

Informatik

Informationen zu den
Studieninhalten der Module



**Hochschule
Hof**

University of
Applied Sciences



Sicher bist du schon gespannt, was dich im Informatik-Studium an der Hochschule Hof erwartet. Diese Informationsbroschüre mit Beschreibungen der Lehrveranstaltungen soll dir davon einen ersten Eindruck vermitteln.






































Zum besseren Verständnis des Studienablaufs eine kleine Info vorab. Das Studium ist bei uns in 3 Abschnitte gegliedert (einen Überblick findest du auf der Folgeseite):



- In den Semestern 1 – 4 vermitteln wir dir neben den notwendigen Grundlagen auch die wichtigsten für den Studiengang profilbildenden Kompetenzen. Damit bist du für den heutigen Arbeitsmarkt bereits bestens gerüstet. Die Studieninhalte in den ersten vier Semestern sind dabei für alle Studierenden gleich.
- Die Semester 5 - 6 dienen der Vertiefung und Spezialisierung in genau den Bereichen, die dich besonders interessieren. Hier darfst du dir daher deinen Studienplan aus unserem umfangreichen Fächerkatalog selbst zusammenstellen.
- Im 7. Semester kannst du schließlich zum einen dein gewonnenes Wissen in einem spannenden Projekt zum Einsatz bringen und zum anderen deine Bachelorarbeit anfertigen.


Und nun viel Spaß beim Stöbern durch deinen vielleicht ja zukünftigen Studiengang!

Falls du noch weitere Fragen hast, stehen dir sowohl der Leiter des Studiengangs als auch unsere Studiengangbotschafter gerne als Ansprechpartner zur Verfügung.

Ablauf des Studiums Informatik

1	 Datenbanken	 Betriebssysteme	 Grundlagen der Programmierung	 Grundlagen der Informationstechnik	 Diskrete Mathematik	 Erfolgreich im Studium
2	 Software Engineering	 Rechnernetze	 Grundlagen Web Development	 Algorithmen und Datenstrukturen	 Statistik	 Gestaltung, Kommunikation und Präsentation
3	 Software-System-Entwicklung	 Angewandte KI	 Full Stack Web Development	 IT-Sicherheit	 Mathematik für Informatiker	 Englisch für Informatiker
4	 Fortgeschrittene Programmier-techniken	 Fortgeschrittene Konzepte der KI	 Cloud Computing	 Data Science	 Effizientes Programmieren mit C/C++	 Digitaletik
5	 Testverfahren für komplexe Softwaresysteme	 Internet of Things	 Robotik	 Angewandtes maschinelles Lernen	 Moderne App- und Webentwicklung	 Software-Architektur
6	 Datenanalyse und Data Mining	 RESTful Web Services	 Software Reverse Engineering	 Software-Qualitätsmanagement	 Kryptologie	 Interdisziplinäre Software-Entwicklung
7	 Praxissemester mit Projektarbeit und Bachelorarbeit					

 Informatik Kompetenzen
 Allgemeine Kompetenzen

 Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Inhalt

1. Semester

Datenbanken	6
Betriebssysteme	7
Grundlagen der Programmierung	8
Grundlagen der Informationstechnik	9
Diskrete Mathematik	10
Erfolgreich im Studium	11

2. Semester

Software Engineering	13
Rechnernetze	14
Grundlagen Web Development	15
Algorithmen und Datenstrukturen	16
Statistik	17
Gestaltung, Kommunikation und Präsentation	18

3. Semester

Software-System-Entwicklung	20
Angewandte KI	21
Full Stack Web Development	22
IT-Sicherheit	23
Mathematik für Informatiker	24
Englisch für Informatiker	25

4. Semester

Fortgeschrittene Programmieretechniken	27
Fortgeschrittene Konzepte der KI	28
Cloud Computing	29
Data Science	30
Effizientes Programmieren mit C/C++	31
Digialethik	32

5. + 6. Semester^w

Software-Architektur	34
Interdisziplinäres Software- entwicklungsprojekt	35
Testverfahren für komplexe Software-Systeme ^w	36
Internet of Things ^w	37
Robotik ^w	38
Angewandtes maschinelles Lernen ^w	39
Moderne App- und Webentwicklung ^w	40
Datenanalyse und Data Mining ^w	41
RESTful Web Services ^w	42
Kryptologie ^w	43

7. Semester

Praxissemester mit Projektarbeit und Bachelorarbeit	45
--	----

^w Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

1. Semester

The image features a minimalist, abstract design. A large yellow square occupies the top-left and top-right portions of the frame. A white square is positioned in the middle-left area, overlapping the yellow one. To the right of the white square, there is a vertical stack of three overlapping squares: a blue square on top, a dark blue square in the middle, and a green square at the bottom. The overall composition is clean and modern, with a focus on geometric shapes and a limited color palette.

