2 Internetprotokolle und URIs
 - 1.3 Architektur von Webanwendungen
 - 1.4 Aktuelle Trends

2. Werkzeuge der Webentwicklung
 - 2.1 Entwicklungstools
 - 2.2 Versionsverwaltung
 - 2.3 Paketmanager
 - 2.4 Upload/Bereitstellung
3. Entwicklung von statischen Webseiten
 - 3.1 Grundlagen von HTML5
 - 3.2 Grundlagen von Cascading Style Sheets
4. Erweiterte Konstruktionstechniken
 - 4.1 Responsives Web-Design
 - 4.2 Seitenlayout
 - 4.3 Media Queries
 - 4.4 CSS-Frameworks
5. Webseitenentwicklung mit JavaScript
 - 5.1 JavaScript-Geschichte, ES5/ES6
 - 5.2 JavaScript-Grundlagen
 - 5.3 Verwendung von JSON
 - 5.4 Gängige JavaScript-Frameworks
6. Testen und Sicherheit von Webanwendungen
 - 6.1 Testen von Webanwendungen
 - 6.2 Grundlegende Sicherheitskonzepte und -prinzipien

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Libby, A, Gupta, G. & Talesra, A. (2016). Responsive Web Design with HTML5 and CSS3 Essentials.
- Ferguson, R. (2019). Beginning JavaScript. The Ultimate Guide to Modern JavaScript Development. (3. Aufl.). Apress.

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur oder Schriftliche Ausarbeitung: Fallstudie, 90 Minuten |

| | | | | | |
|--------------------------------|------------------------------|-------------------------|----------------------------------|----------------------------|------------------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium 100 h | Präsenzstudium 0 h | Tutorium 25 h | Selbstüberprüfung 25 h | Praxisanteil 0 h | Gesamt 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input checked="" type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

IPWA01-01

Programmierung von industriellen Informationssystemen mit Java EE

Modulcode: IPWA2-01

| Modultyp | Zugangsvoraussetzungen | Niveau | ECTS | Zeitaufwand Studierende |
|---------------|------------------------|--------|------|-------------------------|
| s. Curriculum | keine | BA | n/a | 150 h |

| Semester | Dauer | Regulär angeboten im | Kurs- und Prüfungssprache |
|---------------|-----------------------------|----------------------|---------------------------|
| s. Curriculum | Minimaldauer: 1 Semester | WiSe/SoSe | Deutsch |

Modulverantwortliche(r)

Prof. Dr. Marian Benner-Wickner (Programmierung von industriellen Informationssystemen mit Java EE)

Kurse im Modul

- Programmierung von industriellen Informationssystemen mit Java EE (IPWA02-01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur oder Schriftliche Ausarbeitung:
Fallstudie

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Programmierung von Geschäftslogik
- Programmierung von Datenbankanbindungen

Qualifikationsziele des Moduls**Programmierung von industriellen Informationssystemen mit Java EE**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- relevante Technologien, Standards und Frameworks zur Erstellung von Enterprise-Web-Anwendungen mit dem Java-Technologie-Stack zu beschreiben und sie voneinander abzugrenzen.
- verschiedenen Web-Technologien zur Implementierung und Integration von Anwendungsoberflächen, Geschäftslogik und Datenschicht einzusetzen.
- selbstständig geeignete Web-Anwendungen zur Lösung von konkret beschriebenen Problemen zu erstellen.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für weitere Module im Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Programmierung von industriellen Informationssystemen mit Java EE

Kurscode: IPWA02-01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|-----------------------------------|
| BA | Deutsch | | n/a | IPWA01-01, IOBP01, DLBCSDSJCL02_D |

