

Modulhandbuch

für das

Weiterbildende Studium

**„Softwareentwicklung
und Software Engineering“**

Fachbereich Mathematik / Informatik
Akademie für Weiterbildung
Universität Bremen

vom [25.02.2019]

Version: [1.2]

Modulbeschreibungen

Übersicht

Modul-Nr.	Modul-Titel	CP	Seite
	Softwareentwicklung / Programmierung		
Modul 01	Grundlagen der Informatik	6	
Modul 02	Objektorientierte Programmierung	9	
Modul 03	Praktische Informatik	6	
Modul 04	Data Science mit Python	3	
		24	
	Software Engineering		
Modul 05	Datenbanken	3	
Modul 06	Webentwicklung	6	
Modul 07	Softwaretechnik (I)	9	
Modul 08	Softwaretechnik (II)	6	
		24	
	Methoden- und Sozialkompetenzen		
Modul 09	Empowerment und individuelle Profilierung		
	Präsentation und Moderation	2	
	Teamarbeit und Kreativität	2	
	Bewerbungstraining und Karrierecoaching	2	
		6	
	Betriebspraktikum		
Modul 10	Betriebliches Praxisprojekt	6	
	<i>CP gesamt:</i>	60	

Modul 1

Modulkennzeichen	SWE-01
Modulbezeichnung (ggf. Untertitel)	Grundlagen der Informatik
Englischer Titel	Foundations of Computer Science
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Anna Förster Universität Bremen, Fachbereich 1: Physik / Elektrotechnik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Logik und Zahlensysteme <ul style="list-style-type: none"> • Aussagen- und Prädikatenlogik • Dezimal- und Binärsystem • Hexadezimal- und Oktalsystem 2. Einführung in die Programmiersprache C <ul style="list-style-type: none"> • Bezeichner und Namen • Literale und Schlüsselwörter • Grundstruktur eines C-Programms 3. Umgang mit einfachen Datentypen <ul style="list-style-type: none"> • Strukturierte und unstrukturierte Datentypen • Gleitkommatypen • Zeichenketten • Wahrheitswerte • Typenumwandlung 4. Einfache und strukturierte Anweisungen sowie Abläufe <ul style="list-style-type: none"> • Anweisungen • Blöcke und ihre Struktur • Entscheidungs- und Wiederholungsanweisungen 5. Deklaration und Aufruf von Funktionen sowie Rekursionen <ul style="list-style-type: none"> • Deklaration von Funktionen • Parameterübergabe- und -rückgabe • Überladen von Funktionen • Rekursionen / Rekursion vs. Iteration
Lernergebnisse / Kompetenzen (Learning Outcome)	Das Modul vermittelt grundlegende Kenntnisse in der Programmierung. Die Teilnehmer/innen sollen befähigt werden, einfache Strukturen eines Programmes zu erkennen und zu unterscheiden. Sie kennen am Ende des Moduls Programmiersprachen als Notation zur Überführung von Algorithmen in Programmcode. Mittels der ver-

	<p>mittelten, grundlegenden Kenntnisse in der Programmiersprache C, sowie den dazugehörigen Strukturen, können die erarbeiteten Algorithmen in Programmcode umgesetzt werden und mit Hilfe der Arduino Plattform praktisch angewendet werden.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - kennen die Studierenden den Begriff der Logik in Bezug auf die Informatik und können ihn anwenden - wissen sie, welchen Zahlensysteme es gibt und kennen deren Anwendungsbereiche - kennen sie C als imperative und prozedurale Programmiersprache und können einfache Programme implementieren - haben sie einen Überblick über einfache Datentypen sowie deren Aufbau, Struktur und Anwendung und können klassifizieren - können sie Anweisungen, Abläufe, Aktionen und Steuermechanismen erklären sowie implementieren - verstehen sie die Struktur von Methoden und können sie veranschaulichen - kennen sie die grundlegenden Eigenschaften von Algorithmen und können einfache Algorithmen entwerfen sowie implementieren
Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	180 Stunden, davon <u>Präsenzzeit</u> : 160 Stunden <u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 20 Stunden
Leistungspunkte ECTS-Punkte / Credit Points (CP)	6 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	Kombinationsprüfung: i.d.R. Bearbeitung von Übungsaufgaben und mündliche Prüfung oder Klausur
Unterlagen (Skripte, Literatur, Programme usw.)	<ul style="list-style-type: none"> - Grundkurs C: C-Programmierung verständlich erklärt, Jürgen Wolf, ISBN-13: 978-3836241144, Rheinwerk Computing, Auflage 2: (30.05.2016) - Grundlagen der Informatik (Pearson Studium- IT), Helmut Herold und Bruno Lurz, Jürgen Wohlrab, Mathias Hopf, ISBN-13: 978-3868943160, 3., aktualisierte Auflage, 1.September 2017 - Mit Arduino die elektronische Welt entdecken, Erik Bartmann, ISBN-13: 978-3946496007, Bombini Verlags GmbH, Auflage: 3 (25.08.2017)
Lehrende	N. N.