Datenbanken

Die Studierenden kennen die grundlegende Theorie der Datenbanksysteme und gewinnen einen Überblick über praktische Entwurfsverfahren relationaler Datenbankmanagementsysteme. Sie beherrschen die Datenbanksprache SQL (Structured Query Language) und haben ein Grundwissen zur Entwicklung von Datenbankanwendungen.

Studieninhalte

- Grundlegende Begriffe der Datenbanktechnologie
- Datenbankmodelle
- Datenbankentwurfstechniken
- Verwendung von Tools
- Normalisierung des Datenbankentwurfs
- Structured Query Language
- Transaktionskonzepte

Betriebssysteme

Die Studierenden kennen die allgemeinen Grundfunktionen von Betriebssystemen und können Betriebssysteme nach verschiedenen Kriterien und Einsatzbereichen klassifizieren. Konzepte des Betriebssystems Unix / Linux werden theoretisch und praktisch eingeführt. Die Studierenden sollen lernen auf der Kommandozeilenebene Shell-Skripte zu verstehen und zu schreiben. Die Grundlagen, der Aufbau und der Betrieb von einfachen Client-Serverumgebungen sollen beherrscht werden. Die hierzu notwendigen Netzwerk- und Betriebssystemdienste und Konzepte sollen verstanden und in Betrieb genommen werden.

Studieninhalte

- Klassifizierung von Betriebssystemen
- Allgemeiner Aufbau und Funktion von Betriebssystemen und Betriebssystemkernen
- Konzepte des Betriebssystems Unix / Linux
- Unix-Shell-Programmierung
- Grundlagen, Konfiguration und Betrieb von Betriebssystem- und Netzwerkdiensten, z. B. NTP, DHCP, DNS, LDAP, SNMP, SSH, CRON, HTTP, DBMS, SYSLOG

Grundlagen der Programmierung

Die Studierenden erwerben grundlegende Kenntnisse der objektorientierten Programmierung mit Hilfe der Programmiersprache Java. Sie können Konsolen-Programme selbst entwickeln, vorgegebene Programme verstehen und Ideen daraus in eigene, kleinere Projekte einfließen lassen.

Studieninhalte

- Datentypen und Operatoren
- Kontrollstrukturen
- Referenzen
- Arrays
- Klassen und Objekte
- Vererbung
- Polymorphismus
- Packages und Sichtbarkeitsregeln
- Ausnahmebehandlung
- Bibliotheksklassen (String, Math, usw.)
- Ein- und Ausgabe
- Dateioperationen

Grundlagen der Informationstechnik

Die Studierenden kennen die Prinzipien digitaler Informationsverarbeitung und verstehen die Funktionsweise von Rechnern. Sie wissen wie Zahlen, Texte und multimediale Daten im Computer repräsentiert werden und können die grundlegenden Abläufe zur Verarbeitung von Daten erklären.

Studieninhalte

- Grundbegriffe der Informationstechnik
- Funktionsweise von Rechnern
- Binärdaten, digitale Kodierung von Integer und Gleitkommazahlen
- Zeichenkodierungen, z.B. ASCII, Unicode
- Markupssprachen, z.B. XML
- Digitale Kodierung von Multimediainhalten
- Automatentheorie, Schaltnetze, KV-Diagramme
- von-Neumann-Architektur, CPU-Aufbau
- Maschinenbefehle, Assembler, Register, Speicheradressierung
- Funktionseinheiten von Rechnern: Hardware-Komponenten und Peripherie
- Systemsoftware, z.B. BIOS/UEFI, Grundlagen der Betriebssysteme, Treiber, Prozesse

Diskrete Mathematik

Die Studierenden kennen und verstehen grundlegende mathematische Ideen und Konzepte, die der Informatik zugrunde liegen. Sie beherrschen die Grundzüge der Formulierung von konkreten Fragestellungen mit Hilfe einer abstrakten mathematischen Beschreibungssprache und sind in der Lage, geeignete Problemstellungen als mathematische Aufgabenstellung zu formulieren und zu lösen.

Studieninhalte

- Zahlen und Zahlentheorie
- Mengen
- Relationen und Abbildungen
- Folgen und Reihen
- Logik und Beweise
- Gruppen und Körper
- Graphentheorie

Erfolgreich im Studium

Studieninhalte

- Erkenntnisse aus der Gehirnforschung, z.B. Unterschied zwischen Wissen und Können, Verhaltenslernkurve, Lernbiologisches Naturgesetz, Vergessenskurve, Mythos Multitasking
- Lernstrategien, z.B. Lerntypen, Lernpyramide, Prüfungsvorbereitung, Umgang mit Prüfungsstress, Stressreduktion
- Selbstmanagement, z.B. Motivation und Aufschieberitis, Selbstorganisation, Selbstdisziplin, Zeitmanagement
- Hochschulen und Wissenschaft, z.B. Was ist Wissenschaft, Wissenschaftstheorie, Prüfungsordnungen
- Schreiben und Schreibprozess, z.B. präzise, verständliche Sprache, Aufbau und Gliederung von Dokumenten, Recherche
- Textverständnis, z.B. sinnvolles Lesen, Randbemerkungen, Exzerpte, Zitieren