Beschreibung des Kurses

Aufbauend auf den Kurs „Programmierung von Web-Anwendungen“ steht in diesem Kurs der Java-Technologie-Stack zur Erstellung von Web-Anwendungen im Fokus. Nach einer Einführung in die Entwicklung von Web-Anwendungsoberflächen mit JSF werden Funktionen zur Validierung und Konvertierung von Eingaben und die Programmierung von Fehlermeldungen behandelt. Anschließend werden Konzepte zur Umsetzung von Navigationsstrukturen vermittelt. Zum Schluss werden Konzepte und Technologien zur Anbindung an Datenbanksystemen vorgestellt und angewandt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- relevante Technologien, Standards und Frameworks zur Erstellung von Enterprise-Web-Anwendungen mit dem Java-Technologie-Stack zu beschreiben und sie voneinander abzugrenzen.
- verschiedenen Web-Technologien zur Implementierung und Integration von Anwendungsoberflächen, Geschäftslogik und Datenschicht einzusetzen.
- selbstständig geeignete Web-Anwendungen zur Lösung von konkret beschriebenen Problemen zu erstellen.

Kursinhalt

1. Komponentenbasierte Web-Benutzerschnittstellen
 - 1.1 Einführung in Java Server Faces (JSF)
 - 1.2 JSF-Komponenten
2. Verknüpfung von View und Model
 - 2.1 Beans
 - 2.2 Unified Expression Language (UEL)

3. Komponentenbibliotheken
 - 3.1 Komponentenbibliotheken
 - 3.2 PrimeFaces
 - 3.3 AJAX
4. Programmierung von Geschäftslogik
 - 4.1 Validierung von Benutzereingaben
 - 4.2 Konvertierung von Zahlen, Währungen und Datumsfeldern
 - 4.3 Definition von Fehlermeldungen
 - 4.4 Seitennavigation mit JSF-Anwendungen
 - 4.5 JSF-Lebenszyklus und Eventhandling
5. Programmierung von Datenbankanbindungen
 - 5.1 Persistente Speicherung in Datenbanken
 - 5.2 Objektrelationales Mapping
 - 5.3 Datenbankpersistenz mit JPA und Hibernate als Persistence Provider
 - 5.4 Erzeugen von Entitäten
 - 5.5 Assoziationen zwischen Objekten mittels JPA
 - 5.6 Löschen von Objekten mittels JPA

Literatur

Pflichtliteratur

Weiterführende Literatur

- Balzert, H. (2011): Basiswissen Web-Programmierung (2. Aufl.). W3L
- Marinschek, M. (2014): JavaServer Faces 2.2: Grundlagen und erweiterte Konzepte (3. Aufl.). dpunkt.
- Müller, B. (2010): Java Server Faces 2.0: Ein Arbeitsbuch für die Praxis (2. Aufl.). Carl Hanser.
- Wehr, H. & Müller, B. (2012): Java Persistence API 2: Hibernate, EclipseLink, OpenJPA und Erweiterungen. Carl Hanser.

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur oder Schriftliche Ausarbeitung: Fallstudie |

| | | | | | |
|--------------------------------|------------------------------|-------------------------|----------------------------------|----------------------------|------------------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium 100 h | Präsenzstudium 0 h | Tutorium 25 h | Selbstüberprüfung 25 h | Praxisanteil 0 h | Gesamt 150 h |

| | | |
|---|---|--|
| Lehrmethoden | | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium | <input type="checkbox"/> Sprint |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab | <input type="checkbox"/> Interaktive Lehrveranstaltung |
| <input type="checkbox"/> Vodcast | <input checked="" type="checkbox"/> Prüfungsleitfaden | |
| <input checked="" type="checkbox"/> Shortcast | <input checked="" type="checkbox"/> Live Tutorium/Course Feed | |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader | |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien | |

IPWA02-01

Techniken und Methoden der agilen Softwareentwicklung

Modulcode: IWNF1

| Modultyp | Zugangsvoraussetzungen | Niveau | ECTS | Zeitaufwand Studierende |
|---------------|------------------------|--------|------|-------------------------|
| s. Curriculum | keine | BA | n/a | 150 h |

| Semester | Dauer | Regulär angeboten im | Kurs- und Prüfungssprache |
|---------------|-----------------------------|----------------------|---------------------------|
| s. Curriculum | Minimaldauer: 1 Semester | WiSe/SoSe | Deutsch |