	Prof. Dr. Anna Förster, Universität Bremen, Fachbereich 1: Physik / Elektrotechnik et al.
--	---

Modul 2

Modulkennzeichen	SWE-02
Modulbezeichnung (ggf. Untertitel)	Objektorientierte Programmierung
Englischer Titel	Object-oriented Programing
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Anna Förster Universität Bremen, Fachbereich 1: Physik / Elektrotechnik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Prinzipien der objektorientierten Programmierung <ul style="list-style-type: none"> • Abstraktion und Kapselung • Vererbung & Polymorphismus • Methoden Vs. Operationen 2. Einführung in die Programmiersprache Java <ul style="list-style-type: none"> • Bezeichner, Namen und Literale • Schlüsselwörter • Grundstruktur eines Java-Programms 3. Objektorientierte Beziehungen, Klassen und Objekte <ul style="list-style-type: none"> • Aggregation und Komposition • Generalisierung und Spezialisierung • Definition von Klasse und Objekten • Klassenmethoden und -variablen 4. Vererbung und Interfaces <ul style="list-style-type: none"> • Vererbung und Klassenhierarchien • Überschreiben Vs. Überladen • Polymorphe Beziehungen • Definition und Verwendung von Interfaces • Mehrfachimplementierung und –vererbung • Interfaces Vs. abstrakte Klassen 5. Generics & Annotationen <ul style="list-style-type: none"> • Definition und Einsatz von Generics • Wildcards & Bounds • Definition und Einsatz von Annotationen • Implementierung von Annotationen 6. Exceptions <ul style="list-style-type: none"> • Definition von Exceptions • Behandlung von Exceptions • Checked & Unchecked Exceptions

	<p>7. Abstrakte Datenstrukturen</p> <ul style="list-style-type: none"> • Aufzählungstypen • Listen • Stapelspeicher • Warteschlangen • Binärbäume <p>8. Sortieralgorithmen und Suchverfahren</p> <ul style="list-style-type: none"> • Suche in sortierten Listen • Lineare, sequentielle und binäre Suche • Sortierung mittels Einfügen und Auswählen • Bubble-, Quick- und Mergesort <p>9. Binäre Such- und ausgeglichene Bäume</p> <ul style="list-style-type: none"> • Spezifikation von Bäumen in der Informatik • Traversierung (Haupt- und Nebenreihenfolge, symmetrische Reihenfolge) • Balancierte Bäume • Suchen, Einfügen und Löschen
<p>Lernergebnisse / Kompetenzen (Learning Outcome)</p>	<p>Nach Ende des Moduls sind kennen die Teilnehmenden die Strukturen der objektorientierten Programmierung und können diese mit Hilfe der Programmiersprache Java implementieren. Ebenso kennen sie die grundlegenden Datenstrukturen, kennen die Standardoperationen der Datenstrukturen und können diese anwendungsgerecht einsetzen.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - können die Studierenden die grundlegenden Prinzipien oder objektorientierten Programmierung anwenden und auflisten. - kennen sie Java als plattformunabhängige und objektorientierte Programmiersprache. - Kennen sie Objektnetze mit Hilfe der objektorientierten Beziehungen implementieren können. - können die Studierenden Strategien und Anwendungsbereiche der Vererbung, Polymorphie und Interfaces anwenden und implementieren - können sie das Konzept von Metainformationen beschreiben und anwenden - kennen sie parametrisierte Typen und können sie implementieren - kennen die Studierenden Ausnahmebehandlung mit Exceptions und können sie anwenden - kennen sie abstrakte Datenstrukturen und können sie beschreiben - kennen sie Sortier- und Suchverfahren, die Unterschiede, mit Hinblick auf den Begriff der Effizienz, und können sie differenzieren und entsprechend den Anforderungen verwenden - kennen die Studierenden binäre Such- und ausgeglichene Bäume und können die Unterschiede auflisten und entsprechend den Anforderungen verwenden

Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	270 Stunden, davon <u>Präsenzzeit</u> : 240 Stunden <u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 30 Stunden
Leistungspunkte ECTS-Punkte / Credit Points (CP)	9 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	Kombinationsprüfung: Projektarbeit, Präsentation zur Projektaufgabe und mündliche Prüfung
Unterlagen (Skripte, Literatur, Programme usw.)	<ul style="list-style-type: none"> - Java SE 9-Standard-Bibliothek: Das Handbuch für Entwickler. Die zweite Insel, aktuell zu Java 9, Christian Ullenboom, ISBN-13: 978-3836258746, Rheinwerk Computing; Auflage 3 (27. November 2017) - Effective Java: Third Edition, Joshua Bloch, ISBN-13: 9780134685991, Addison-Wesley Professional; Auflage: 3. (27. Dezember 2017) - Algorithmen und Datenstrukturen – Eine Einführung mit Java, Gunter Saake, Kai-Uwe Sattler, ISBN-13: 978-3-86490-136-2, dpunkt Verlag, Auflage 5 (11/2013)
Lehrende	N. N. Prof. Dr. Anna Förster, Universität Bremen, Fachbereich 1: Physik / Elektrotechnik et al.