2. Semester

The image features a minimalist, abstract design. The background is primarily yellow, with a white horizontal band across the middle. On the right side, there are several overlapping rectangular blocks of color: a dark olive green block at the top, a bright blue block below it, a dark teal block overlapping the blue one, and a vibrant green block at the bottom right. The text '2. Semester' is positioned on the white band, to the left of the colored blocks.

Software Engineering

Die Studierenden kennen grundlegende Ziele, Methoden, Techniken und Vorgehensweisen des Software Engineerings. Sie sind in der Lage, diese Methoden eigenständig anzuwenden, um technisch saubere, klar strukturierte Programme zu schreiben.

Studieninhalte

- Phasen der Softwareentwicklung
- Grundlagen der Vorgehensmodelle für die Softwareentwicklung
- Klassendiagramme, Objektdiagramme
- Zustandsdiagramme
- Sequenzdiagramme, Kommunikationsdiagramme
- Elementare Design Patterns
- Grundlagen des Modultests

Rechnernetze

Die Studierenden sollen ein gutes Verständnis für die fundamentalen Kommunikationsarchitekturen (OSI, TCP/IP) erwerben. Sie sollen die Aufgaben, Prinzipien, Mechanismen und Architekturen auf den unterschiedlichen Kommunikationsebenen kennenlernen. Praktische Erfahrungen sollen mit den Kommunikationsdiensten und -anwendungen im Rahmen von Laborversuchen unter der Anwendung von Protokollanalyse-Tools vermittelt werden.

Studieninhalte

- Netzwerktopologien
- ISO/OSI-Schichtenmodell
- Physische Grundlagen der Datenübertragung auf OSI-Schicht 1
- Grundlegende Protokolle der OSI-Schichten 2 bis 4
- Grundlegende Anwendungsprotokolle der OSI-Schichten 5 bis 7
- Grundlagen der Netzwerksicherheit

Grundlagen Web Development

Die Studierenden haben Einsicht in die grundlegenden Techniken, die bei der Entwicklung moderner Web-Anwendungen eingesetzt werden. Dafür kennen sie den grundlegenden Aufbau einer Web-Anwendung, beherrschen Skriptsprachen, sowie HTML und CSS. Die Studierenden kennen außerdem Standards zur Entwicklung von barrierefreien Webseiten und können entsprechende Testwerkzeuge einsetzen.

Studieninhalte

- Konzeption und Architektur einer Web-Anwendung
- Einführung und Vertiefung grundlegender Techniken: HTML, CSS, JavaScript
- Einführung in eine Skriptsprache (z.B. Python, PHP, Node.js)
- Gestaltung grafischer Benutzeroberflächen
- Webdesign und Usability
- Barrierefreie Informationstechnik

Gestaltung, Kommunikation und Präsentation

Die Studierenden sind in der Lage, unter Berücksichtigung des Präsentationsziels, der Präsentationssituation sowie der Zielgruppe, Präsentationen zielführend vorzubereiten, zu erstellen und durchzuführen. Sie kennen verschiedene Präsentationsmethoden und -medien und können diese kombinieren und handhaben. Die Studierenden sind fähig, Bewertungskriterien für Präsentationen zu erarbeiten und diese in eigenen Präsentationen anzuwenden.

Studieninhalte

- Präsentationsarten und -methoden
- Gestaltung von Präsentationsmedien
- Gestaltung von Bildschirmpräsentationen
- Kommunikationsmodelle
- Sprache, Stimme, Rhetorik
- Persönlicher Auftritt, Körpersprache, Verhalten
- Interaktionen in Präsentationen
- Aufbau von Präsentationen
- Ein guter Anfang und Schluss
- Vorbereitung, Ablauf und Nachbereitung von Präsentationen

Algorithmen und Datenstrukturen

Die Studierenden kennen typische Datenstrukturen der Informatik sowie Algorithmen auf diesen Datenstrukturen, insbesondere Such- und Sortieralgorithmen. Sie werden in die Lage versetzt, für gegebene Aufgabenstellungen geeignete Datenstrukturen und Algorithmen anhand ihrer Laufzeit und ihres Speicherplatzverbrauchs auszuwählen und zu programmieren. Dazu gehört auch die Fähigkeit Programme hinsichtlich ihrer Effizienz zu analysieren und zu bewerten.