Modulverantwortliche(r)

Prof. Dr. Damir Ismailovic (Techniken und Methoden der agilen Softwareentwicklung)

Kurse im Modul

- Techniken und Methoden der agilen Softwareentwicklung (IWNF01)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Klausur, 90 Minuten

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

- Merkmale und Prinzipien von Agilität
- Agilität in kleinen Teams mit Scrum
- Agiles Portfolio- und Projektmanagement
- Agiles Anforderungs- und IT-Architekturmanagement
- Agiles Testen
- Agile Delivery and Deployment

Qualifikationsziele des Moduls**Techniken und Methoden der agilen Softwareentwicklung**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Probleme und Risiken der industriellen SW-Entwicklung und ihre Konsequenzen für Entwicklungsprozesse zu analysieren und zu beurteilen.
- die Grundprinzipien des „No-Frills Software Engineering“ zu erläutern.
- Praxisszenarien zu analysieren und selbständig geeignete Methoden und Werkzeuge des „No-Frills Software Engineering“ anzuwenden.

Bezüge zu anderen Modulen im Studiengang

Ist Grundlage für alle weiteren Module aus dem Bereich Informatik & Software-Entwicklung

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Techniken und Methoden der agilen Softwareentwicklung

Kurscode: IWNF01

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

Ziel des Kurses ist es, den Studierenden einen vertiefenden Einblick in das Thema agile Softwareentwicklung zu vermitteln. Dazu werden zunächst die grundlegenden Merkmale und Prinzipien von Agilität vorgestellt und diskutiert. Danach wird dargestellt, wie kleine Projekt und Teams agiles Software-Engineering betreiben können und wie sich die agilen Prinzipien auf große Projekte übertragen und dort anwenden lassen. Anschließend werden agile Techniken für ausgewählte Kernaktivitäten im Software-Engineering vermittelt, wobei ein Schwerpunkt auf dem Gebiet Testen, Delivery und Deployment liegt.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- Probleme und Risiken der industriellen SW-Entwicklung und ihre Konsequenzen für Entwicklungsprozesse zu analysieren und zu beurteilen.
- die Grundprinzipien des „No-Frills Software Engineering“ zu erläutern.
- Praxisszenarien zu analysieren und selbständig geeignete Methoden und Werkzeuge des „No-Frills Software Engineering“ anzuwenden.

Kursinhalt

1. Merkmale und Prinzipien von Agilität
 - 1.1 Merkmale und Herausforderungen von Softwareprojekten
 - 1.2 Klassifikation von Unsicherheit
 - 1.3 Gegenüberstellung von agiler und klassischer Softwareentwicklung
 - 1.4 Prinzipien von Agilität
2. Agilität in kleinen Teams mit Scrum
 - 2.1 Grundlagen und allgemeiner Aufbau mit Scrum
 - 2.2 Zentrales Managementartefakt: Product Backlog
 - 2.3 Weitere Managementartefakte

3. Agiles Portfolio- und Projektmanagement
 - 3.1 Planungsebenen im agilen Projektmanagement
 - 3.2 Agiles Portfoliomanagement
 - 3.3 Organisation mehrerer Teams in einem Projekt
 - 3.4 Produkt- und Release-Planung
4. Agiles Anforderungs- und IT-Architekturmanagement
 - 4.1 Requirements Engineering in agilen Projekten
 - 4.2 Architekturmanagement in agilen Projekten
5. Agiles Testen
 - 5.1 Grundlagen und Anforderungen an die QS-Organisation
 - 5.2 Teststufen und Agilität
 - 5.3 Testautomatisierung
6. Agile Delivery and Deployment
 - 6.1 Grundlagen und Continuous Delivery Pipeline
 - 6.2 Continuous Build and Continuous Integration
 - 6.3 Akzeptanztests, Lasttests und Continuous Deployment