Modul 3

Modulkennzeichen	SWE-03
Modulbezeichnung (ggf. Untertitel)	Praktische Informatik
Englischer Titel	Practical Informatics
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Anna Förster Universität Bremen, Fachbereich 1: Physik / Elektrotechnik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Threads <ul style="list-style-type: none"> • Einführung in den Begriff des Multithreadings • Erzeugen, Verwalten und Synchronisation von Threads • Monitore und Semaphoren 2. Collections <ul style="list-style-type: none"> • Vector, Stacks und Queues • Hashtable, BitSet und Map • Immutable Collections und Iteratoren • Typisierte Klassen und generische Collections 3. Grafische Oberflächen mit JavaFX <ul style="list-style-type: none"> • Grundlagen der Grafikausgabe • Grafische Primitive • Event-Handling und Entwurfsmuster • Layoutmanager • Container & Menüs 4. Lambdas & Streams <ul style="list-style-type: none"> • Grundlagen der funktionalen Programmierung • Lambda-Ausdrücke • Bulk-Operations 5. I/O-Streams <ul style="list-style-type: none"> • Grundlagen von I/O-Streams • Character- und Bytestreams • Ein- und Ausgabe von Streams 6. Sockets <ul style="list-style-type: none"> • Grundlagen der Netzwerkprogrammierung • Protokolle, Adressierung und Ports • Sockets als plattformübergreifende Netzwerkschnittstelle • Austausch von Daten mittels Sockets

	<ul style="list-style-type: none"> • Serverdienste mittels Threads
Lernergebnisse / Kompetenzen (Learning Outcome)	<p>Ziel des Moduls ist es einen tieferen Einblick in die Informatik zu vermitteln. Dabei setzen sich die Teilnehmenden mit den Konzepten der Nebenläufigkeit, den Lambdas, den (IO)-Streams und Sockets auseinander und lernen grafische Oberfläche mit Hilfe von JavaFX zu entwickeln.</p> <p>Durch die Bearbeitung dieses Moduls</p> <ul style="list-style-type: none"> - Verstehen die Studierenden das Konzept der Nebenläufigkeit und können es anwenden - kennen sie das Java-Collections-Framework und können anwendungsbezogen passende Datenstrukturen auswählen und anwenden - können die Studierenden grafische Oberflächen entwerfen und mittels von JavaFX implementieren - kennen sie das Konzept von Lambdas und (IO)-Streams, verstehen es und können es anwenden - verstehen die Studierenden Sockets als plattformübergreifendes Konzept und können es implementieren
Lehr- und Lern-Formen	<p>Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium</p>
Workload Berechnung des Arbeitsaufwands	<p>180 Stunden, davon <u>Präsenzzeit</u>: 160 Stunden <u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 20 Stunden</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	<p>6 CP nach ECTS (European Credit Transfer and Accumulation System)</p>
Prüfungsart Prüfungsform / Prüfungsdauer	<p>Kombinationsprüfung: Projektarbeit und mündliche Prüfung</p>
Unterlagen (Skripte, Literatur, Programme usw.)	
Lehrende	<p>N. N. Prof. Dr. Anna Förster, Universität Bremen, Fachbereich 1: Physik / Elektrotechnik et al.</p>

Modul 4

Modulkennzeichen	SWE-04
Modulbezeichnung (ggf. Untertitel)	Data Science mit Python
Englischer Titel	Data Science with Python
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Anna Förster Universität Bremen, Fachbereich 1: Physik / Elektrotechnik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Einführung in Data Science mit Python <ul style="list-style-type: none"> • Grundlagen des Begriffs „Data Science“ • Grundelemente der Programmiersprache Python 2. Statistische Methoden 3. NumPy, Visualisierung von Daten <ul style="list-style-type: none"> • Balken- und Liniendiagramme • Scatterplots 4. Maschinelles Lernen <ul style="list-style-type: none"> • Grundlagen des maschinellen Lernens • Overfittung und Underfittung • Genauigkeit • Bias vs. Varianz • Extraktion und Auswahl 5. Regressionsanalyse <ul style="list-style-type: none"> • Grundlagen der Regressionsanalyse • Lineare, multiple und logistische Regression
Lernergebnisse / Kompetenzen (Learning Outcome)	<p>In diesem Modul lernen die Studierenden, welche Methoden zur Verarbeiten von großen Datenmengen zur Verfügung stehen und wie diese angewendet werden. Dazu lernen sie die Grundlagen des maschinellen Lernens, wie diese mit Hilfe der Programmiersprache Python umgesetzt werden und erstellen im Laufe des Moduls ein eigenes Übungsprojekt.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - kennen die Studierenden die Elemente der Programmiersprache Python mit Hinblick auf die Thematik Data Science und können sie anwenden - wissen sie, wie Daten visualisiert werden

	- kennen sie die Konzepte des maschinellen Lernens
Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	90 Stunden, davon <u>Präsenzzeit</u> : 80 Stunden <u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 10 Stunden
Leistungspunkte ECTS-Punkte / Credit Points (CP)	3 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	Modulprüfung: Projektbericht
Unterlagen (Skripte, Literatur, Programme usw.)	Grus, Joel: Einführung in Data Science. Grundprinzipien der Datenanalyse mit Python. (Übersetzung aus dem Amerikanischen von Kristian Rother.) Heidelberg: O'Reilly Verlag, März 2016.
Lehrende	N. N. Prof. Dr. Anna Förster, Universität Bremen, Fachbereich 1: Physik / Elektrotechnik et al.