Studieninhalte

- Abstrakte Datentypen
- Generics
- Komplexitätsanalyse von Algorithmen
- Listen, Stacks und Queues
- Effiziente Sortieralgorithmen
- Binäre Suchbäume (z.B. AVL-Bäume)
- Perfekt ausgeglichene Suchbäume (z.B. B-Bäume)
- Hashverfahren
- Collections
- Backtracking-Algorithmen

Statistik

Die Studierenden verfügen über Grundkenntnisse in der Statistik sowie über die Fähigkeit, diese Kenntnisse auf praktische Fragestellungen anzuwenden.

Studieninhalte

- Kombinatorik
- Wahrscheinlichkeitsdefinitionen
- Berechnen von Wahrscheinlichkeiten
- Bedingte Wahrscheinlichkeiten
- Diskrete und kontinuierliche Wahrscheinlichkeitsverteilungen
- Häufigkeitstabellen und Histogramme
- Lage- und Streuparameter
- Korrelation und Regression

3. Semester

The image features a minimalist, abstract design. A large yellow rectangle occupies the top half of the frame. Below it, a white horizontal band is partially obscured by a series of overlapping colored rectangles on the right side. These rectangles include a dark blue one, a medium blue one, a dark green one, and a light green one, creating a layered, geometric effect.

Software-System-Entwicklung

Die Studierenden kennen die Probleme größerer Softwareprojekte und der industriellen Software-Entwicklung. Sie sind in der Lage, Softwareprojekte über die reine Programmierung hinaus nach klassischen Methoden professionell abzuwickeln und im Team zu planen, zu entwickeln und zu testen.

Studieninhalte

- Vertiefung von Phasenmodellen
- Überblick Projektmanagement
- Systemanalyse, Systementwurf, Systemtest
- Use-Case-Diagramme
- Aktivitätsdiagramme
- Komponenten-Diagramme, Architekturmuster

Angewandte KI

Die Studierenden kennen die grundlegenden Methoden des überwachten und unüberwachten Lernens. Sie verstehen grundlegende Prinzipien, Fragestellungen und Ziele des maschinellen Lernens. Sie sind in der Lage, praktische Problemstellungen mit Verfahren des maschinellen Lernens eigenständig zu lösen. Hierfür sind sie in der Lage mit den entsprechenden Tools und Frameworks umzugehen. Die Studierenden verstehen insbesondere die Funktionsweise von neuronalen Netzen, kennen die wichtigsten Architekturen im Bereich Deep Learning und können diese auf ausgewählte Bereiche anwenden.

Studieninhalte

- Grundlagen der Klassifikation und Regression
- Stützvektormaschinen (SVM)
- Entscheidungsbäume
- Clustering, z.B. K-NN und EM-Algorithmus
- Neuronale Netze
- Deep Learning
- Netzarchitekturen, z.B. CNN oder R-NN
- Bestärkendes Lernen

Full Stack Web Development

Die Studierenden sind in der Lage Web-basierte Anwendungen mit modernen Methoden, Techniken und Werkzeugen im Team unter Anwendung agiler Projektmanagementmethoden zu entwickeln. Dafür kennen sie je ein server- und client-seitiges Web-Framework und können Datenbank-basierte Web-Anwendungen entwerfen und umsetzen.

Studieninhalte

- Kombinierte Anwendung der Techniken aus Grundlagen Web Development
- Grundlagen aktueller Web-Architekturen
- Serverseitiges Web-Framework, z.B. Spring, Laravel, Express
- Clientseitiges Web-Framework, z.B. React, Angular, Vue.js
- Testverfahren
- Datenpersistenz
- Einblick in Webservices
- Projektmanagement

IT-Sicherheit

Die Studierenden kennen die Funktionsweise von Angriffen auf Web-Anwendungen und Netzwerk-Protokollen und können diese analysieren, bewerten und abwehren.

Die Studierenden können sichere Informationssysteme entwickeln und sind sensibilisiert für Sicherheitsfragen.

Studieninhalte

- Grundlagen von IT-Sicherheit (Security Mindset, Kerckhoffs' Prinzip)
- Grundlagen der Kryptografie (symmetrische und asymmetrische Kryptografie, PKI, SSL/TLS)
- Verständnis von gängigen Schwachstellen in Web-Anwendungen (XSS, CSRF, SQL-Injection, etc) und die passenden Gegenmaßnahmen
- Vorstellung von Angriffen auf Netzwerkprotokolle (Sniffer, TCP/IP-Attacken, Spoofing, Man-in-the-Middle, DoS/DDoS, etc) und passende Gegenmaßnahmen

Mathematik für Informatiker

Die Studierenden kennen und verstehen grundlegende mathematische Ideen und Konzepte der Linearen Algebra und der Analysis. Sie beherrschen die Grundzüge der Formulierung von konkreten Fragestellungen mit Hilfe einer abstrakten mathematischen Beschreibungssprache und sind in der Lage, geeignete Problemstellungen als mathematische Aufgabenstellung zu formulieren und zu lösen.