Literatur**Pflichtliteratur****Weiterführende Literatur**

- Baumgartner, M. et al. (2013): Agile Testing. Der agile Weg zur Qualität. Hanser, München.
- Biffel, S. et al. (Hrsg.) (2005): Value-Based Software Engineering. Springer, Berlin/Heidelberg.
- Cockburn, A. (2007): Agile Software Development. The Cooperative Game. 2. Auflage, Addison-Wesley, Upper Saddle River (NJ).
- DeMarco, T. (2003): Bärenango. Mit Risikomanagement Projekte zum Erfolg führen. Hanser, München.
- Epping, T. (2011): Kanban für die Softwareentwicklung. Springer, Berlin/Heidelberg.
- Geirhos, M. (2011): IT-Projektmanagement. Was wirklich funktioniert – und was nicht. Galileo Computing, Bonn.
- Hummel, H. (2011): Aufwandsschätzungen in der Software- und Systementwicklung. Spektrum, Wiesbaden.
- Künneth, T. (2012): Android 4. Apps entwickeln mit dem Android SDK. Galileo Computing, Bonn.
- Link, J. (2005): Softwaretests mit JUnit. 2.Auflage, dpunkt.verlag, Heidelberg.
- Mangold, P. (2009): IT-Projektmanagement. 3. Auflage, Spektrum, Wiesbaden.
- Motzel, E./O. Pannenbäcker (1998): Projektmanagement-Kanon. Der deutsche Zugang zum Project Management Body of Knowledge. TÜV-Verlag, Köln.
- Pichler, R. (2007): Scrum. Agiles Projektmanagement erfolgreich einsetzen. dpunkt.verlag, Heidelberg. (2007)
- Röpstorff, S./Wiechmann, R. (2012): Scrum in der Praxis. Erfahrungen, Problemfelder und Erfolgsfaktoren. dpunkt.verlag, Heidelberg.
- Rubin, K. S. (2014): Essential Scrum. Umfassendes Scrum-Wissen aus der Praxis. mitp, Frechen.
- Tiemeyer, E. (2010): Handbuch IT-Projektmanagement, Vorgehensmodelle, Managementinstrumente, Good Practices. Hanser, München.
- Wirdemann, R. (2011): Scrum mit User Stories. 2. Auflage, Hanser, München.
- Wolff, E. (2014): Continuous Delivery. Der pragmatische Einstieg. dpunkt.verlag, Heidelberg.
- Wolf, H./Bleek/W.-G. (2010): Agile Softwareentwicklung. Werte, Konzepte und Methoden. 2. Auflage, dpunkt.verlag, Heidelberg.

Studienformat Fernstudium

| | |
|-----------------------------------|------------------------------------|
| Studienform Fernstudium | Kursart Online-Vorlesung |
|-----------------------------------|------------------------------------|

| | |
|--|---------------------|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Ja |
| Prüfungsleistung | Klausur, 90 Minuten |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 90 h | 0 h | 30 h | 30 h | 0 h | 150 h |

| | |
|---|--|
| Lehrmethoden | |
| <input type="checkbox"/> Learning Sprints® | <input type="checkbox"/> Repetitorium |
| <input checked="" type="checkbox"/> Skript | <input type="checkbox"/> Creative Lab |
| <input type="checkbox"/> Vodcast | <input type="checkbox"/> Prüfungsleitfaden |
| <input checked="" type="checkbox"/> Shortcast | <input type="checkbox"/> Live Tutorium/Course Feed |
| <input checked="" type="checkbox"/> Audio | <input type="checkbox"/> Reader |
| <input checked="" type="checkbox"/> Musterklausur | <input checked="" type="checkbox"/> Folien |

Projekt Agiles Software Engineering

Modulcode: IWNF2

| | | | | |
|----------------------------------|--|---------------------|--------------------|---|
| Modultyp s. Curriculum | Zugangsvoraussetzungen keine | Niveau BA | ECTS n/a | Zeitaufwand Studierende 150 h |
|----------------------------------|--|---------------------|--------------------|---|

| | | | |
|----------------------------------|---|--|---|
| Semester s. Curriculum | Dauer Minimaldauer: 1 Semester | Regulär angeboten im WiSe/SoSe | Kurs- und Prüfungssprache Deutsch |
|----------------------------------|---|--|---|