Modul 5

Modulkennzeichen	SWE-05
Modulbezeichnung (ggf. Untertitel)	Datenbanken
Englischer Titel	Databases
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr.-Ing. Heide-Rose Vatterrott Hochschule Bremen, Fakultät 4: Elektrotechnik und Informatik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Datenmodelle und Datenmodellierung <ul style="list-style-type: none"> • Methoden der Datenmodellierung • Techniken des Entwurfs von Datenbanksystemen • Einsatz von Datenbanksystemen 2. Relationale Datenbanken und SQL <ul style="list-style-type: none"> • Aufbau relationaler Datenbanksysteme • SQL-Grundlagen • Datendefinition und -manipulation mit SQL 3. Datenintegrität und Datenschutz <ul style="list-style-type: none"> • Referentielle Integrität • Transaktionsmechanismus • Authentifizierung und Autorisierung 4. Programmierschnittstellen (JDBC) <ul style="list-style-type: none"> • Die JDBC-API • Arbeiten mit ResultSets • Prepared Statements • ConnectionPooling
Lernergebnisse / Kompetenzen (Learning Outcome)	Die Studierenden kennen die grundsätzlichen Paradigmen beim Entwurf und der Manipulation von relationalen Datenbanksystemen. Es können Fragen nach den Komponenten eines relationalen Datenbankmanagementsystem beantwortet werden, welche Techniken und Konzepte beim Entwurf zur Anwendung gelangen, wie mit Hilfe der Structured Query Language (SQL) relationale Datenbanken erstellt, abgefragt sowie manipuliert werden können und wie die Integrität der Daten sichergestellt werden kann. Mit der Programmiersprache Java und der Datenbankschnittstelle (JDBC) können die Teilnehmenden aus einem Java-Programm heraus auf die Datenbank zugreifen und die Datenhaltung im Kontext einer Applikation

	<p>einordnen.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - können die Studierenden die Komponenten eines relationalen Datenbankmanagementsystems benennen und beschreiben - kennen Techniken und Konzepte zum Entwurf von Datenbank und können sie anwenden - kennen die Spezifika der SQL und können sie anwenden - können Konzepte der Datenintegrität und des Datenschutzes erläutern, benennen und anwenden - kennen die Programmierschnittstelle JDBC und können den spezifischen Code in Java für die Programmierschnittstelle implementieren - können aus Vorgaben die Anforderungen für eine Datenbankmodell analysieren, implementieren und die entstanden Lösung hinsichtlich der erfüllten Anforderungen bewerten
Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	<p>90 Stunden, davon</p> <p><u>Präsenzzeit</u>: 80 Stunden</p> <p><u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 10 Stunden</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	3 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	<p>Kombinationsprüfung:</p> <p>i.d.R. Bearbeitung von Übungsaufgaben und mündliche Prüfung oder Klausur</p>
Unterlagen (Skripte, Literatur, Programme usw.)	Elmasri, Ramez A. / Navathe, Shamkant B.: Grundlagen von Datenbanksystemen. Bachelorausgabe (3., aktualisierte Auflage) Hallbergmoos: Pearson, 2009.
Lehrende	<p>N. N.</p> <p>Prof. Dr.-Ing. Heide-Rose Vatterrott Hochschule Bremen, Fakultät 4: Elektrotechnik und Informatik et al.</p>

Modul 6

Modulkennzeichen	SWE-06
Modulbezeichnung (ggf. Untertitel)	Webentwicklung
Englischer Titel	Web Development
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr.-Ing. Heide-Rose Vatterrott Hochschule Bremen, Fakultät 4: Elektrotechnik und Informatik
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<ol style="list-style-type: none"> 1. Standards in der Webentwicklung <ul style="list-style-type: none"> • Architekturprinzipien • HTTP/S Protokoll • Client/Server Modelle 2. HTML 5 & CSS 3 <ul style="list-style-type: none"> • Grundlagen von Beschreibungssprachen • Grundelemente von HTML5 • Grundelemente von CSS3 • Document Object Model 3. Einführung in JavaScript <ul style="list-style-type: none"> • Besonderheiten von dynamisch typisierten und deklarativen Programmiersprachen • Werte und Variablen • Funktionen • Datenstrukturen • Funktionale und objektorientierte Programmierung • Reguläre Ausdrücke 4. Übungsprojekt: Mobile Application Development mit Hilfe eines Frameworks zur Erstellung von Hybrid- und Progressive Web Apps
Lernergebnisse / Kompetenzen (Learning Outcome)	<p>In diesem Modul lernen die Studierenden die grundlegenden Techniken und Architekturen, welche bei der Entwicklung im Web und bei mobilen Anwendungen anzuwenden sind. Dabei werden zunächst die grundsätzlichen Architekturprinzipien beleuchtet, Elemente der Beschreibungssprachen HTML 5 & CSS3 vermittelt und in die Studierenden in die Programmiersprache JavaScript eingeführt.</p> <p>Ein weiter Bestandteil dieses Moduls ist ein Projekt, welches die Studierende gemeinsam im Team, unter vorge-</p>

	<p>gebenen Rahmenbedingungen, erarbeiten sollen. Dabei sollen modulübergreifenden Kompetenzen aus dem Bereich der Programmierung zum Einsatz kommen. Ebenso soll die Studierenden vor die Herausforderung gestellt werden, unbekannte Begriffe und Technologien sich weitestgehend selbständig anzueignen, zu bewerten und zu implementieren.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - kennen die Studierenden die Architekturprinzipien von Client-Server-Modell und können sie beschreiben - können sie die Komponenten des HTTP/S-Protokolls skizzieren - kennen sie die Funktionsweisen von HTML5 & CSS 3 und können einfache Weboberflächen implementieren - kennen die Studierenden die Grundlagen von JavaScript und können sie implementieren - können sie unbekannte Begriffe und Technologien erarbeiten und bewerten und implementieren - können sie im Team ein Webprojekt nach Vorgaben planen und implementieren
Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	<p>180 Stunden, davon</p> <p><u>Präsenzzeit</u>: 160 Stunden</p> <p><u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 20 Stunden</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	6 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	<p>Kombinationsprüfung:</p> <p>Projektarbeit und mündliche Prüfung</p>
Unterlagen (Skripte, Literatur, Programme usw.)	<ul style="list-style-type: none"> - Die Kunst der JavaScript-Programmierung: Eine moderne Einführung in die Sprache des Web, Marijn Haverbeke, ISBN-13: 978-3898647878, dpunkt.verlag GmbH; Auflage: 1 (19. Dezember 2011) - Angular: Das Praxisbuch zu Grundlagen und Best Practices, Manfred Steyer, Daniel Schwab, ISBN-13 978-3960090267, O'Reilly; Auflage: akt. u. überarb. (28. August 2017)
Lehrende	<p>N. N.</p> <p>Prof. Dr.-Ing. Heide-Rose Vatterrott Hochschule Bremen, Fakultät 4: Elektrotechnik und Informatik et al.</p>