Studieninhalte

- Lineare Algebra, z.B. Vektorräume, Rechenregeln für Vektoren und Matrizen, Lineare Gleichungssysteme
- Analysis, z.B. Grenzwerte, Stetigkeit, Differenzierbarkeit reeller Funktionen, Optimierung und Integration, Ausblick auf Funktionen mehrerer Veränderlicher

Englisch für Informatiker

Studieninhalte

- Vor- und Nachteile verschiedener Betriebssysteme erklären
- Verschiedene Arten von Benutzeroberflächen beschreiben
- IT-Probleme und mögliche Lösungen erörtern
- Unterschiedliche Lebenszyklusphasen der Softwareentwicklung beschreiben
- Vorschläge zur Softwareentwicklung und Fehlerbehebung machen
- Ungefähre Angaben machen und über Möglichkeiten und Abhängigkeiten sprechen
- Screenshots beschreiben (Symbole und Positionierung)
- Über Dokumentation, Tests und Debugging sprechen
- Funktionen und Vorteile eines Systems beschreiben
- Ihre Meinung zu Must-Have- und Nice-to-Have-Funktionen äußern
- User Support auf Englisch anbieten
- Verschiedene IT-Bedrohungen und Möglichkeiten sich online zu schützen zu diskutieren
- Verschiedene Arten digitaler Bedrohungen beschreiben und verschiedene Möglichkeiten zur Behebung dieser Bedrohungen erläutern
- Geschäftliche E-Mails auf Englisch schreiben
- Geschäftliche Anrufe auf Englisch tätigen

4. Semester

The image features a minimalist, abstract design. The background is primarily yellow, with a white horizontal band across the middle. On the right side, there are several overlapping squares: a dark blue square at the top, a medium blue square below it, a dark teal square overlapping the medium blue one, and a green square at the bottom. The text '4. Semester' is positioned on the white band, to the left of the blue and green squares.

Fortgeschrittene Programmier- techniken

Die Studierenden gewinnen einen tieferen Einblick in verschiedene Programmierparadigmen, insbesondere in die funktionale Programmierung, und erproben diese in konkreten Szenarien. Sie kennen typische Probleme und Szenarien der parallelen Programmierung und sind in der Lage, die entsprechenden Methoden, Techniken und Werkzeuge anhand praktischer Beispiele anzuwenden.

Studieninhalte

- Überblick Programmierparadigmen
- Parallele Programmierung, z.B. Threads und Prozesse, Synchronisation, Mutexe und Semaphore, bekannte Szenarien wie das Erzeuger-Verbraucher-Problem
- Funktionale Programmierung, z.B. Lambda-Ausdrücke, Stream API, Filter, Map und Reduce

Fortgeschrittene Konzepte der KI

Die Studierenden kennen weiterführende Methoden und Konzepte aus dem Bereich der Künstlichen Intelligenz. Sie kennen die unterschiedlichen Möglichkeiten, wie sich Systeme neues Wissen aneignen können, wie dieses repräsentiert wird und wie es verwendet wird, um zielgerichtete Aktionen auszulösen oder zu planen. Sie haben ein Verständnis für die Vor- und Nachteile verschiedener Such- und Problemlösungsstrategien entwickelt und sind fähig, bei Problemen eigenständig geeignete Algorithmen auszuwählen und anzuwenden. Sie haben Einblicke in die Komplexität der Entwicklung von Systemen mit künstlicher Intelligenz und der Unterscheidung der verschiedenen Formen künstlicher Intelligenz erlangt. Die Studierenden sind sensibilisiert für die Notwendigkeit, Entscheidungen einer KI nachvollziehbar zu gestalten und kennen die aktuellen Ansätze zur Erklärung, Interpretation und Visualisierung der Entscheidungsfindung.

Studieninhalte

- Aufbau und Klassifikation von intelligenten Agenten
- Symbolische und Subsymbolische Verfahren
- Problemlösen, z.B. A*, MiniMax & Pruning
- Schließen und Planen bei sicherem und unsicherem Wissen
- Klassifikation und maschinelles Lernen, z.B. Lazy-Learning
- Multiagentenlernen
- Erklärbare KI, z.B. Nachvollziehbarkeit, Verlässlichkeit

Cloud Computing

Die Studierenden beherrschen die Konzepte des Betriebssystems Unix / Linux theoretisch und praktisch. Auf Kommandozeilenebene verstehen und entwickeln die Studierenden eigenständig Shell-Skripte zur Lösung von Aufgaben der Systemverwaltung. Netzwerk- und Betriebssystemdienste und Konzepte werden von den Studierenden verstanden und in Betrieb genommen. Weiterhin werden die Studierenden in den Bereich der Server-Virtualisierung am praktischen Beispiel eingeführt. Die Studierenden sollen die Konzepte des Cloud Computings verstanden haben und auf verschiedene Anwendungsfälle (z. B. IaaS, PaaS, SaaS, etc.) anwenden können. Aufbau und Betrieb skalierbarer Container-Infrastrukturen werden vom Studierenden verstanden, an Problemstellungen angepasst und konfiguriert.