Modulverantwortliche(r)

Prof. Dr. Damir Ismailovic (Projekt Agiles Software Engineering)

Kurse im Modul

- Projekt Agiles Software Engineering (IWNF02)

Art der Prüfung(en)

Modulprüfung

Studienformat: Fernstudium
Schriftliche Ausarbeitung: Projektbericht

Teilmodulprüfung

Anteil der Modulnote an der Gesamtnote

s. Curriculum

Lehrinhalt des Moduls

Umsetzung und Dokumentation von Projekten zum Einsatz agiler Techniken sowie Vertiefung ausgewählter Themengebiete aus dem Kurs IWNF01.

Qualifikationsziele des Moduls**Projekt Agiles Software Engineering**

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- typische Probleme in verschiedenen Projektsituationen durch den gezielten Einsatz von agilen Techniken und Methoden zu adressieren.
- die Konzeption und den projektspezifischen Einsatz von Techniken und Werkzeugen zu dokumentieren.

Bezüge zu anderen Modulen im Studiengang

Baut auf Modulen aus dem Bereich Informatik & Software-Entwicklung auf

Bezüge zu anderen Studiengängen der Hochschule

Alle Bachelor-Programme im Bereich IT & Technik

Projekt Agiles Software Engineering

Kurscode: IWNF02

| Niveau | Kurs- und Prüfungssprache | SWS | ECTS | Zugangsvoraussetzungen |
|--------|---------------------------|-----|------|------------------------|
| BA | Deutsch | | n/a | keine |

Beschreibung des Kurses

Aufbauend auf ihren Kenntnissen im Bereich der agilen Softwareentwicklung setzen die Studierenden in diesem Kurs selbstständig Projekte in diesem Themengebiet um und dokumentieren ihr Ergebnis in Form einer schriftlichen Ausarbeitung.

Kursziele

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,

- typische Probleme in verschiedenen Projektsituationen durch den gezielten Einsatz von agilen Techniken und Methoden zu adressieren.
- die Konzeption und den projektspezifischen Einsatz von Techniken und Werkzeugen zu dokumentieren.

Kursinhalt

- Umsetzung und Dokumentation eines Projektes unter Einsatz agiler Techniken sowie Vertiefung der Kenntnisse im Bereich agiler Softwareentwicklung.

Literatur

Pflichtliteratur

Weiterführende Literatur

- Pichler, R. (2007): Scrum. Agiles Projektmanagement erfolgreich einsetzen. dpunkt.verlag, Heidelberg.
- Röpstorff, S./Wiechmann, R. (2012): Scrum in der Praxis. Erfahrungen, Problemfelder und Erfolgsfaktoren. dpunkt.verlag, Heidelberg.

Studienformat Fernstudium

| | |
|-----------------------------------|---------------------------|
| Studienform Fernstudium | Kursart Projekt |
|-----------------------------------|---------------------------|

| | |
|--|---|
| Informationen zur Prüfung | |
| Prüfungszulassungsvoraussetzungen | BOLK: Nein |
| Prüfungsleistung | Schriftliche Ausarbeitung: Projektbericht |

| | | | | | |
|--------------------------------|-----------------------|-----------------|--------------------------|---------------------|---------------|
| Zeitaufwand Studierende | | | | | |
| Selbststudium | Präsenzstudium | Tutorium | Selbstüberprüfung | Praxisanteil | Gesamt |
| 120 h | 0 h | 30 h | 0 h | 0 h | 150 h |

| | |
|---|---|
| Lehrmethoden | |
| <input type="checkbox"/> Learning Sprints® <input type="checkbox"/> Skript <input type="checkbox"/> Vodcast <input type="checkbox"/> Shortcast <input type="checkbox"/> Audio <input type="checkbox"/> Musterklausur | <input type="checkbox"/> Repetitorium <input type="checkbox"/> Creative Lab <input checked="" type="checkbox"/> Prüfungsleitfaden <input type="checkbox"/> Live Tutorium/Course Feed <input checked="" type="checkbox"/> Folien |