Modul 7

Modulkennzeichen	SWE-07
Modulbezeichnung (ggf. Untertitel)	Softwaretechnik I
Englischer Titel	Software Engineering (I)
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Rainer Koschke Universität Bremen, Fachbereich 3, Technologie-Zentrum Informatik und Informationstechnik (TZI)
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtsprache	Deutsch
Lern-Inhalte (Contents of the course)	<p>Dieses Modul liefert eine Einführung in die zentralen Begriffe und Konzepte der Softwaretechnik und stellt die Probleme der Softwareentwicklung kunden- und praxisorientiert dar.</p> <ol style="list-style-type: none"> 1. Einführung und Grundlagen <ul style="list-style-type: none"> • Was Software ist und welche Eigenschaften sie hat • Der Einfluss von Lehmann und Beladys Software-Gesetzen • Softwaretechnik als Ingenieursdisziplin 2. Vorgehens- und Prozessmodelle <ul style="list-style-type: none"> • Arten von Vorgehens- und Prozessmodellen • Agile Methoden • SCRUM, DevOps, Continuous Delivery 3. Projektmanagement <ul style="list-style-type: none"> • Stellungwert von Planungen • Bestandteile von Softwareprojekten • Arten von Projektplänen • Identifikation und Gliederung von Aufgaben • Risikomanagement 4. Objektorientierte Modellierung mit der UML <ul style="list-style-type: none"> • Bedeutung von der Modellierung • Diagrammtypen und Einsatzvarianten • Notationselemente 5. Anforderungsanalyse <ul style="list-style-type: none"> • Schritte der Anforderungsanalyse • Schritte der Ist-Analyse • Erhebungstechniken 6. Anforderungsspezifikation <ul style="list-style-type: none"> • Bedeutung und Eigenschaften • Gliederung und Kriterien

	<ul style="list-style-type: none"> • Prozesse der Anforderungsspezifikation • Prüfung der Anforderungsspezifikation <p>7. Testverfahren</p> <ul style="list-style-type: none"> • Testarten und -varianten • Testabdeckungsmaße • Testvorbereitung,-durchführung und -protokollierung <p>8. Änderungs- und Konfigurationsmanagement</p> <ul style="list-style-type: none"> • Wartung, Evolution und Reengineering • Bedeutung der Softwarewartung • Änderungsprozesse • Werkzeuge für das Konfigurationsmanagement <p>9. Implementierung</p> <ul style="list-style-type: none"> • Programmiersprachen und -richtlinien • Code-Qualität und Metriken • Vermeidung von Code-Redundanz • Entwicklungsumgebungen
<p>Lernergebnisse / Kompetenzen (Learning Outcome)</p>	<p>Lernziel ist es, die Kompetenzen zu erlangen, die notwendig sind, um Softwareprojekte erfolgreich zu meistern, und aufzuzeigen, wie man sie idealer Weise durchführt, um die Kosten gering zu halten. Nach dem Modul kennen die Teilnehmer/innen die wichtigsten Gangarten moderner Softwareentwicklung und sind in der Lage, sie im Kanon moderner Verfahrensweisen zu identifizieren.</p> <p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - verstehen die Studierenden die Eigenschaften des abstrakten Produkts „Software“ kennen und die daraus folgenden Besonderheiten verstehen - besitzen sie das Verständnis, was Vorgehens- und was Prozessmodelle sind, die Vor- und Nachteile dieser zu kennen und, abhängig von den Rahmenbedingungen, das richtige auszuwählen - können sie mit Hilfe des Projektmanagements ein Softwareprojekt planen und organisieren - kennen die Studierenden die Begriffe der Teamsoziologie - können sie mit Hilfe der UML Ergebnisse aus einer objektorientierten Analyse in eine allgemeine Notation überführen. Insbesondere haben sie die Kompetenz, mit welchem Diagramm welcher Sachverhalt dargestellt wird - können sie den Begriff der Qualität in den Kontext der Softwaretechnik einordnen und Metriken erstellen, mit welchen Qualitätsaspekte einer Software gemessen werden können - kennen sie die Bedeutung der Analyse, Herausforderungen, beteiligte Akteure und Aktivitäten der Analysephase - können sie Inhalt und Gegenstand des Ist-Zustandes, mit den geeigneten Methoden, selbstständig ermitteln und kennen die Abgrenzung gegenüber der Soll-Analyse - wissen sie, wie die Anforderungsspezifikation inhaltlich und strukturell organisiert ist - können sie Testarten- und -varianten kennen, be-