Studieninhalte

- Konzepte des Betriebssystems Unix / Linux
- Unix-Shell-Programmierung, z. B. BASH, ZSH, CSH
- Grundlagen, Konfiguration und Betrieb von Betriebssystem- und Netzwerkdiensten, z. B. NTP, SSH, HTTP, DBMS
- Grundlagen der Server Virtualisierung, z. B. VMware, Azure, VirtualBox
- Einführung in Cloud Computing
- Skalierbare Container-Infrastrukturen, z. B. Docker, Kubernetes

Data Science

Die Studierenden können eigene Data-Science-Projekte durchführen. Neben der Bewertung von Datenverwaltungssystemen haben sie Methoden und Verfahren gelernt, wie man die Herausforderungen an Datenmenge, Datenvielfalt und Zugriffsgeschwindigkeit meistern kann.

Studieninhalte

- Methoden zur Durchführung eines Data-Science-Projektes
- Aufgabenstellungen durch die 3 V's (Volume, Velocity, Variety)
- Systeme zur Verwaltung von Daten, z.B. für Small, Medium und Big Data
- Betrachtung von Architekturen zur Durchführung von Auswertungen, z.B. Batch-Verarbeitung oder Streaming
- Modellierung unterschiedlicher Datentypen

Effizientes Programmieren mit C/C++

Aufbauend auf der Programmiersprache Java werden die Programmierkenntnisse um die Programmiersprache C/C++ erweitert. Dabei wird insbesondere Wert auf effiziente Lösungen für anspruchsvolle Problemstellungen gelegt.

Studieninhalte

- Variablen
- Funktionen
- Kontrollstrukturen
- Präprozessor
- Pointer und Arrays
- Referenzen
- Klassen und Objekte
- Speicherverwaltung
- Templates
- Vererbung
- Operatorüberladung

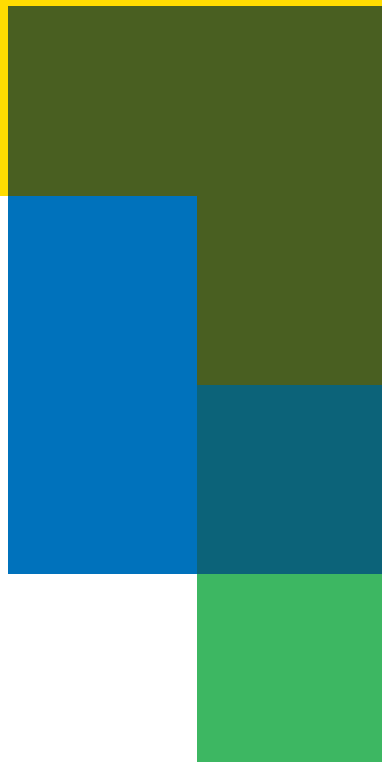
Digitalethik

Die Studierenden können wichtige Begrifflichkeiten der Digitalethik erklären und kennen unterschiedliche ethische Theorien, die sie an Praxisbeispielen anwenden können.

Studieninhalte

- Einführung und Begriffe, z.B. Moral, Ethik, Normen, Recht, Verantwortung, Informationsethik, Information, Medien
- Geschichtliche Betrachtungen zu Moral und Ethik
- Ausgewählte ethische Theorien, z.B. Utilitarismus
- Systematische Betrachtung ausgewählter Themen, z.B. Privatsphäre, Genauigkeit, Korrektheit, Eigentum, Zugang zu Informationen, Produktion von Informationen, Sammeln und Erschließen von Informationen, Verbreitung von Informationen, Machtverhältnisse, Informationsgesellschaft, Gender und Informationen
- Bestehende Codes of Ethics in der IT, z.B. ACM, IEEE
- Ausgewählte konkrete Anwendungsszenarien

**5. + 6.
Semester**



Software-Architektur

Die Studierenden verstehen die Konzepte guter Software-Architektur, insbesondere Architekturmuster, Entwurfsmuster und Idiome und sind in der Lage, diese anhand konkreter Szenarien zur Erstellung von Anwendungen auf Basis von State-of-the-Art-Architekturen einzusetzen.

Studieninhalte

- Design-Prinzipien, z.B. SOLID, Dependency Injection, Law of Demeter
- Grundlegende Software-Architekturen, z.B. objektbasierte Systeme, ereignisbasierte Systeme, verteilte Systeme
- Architekturmuster, z.B. MVC, Layers, Peer-to-Peer, Pipes and Filters, SOA
- Entwurfsmuster, z.B. Observer, Proxy, Adapter
- Idiome, z.B. equals, clone, immutability

Interdisziplinäres Softwareentwicklungsprojekt

Die Studierenden können die erworbenen Kenntnisse aus Veranstaltungen in den Bereichen Programmierung und Software Engineering im Rahmen eines Softwareprojekts erfolgreich anwenden. Sie sind in der Lage, Software systematisch und in studiengangübergreifenden Teams zu konstruieren, zu implementieren und zu testen.

Studieninhalte

- Aufsetzen und Durchführen eines Softwareprojekts nach einem vorgegebenen Vorgehensmodell
- Spezifikation, Konstruktion, Implementierung und Testen des Systems
- Agile Softwareentwicklung
- Refactoring
- Versionsverwaltung
- Arbeiten in heterogen zusammengesetzten Teams

Testverfahren für komplexe Software-Systeme ^W

Die Studierenden lernen die Methoden und Techniken zum Testen komplexer Software-Systeme kennen und erhalten einen Überblick über aktuelle Werkzeuge zur Unterstützung des Testprozesses.

Studieninhalte

- Unit Testing (JUnit)
- Mocking (Mockito)
- API Testing (Postman, REST Assured, SoapUI)
- Web Application Testing (Selenium)
- Acceptance Testing (Robot)
- Static Code Analysis (SonarQube)
- Performance Testing (Apache JMeter)
- Penetration Testing (Kali Linux, Metasploit)
- CI/CD (Jenkins)
- Consumer-Driven Contract Testing (Pact)
- Fuzz Testing (ClusterFuzz, american fuzzy lop)

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Internet of Things ^W

Die Studierenden können selbstständig eine IoT-Lösung entwickeln, die Sensordaten auswertet, interessante Ereignisse auf Grundlage dieser Daten erkennt und situationsgerecht Aktoren schaltet. Die Studierenden können Sensordaten mit Microcontrollern auslesen, verarbeiten und per Funktransceiver versenden. Die Studierenden haben ein Bewusstsein für den sparsamen Umgang mit Energie in batteriebetriebenen Sensoren und können ihr Microcontroller-Programm entsprechend optimieren. Die Studierenden kennen gängige Funkprotokolle für IoT-Anwendungen, sowie deren Vor- und Nachteile und können für Anwendungen je nach Einsatzzweck ein geeignetes wählen.

Studieninhalte

- Einsatzszenarien für IoT von Predictive Maintenance bis zu Wearables
- Microcontroller und Single-Board Computer für das IoT (Raspberry Pi, Arduino, Flora)
- Funkprotokolle für das IoT (Z-Wave, IP6LoWPAN, EnOcean)
- Datenübertragung in der digitalen Fabrik (OPC-UA, CoAP, MQTT)
- Microcontroller Programmierung mit Arduino
- Grundlagen des Entwurfs elektronischer Schaltkreise
- Prototypische Umsetzung von Schaltungen mit Arduino und Grove
- Funksteuerung von IoT Geräten über Android Smartphones (Bluetooth Smart / LE)
- Complex Event Processing und Datenauswertung im IoT
- Szenarien für das Internet der Dinge selbst umsetzen, insb. Maschinensteuerung, Lichtsteuerung, autonome Fahrzeuge, Wearables

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Robotik ^W

Ziel der Vorlesung ist, den Studierenden einen fundierten Überblick über die Grundlagen der Robotik zu vermitteln. Nach dem Besuch der Veranstaltung kennen und verstehen die Studierenden die grundlegenden Methoden zur Modellierung, Analyse und Steuerung von Robotern. Sie kennen und verstehen die Komponenten eines modernen Robotersystems und sind in der Lage ein solches System auszulegen, zu entwickeln und die Bewegungsabläufe zu planen. Die Studierenden haben einen Überblick über die aktuell verfügbare Sensorik und Aktorik und sind in der Lage, diese zielorientiert einzusetzen. Zudem sind sie mit den verfügbaren open-source Frameworks in der Lage selbst einen Roboter zu programmieren und zu steuern.

Studieninhalte

- Historie und aktuelle Anwendungsgebiete
- Bauformen und Komponenten
- Sensorik und Aktorik
- Kinematik und Dynamik
- Steuerung, Regelung und kinematische Bahnplanung
- Programmierung und lernende Systeme
- Frameworks: Orocos, Ros, OpenCV, PCL

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Angewandtes maschinelles Lernen ^W

Einführung in das Gebiet des maschinellen Lernens. Die grundlegenden Prinzipien werden vermittelt und ausgewählte Techniken und Ansätze werden theoretisch erläutert und praktisch evaluiert.

Studieninhalte

- Grundlegende Begriffe des maschinellen Lernens
- Aufbereitung von Daten
- Überwachtes Lernen
- Unüberwachtes Lernen
- Bestärkendes Lernen

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Moderne App- und Webentwicklung ^W

Einstieg in die Entwicklung moderner Webanwendungen und mobiler Apps mit aktuellen Webtechnologien.