	<p>schreiben und anwenden</p> <ul style="list-style-type: none"> - kennen sie die Bedeutung des Änderungs- und Konfigurationsmanagement und können Werkzeuge zum Management anwenden
Lehr- und Lern-Formen	Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium
Workload Berechnung des Arbeitsaufwands	<p>270 Stunden, davon</p> <p><u>Präsenzzeit</u>: 240 Stunden</p> <p><u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 30 Stunden</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	9 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	<p>Kombinationsprüfung:</p> <p>Projektarbeit, Vortrag zur Projektarbeit, mündliche Prüfung oder Klausur</p>
Unterlagen (Skripte, Literatur, Programme usw.)	<ul style="list-style-type: none"> - Analyse und Design mit der UML 2.5: Objektorientierte Softwareentwicklung, Bernd Oestereich, Axel Scheithauer, Stefan Bremer, ISBN: 978- 3486721409, 11. Auflage, Oldenbourg Verlag - 26. September 2013 - Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering, Helmut Balzert, ISBN-13: 978-3827417053, Spektrum Akademischer Verlag; Auflage: 3. (17. September 2009" - UML 2 glasklar: Praxiswissen für die UML-Modellierung, Christine Rupp et al, ISBN: 978-3446430570, Carl Hanser Verlag GmbH & Co. KG; Auflage: 4., aktualisierte und erweiterte (3. April 2012)
Lehrende	<p>N. N.</p> <p>Prof. Dr. Rainer Koschke Universität Bremen, Fachbereich 3, Technologie-Zentrum Informatik und Informationstechnik (TZI)</p> <p>et al.</p>

Modul 8

Modulkennzeichen	SWE-08
Modulbezeichnung (ggf. Untertitel)	Softwaretechnik II
Englischer Titel	Software Engineering (II)
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	Prof. Dr. Rainer Koschke Universität Bremen, Fachbereich 3, Technologie-Zentrum Informatik und Informationstechnik (TZI)
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtssprache	Deutsch
Lern-Inhalte (Contents of the course)	<p>Die Inhalte dieses Moduls erweitern die Prozesse der Softwareentwicklung und beschäftigen sich mit den Aspekten des Entwurfs einer Architektur und Benutzerschnittstelle. Dabei wird den Fragen nachgegangen, was sich hinter dem Begriff der Softwarearchitektur verbirgt, was eine Modularisierung ist und welche Kriterien und Richtlinien beim Entwurf einer Architektur zu beachten sind, was Architekturstile sind und welche Rolle die Softwareergonomie spielt.</p> <p>Ein weiter Bestandteil dieses Moduls ist ein modulübergreifendes Softwareprojekt, bei welchem die Studierenden möglichst alle Aspekte bei der Entwicklung und Programmierung eines Softwaresystems anwenden sollen.</p> <ol style="list-style-type: none"> 1. Softwarearchitektur <ul style="list-style-type: none"> • Was ist Softwarearchitektur? • Views und Viewpoints • Einflussfaktoren • Architekturstile • Entwurfsmuster • Modularisierung • Architekturreview 2. Benutzerschnittstellenentwurf <ul style="list-style-type: none"> • Softwareergonomie: Aspekte und Qualität • Interaktionsformen und -mittel • Werkzeuge • Usability-Evaluationsverfahren 3. Dokumentation <ul style="list-style-type: none"> • interne Software-Dokumentation • Benutzungshandbücher und Online-Hilfen

	<p>4. Softwareprojekt</p> <ul style="list-style-type: none"> • Analyse, Entwurf, Test, Implementierung und Dokumentation eines selbstorganisierten Softwareprojekts
<p>Lernergebnisse / Kompetenzen (Learning Outcome)</p>	<p>Durch die Bearbeitung dieses Moduls:</p> <ul style="list-style-type: none"> - kennen die Studierenden die zentralen Begriffe von Softwarearchitektur und können sie beschreiben und anwenden - kennen sie Arten von Softwarearchitekturen und können sie beschreiben - kennen sie Methoden zur Modularisierung und können sie anwenden - kennen die Studierenden Architekturstile und -muster und können sie beschreiben und anforderungsgerecht implementieren - können sie Benutzerschnittstellen entwerfen und evaluieren - kennen sie Methoden und Werkzeuge zur Erstellung eine Software-Dokumentation und können sie anwenden - können sie Benutzungshandbücher und Online-Hilfe erstellen - können sie ein begleitetes Softwareprojekt gemeinsam im Team durchführen
<p>Lehr- und Lern-Formen</p>	<p>Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie begleitendem Selbststudium</p>
<p>Workload Berechnung des Arbeitsaufwands</p>	<p>180 Stunden, davon <u>Präsenzzeit</u>: 160 Stunden <u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 20 Stunden</p>
<p>Leistungspunkte ECTS-Punkte / Credit Points (CP)</p>	<p>6 CP nach ECTS (European Credit Transfer and Accumulation System)</p>
<p>Prüfungsart Prüfungsform / Prüfungsdauer</p>	<p>Kombinationsprüfung: Projektarbeit, Vortrag zur Projektarbeit, mündliche Prüfung oder Klausur</p>
<p>Unterlagen (Skripte, Literatur, Programme usw.)</p>	<p>Vogel, Oliver / Arnold, Ingo / Chughtai, Arif / Ihler, Edmund / Kehrer, Timo / Mehlig, Uwe / Zdun, Uwe: Software-Architektur. Grundlagen – Konzepte – Praxis. (2. Auflage) Heidelberg: Spektrum Akademischer Verlag, 2009.</p> <p>Eilebrecht, Karl / Starke, Gernot: Patterns kompakt. Entwurfsmuster für effektive Softwareentwicklung (5., vollständig aktualisierte Auflage) Springer Vieweg, 2019.</p>