Studieninhalte

- Grundlagen (Web, JavaScript, TypeScript, SPAs, etc)
- Konzepte moderner clientseitiger Frameworks (MVC, Komponenten, Templates, etc)
- Webanwendungen und mobile Apps mit Angular, React, Vue und Svelte
- PWAs und mobile Frameworks wie Ionic
- Weitere Ansätze

W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Datenanalyse und Data Mining ^W

Die Studierenden kennen die grundlegenden statistischen Tests sowie eine Auswahl an fortgeschrittenen Verfahren. Sie können entscheiden, welche Tests für welche Problemstellungen adäquat sind. Sie kennen wichtige Anwendungen der Datenanalyse. Die Studierenden haben einen grundlegenden Überblick über Data- und Text-Mining. Sie sind in der Lage, einfache Algorithmen des Data- und Text-Mining praktisch anzuwenden und zu programmieren.

Studieninhalte

- Wiederholung Statistik
- Einführung
- Wahrscheinlichkeitsverteilungen
- Grundlegende statistische Tests
- Anwendungen
- Data Mining (Clustering, Klassifikation, Assoziationsregeln)
- Text Mining (Preprocessing, Clustering)
- Maschinelles Lernen (Einführung, Support Vector Machines, Künstliche Neuronale Netze)

W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

RESTful Web Services ^W

Die Studierenden sind in der Lage, verteilte Web-Anwendungen aufbauend auf der REST-Architektur zu konzipieren und zu implementieren.

Studieninhalte

- Design Prinzipien von serviceorientierten Architekturen
- Einführung in das CRUD-Entwurfsmuster
- Grundlegende Techniken zur Implementierung von verteilten, REST-basierten Web-Anwendungen (Client und Server)
- Webservices und Sicherheit
- Auswirkungen von Semantic Web auf die Architektur einer Web-Anwendung

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

Kryptologie^W

Die Studierenden haben einen Überblick über die technischen Voraussetzungen und die Methoden sicherer Datenkommunikation. Sie wissen um die historische Entwicklung und Bedeutung der Kryptografie. Sie kennen aktuelle, moderne Verfahren der Datenverschlüsselung sowie deren mathematische Grundlagen. Ihnen ist bewusst, wie technische Verfahren angewendet werden müssen, um eine sichere Datenkommunikation zu gewährleisten. Sie verstehen die Konzepte moderner Anwendungen der Kryptografie wie beispielsweise die digitale Signatur.

Studieninhalte

- Historische Verschlüsselungsverfahren
- Steganografie
- Transpositions- und Substitutionschiffren
- Strom- und Blockchiffrierungen
- Mathematische Grundlagen moderner symmetrischer und asymmetrischer Verschlüsselungsverfahren
- Ablaufpläne sicherer Kommunikation (kryptografische Protokolle)
- Authentifizierung
- Kryptografische Prüfsummen
- Public Key Infrastrukturen
- Moderne Anwendungen der Kryptografie

^W Beispiele für Wahlmodule: Wählbar aus dem Fächerkatalog im 5. und 6. Semester

7. Semester

The image features a minimalist, abstract design. The background is primarily yellow, with a white horizontal band across the middle. On the right side, there are several overlapping rectangular blocks of color: a dark olive green block at the top, a bright blue block below it, a dark teal block overlapping the blue one, and a medium green block at the bottom. The text '7. Semester' is positioned on the white band, to the left of the colored blocks.

Praxissemester mit Projektarbeit und Bachelorarbeit

Die Studierenden bearbeiten mit wissenschaftlichen Methoden ein praxisnahes Thema, welches den Inhalten und Zielen des Informatik-Studiums entspricht. Sie beweisen, dass sie im Studium gelerntes Wissen auf konkrete Aufgabenstellungen anwenden können und darüber hinaus in der Lage sind, sich weiteres spezifisches Wissen aus Literatur und anderen Quellen anzueignen und dieses zu vernetzen. Sie belegen ihre Fähigkeit, selbständig ein abgegrenztes Arbeitsgebiet zu strukturieren sowie innerhalb der vorgegebenen Bearbeitungszeit die vereinbarten Ziele zu erreichen.

Studieninhalte

Die Studieninhalte werden in Absprache mit dem Aufgabensteller festgelegt.



**Hochschule
Hof**

University of
Applied Sciences

95028 Hof
Alfons-Goppel-Platz 1
Phone +49 9281 409-3000
Fax +49 9281 409-4000
mail@hof-university.de
www.hof-university.de

Studiengangleiter/Fachberater

Prof. Dr. Matthias Meitner
matthias.meitner@hof-university.de
Phone +49 9281 409-4515