Lehrende	N. N. Prof. Dr. Rainer Koschke Universität Bremen, Fachbereich 3, Technologie-Zentrum Informatik und Informationstechnik (TZI) et al.
-----------------	---

Modul 9

Modulkennzeichen	SWE-09
Modulbezeichnung (ggf. Untertitel)	Empowerment und individuelle Profilierung
Englischer Titel	Empowerment and Profiling
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Teilmodule	<u>Teilmodul 9-1</u> Präsentation & Moderation <u>Teilmodul 9-2</u> Teamarbeit & Kreativität <u>Teilmodul 9-3</u> Bewerbungstraining & Karrierecoaching
Modulverantwortung	N. N.
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtssprache	deutsch
Lern-Inhalte (Contents of the course)	<p>In diesem Modul sollen die Teilnehmenden ihre Stärken und Potenziale entdecken, ihre Selbstpräsentation und Kommunikation verbessern und gleichzeitig die individuelle Kompetenz zur eigenverantwortlichen Planung und Steuerung ihrer beruflichen Weiterentwicklung ausbauen.</p> <p>+++++</p> <p>Teil 9-1: Präsentation und Moderation</p> <p><i>Präsentations-, Vortrags- und Moderationstechniken</i></p> <p>Die Anwendung von wirkungsvollen Präsentations-, Vortrags- und Moderationstechniken mit rhetorischem Geschick, sind weitere Bestandteile dieses Teilmoduls. Da der Erfolg einer Präsentation häufig über Aufträge, Projekte und Karrieren entscheidet, dürfen diese Punkte nicht vernachlässigt werden. Die Teilnehmer/innen sollen dabei lernen, welche Techniken zur Verfügung stehen, wie diese richtig angewendet und professionell miteinander kombiniert werden, um so den Vortrag, die Rede oder Präsentation überzeugend halten und souverän moderieren zu können.</p> <p>Inhalte:</p> <ol style="list-style-type: none"> 1. Präsentations- und Vortragstechniken <ul style="list-style-type: none"> • Aufarbeitung und Visualisierung von Datenmaterial • Vortragstechniken

	<ul style="list-style-type: none"> • Medieneinsatz <p>2. Professionelle Moderation</p> <ul style="list-style-type: none"> • Rhetorik und Körpersprache • Moderationsstile • Umgang mit Störungen und Konflikten • Erprobung der Techniken im Rahmen von Einzel- und/oder Gruppenpräsentationen <p>+++++</p> <p>Teil 9-2: Teamarbeit & Kreativität</p> <p>Die Teilnehmer/innen lernen und verstehen, was die Begriffe Kreativität und Innovation bedeuten und wie diese gezielt in den alltäglichen Arbeitsablauf angewendet werden. In weiterem Verlauf werden kommunikative Kompetenzen aufgebaut, die es ermöglichen, im beruflichen Kontext (Gesprächs-) Ziele zu erreichen. Dabei werden verschiedene Kommunikationsmodelle und -ebenen sowie deren Einsatzfelder praxisrelevant dargestellt.</p> <p>Inhalte:</p> <p>1. Einführung</p> <ul style="list-style-type: none"> • Grundlagen- und Begriffsbestimmung • Vor- und Nachteile von Teamarbeit • Team- und Projektkultur • Prozessmodell der Teamarbeit <p>2. Aspekte der Teamentwicklung</p> <ul style="list-style-type: none"> • Teamfähigkeit - Remote Teams und virtuelle Teams • Teamkommunikation: Teammeetings effektiv gestalten • Stufen der Teamentwicklung (Phasenmodelle) • Besondere Teameffekte (Groupthink, Social Loafing, Risk shifting) • Teamzusammensetzung und Rollen im Team <p>3. Gesprächstechniken</p> <ul style="list-style-type: none"> • Kommunikationsmodelle • Gesprächstechniken • Körpersprache • Gesprächssituationen: Mitarbeitergespräche, Zielvereinbarungen, Konfliktgespräche, Aufgabengespräche, Kundengespräche etc. <p>4. Was ist Kreativität?</p> <ul style="list-style-type: none"> • Problemlösung und Entstehung von Neuem • Hemmung und Förderung von Kreativität • Kreativität im Team vs. Kreativität des Individuums • Die Entwicklung von Ideen <p>5. Der kreative Prozess im Team</p> <ul style="list-style-type: none"> • Phasen des kreativen Prozesses: Präparation, Inkubation, Illumination und Verifikation • Methoden zur konzeptionellen Entwicklung von Ideen
--	--

	<p>6. Kreativitätstechniken</p> <ul style="list-style-type: none"> • Übersicht, Einteilung und Grundregeln für die Anwendung • Assoziations-Techniken, Analogietechniken, Konfrontationstechniken • Analytische (oder diskursive) Techniken • Mapping-Techniken <p>+++++</p> <p>Teil 9-3: Bewerbungstraining & Karrierecoaching</p> <p>Durch die Analyse eines branchenübergreifenden und sehr heterogenen Arbeitsmarkts, mit Firmenportraits, Tipps zur individuellen Erschließung einzelner Tätigkeitsfelder und dem neuen Wissen aus den Fachmodulen der Weiterbildung sind die Teilnehmenden aufgefordert, ihr individuelles Kompetenzprofil kontinuierlich fortzuschreiben. Bereits im Modulverlauf werden sie sich bei ausgewählten Unternehmen bewerben.</p> <p>Inhalte:</p> <ol style="list-style-type: none"> 1. Gewappnet für den Einstieg, Umstieg und Aufstieg <ul style="list-style-type: none"> • Persönliche Standortbestimmung und Kompetenzdokumentation • Selbstmanagement und Selbstmarketing • Strategieplanung 2. Arbeitsmarkt und Branchenstruktur <ul style="list-style-type: none"> • Branchenkenntnis erwerben • Arbeitsmarkt analysieren 3. Optimierung der Bewerbungsunterlagen Stunden <ul style="list-style-type: none"> • Berufsbild kennenlernen • Bewerbungsfotos erneuern • spezifische Bewerbungsunterlagen erstellen 4. Bewerbungs- und Karrierecoaching Stunden <ul style="list-style-type: none"> • Vorstellungs- und Bewerbungsgespräche führen • Individual-Coaching
<p>Lernergebnisse / Kompetenzen (Learning Outcome)</p>	<p>Die Studierenden können:</p> <ul style="list-style-type: none"> – (sich) wirksam präsentieren, ihre Kompetenzen überzeugend darstellen sowie ihre beruflichen Ziele charakterisieren – den Arbeitsmarkt entsprechend ihrer beruflichen Zielplanung analysieren – Bewerbungsstrategien entwickeln und ihre Leistungen wirksam auf dem Arbeitsmarkt offerieren.
<p>Lehr- und Lern-Formen</p>	<p>Präsenzveranstaltungen mit Vortrag, Diskussion und Übungen sowie ergänzendem E-Learning und Selbststudium</p>
<p>Workload</p>	<p>180 Stunden, davon</p>

Berechnung des Arbeitsaufwands	<p>Präsenzzeit: 160 Stunden</p> <p><u>Angeleitetes Selbststudium</u> (inkl. Prüfungsvorbereitung): 20 Stunden</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	6 CP nach ECTS (European Credit Transfer and Accumulation System)
Prüfungsart Prüfungsform / Prüfungsdauer	<p>Teilprüfungen:</p> <p>bestehend aus drei Studienleistungen</p>
Unterlagen (Skripte, Literatur, Programme usw.)	<p>Püttjer, Christian / Schnierda, Uwe: Perfekte Bewerbungsunterlagen für Hochschulabsolventen. Erfolgreich zum Traumjob – auch für Onlinebewerbungen. Bachelor – Master – Diplom – Magister – Staatsexamen – Promotion. [2004] (9. aktualisierte Auflage) Frankfurt/Main: Campus Verlag, 2014.</p>
Lehrende	N. N.

Modul 10

Modulkennzeichen	SWE-10
Modulbezeichnung (ggf. Untertitel)	Betriebliches Praxisprojekt
Englischer Titel	Internship Project
Zuordnung zum Curriculum	„Softwareentwicklung und Software Engineering“ (SWE)
Modulverantwortung	N. N.
Modulart	Pflichtmodul
Häufigkeit	einmal pro Weiterbildungsstudiengang
Unterrichtssprache	Deutsch
Lern-Inhalte (Contents of the course)	(Siehe Praktikumsvertrag)
Lernergebnisse / Kompetenzen (Learning Outcome)	<p>Mit der Durchführung des Praxisprojekts haben die Teilnehmer/-innen ihr Kompetenzprofil gestärkt:</p> <p>Sie haben vertiefte Kenntnisse über Organisation und Arbeitsweise des Berufs- bzw. Tätigkeitsfelds.</p> <p>Sie können die erworbenen Fachkenntnisse sowie Methoden- und Sozialkompetenzen in Realsituation anwenden.</p> <p>Sie haben ihre Kompetenzen anhand einer praktischen Fragestellung aus einem Unternehmen erprobt und ihr Wissen aus dem Weiterbildungsprogramm vertieft.</p>
Lehr- und Lern-Formen	Projektpraktikum (3 Monate) in einer Unternehmung (Betrieb, Organisation, Institution, Einrichtung etc.) begleitet von Präsenztagen in der Universität
Workload Berechnung des Arbeitsaufwands	<p>180 Stunden, davon</p> <p><u>Präsenzzeit</u>: 30 Stunden</p> <p><u>Selbststudium</u> (inkl. Prüfungsvorbereitung): 150 Stunden</p> <p>Durchführung des Praxisprojekts und Erstellung eines Projektberichts (140 Stunden)</p> <p>Vorbereitung der Prüfungspräsentation (10 Stunden).</p>
Leistungspunkte ECTS-Punkte / Credit Points (CP)	6 CP nach ECTS (European Credit Transfer and Accumulation System)

<p>Prüfungsart Prüfungsform / Prüfungsdauer</p>	<p>Kombinationsprüfung: bestehend aus drei Studienleistung, i.e. Praxisprojekt, Projektbericht sowie Präsentation mit anschließendem Fachgespräch (Dauer: ca. 30 Min.)</p>
<p>Unterlagen (Skripte, Literatur, Programme usw.)</p>	<p>(keine)</p>
<p>Lehrende</p>	<p><u>Fachbetreuung im Unternehmen:</u> N. N. (Fachexpertin/Fachexperte – siehe Praktikumsvertrag)</p> <p><u>Lernprozessbegleitung durch Akademie für Weiterbildung:</u> N. N. (in Kooperation mit den Modulverantwortlichen des Weiterbildenden Studiums SWE)</p>