

Module Catalogue

»Required elective modules«

Bachelor



Draft, Expected date of publication: Oktober 2024

The following text is a commentary in English language on the module manual of Technische Hochschule Augsburg, helping you to understand the contents of the German document. The legally binding text remains the German version of the module manual. Please refer to the German text if possible or seek advice in case of uncertainties. The purpose of the module descriptions is to provide a content-related overview of your degree course.

Only the current version of the university catalogue and examination regulations shall be deemed legally binding.

Contents

1	Required elective modules Bachelor - Offer	4
1.1	Current semester: Winter semester 2024/25 - Draft	4
1.2	Past semester: Summer semester 2024	6
2	Required elective modules - Overview	8
2.1	3D printing	8
2.2	Fundamentals of ABAP	12
2.3	Advanced Security Testing	16
2.4	Agile development of a Click-Dummy Game	20
2.5	Agile Software Development (Scrum)	24
2.6	Agile Web Applications with Python	28
2.7	Business Information Systems	30
2.8	Chance and Risk Management in Digitized Value Networks	32
2.9	Compiler	36
2.10	Computer Games Development	40
2.11	Corporate Entrepreneurship	42
2.12	Fundamentals of Data Communications	46
2.13	Data communication in the vehicle	48
2.14	Digital Innovation	52
2.15	Digital Business Leadership Skills	56
2.16	Digital Transformation in Organizations	60
2.17	Introduction to Natural Language Processing	64
2.18	Introduction to Robotics	68
2.19	Electronic Trading Systems	70
2.20	Embedded Linux	72
2.21	Flying Robots	74
2.22	Formula Student Driverless	76
2.23	Fullstack Web Development	80
2.24	Fundamentals of DevOps	84
2.25	Hard- and software for the internet of things	86
2.26	University Innovation Project	88
2.27	Industrial Image Processing	90
2.28	Industrial Data Processing	94
2.29	Information technology and the environment	98
2.30	Integrated Business Processes with SAP ERP	102
2.31	Interaction Engineering	104
2.32	Interactive Computer Graphics	108
2.33	IT-Consulting	112
2.34	IT Forensics	116
2.35	IT Security	118
2.36	IT Sourcing and Cloud Transformation	122
2.37	Classic Project Management Modernized	124

2.38 Concepts of Database Technology	128
2.39 Artificial intelligence in safety-critical applications	132
2.40 Lean IT & Enterprise Architecture	136
2.41 Linux LPIC	140
2.42 Artificial Intelligence	144
2.43 Mobile Robots	148
2.44 Pattern recognition and machine learning	150
2.45 Network Penetration Testing	152
2.46 Neural Networks and Deep Learning	154
2.47 NoSQL	158
2.48 Object-oriented programming with Python	160
2.49 Open Source Software Development	164
2.50 Practical Robotics with Matlab	168
2.51 Process Intelligence	172
2.52 Programming using Databases	174
2.53 Programming with Python	176
2.54 Programming of Web Applications	180
2.55 Project Jupyter	184
2.56 Project - Research and Transfer	188
2.57 RFID and NFC technology	192
2.58 Swabia Innovation Masterclass	194
2.59 Service Learning Project	202
2.60 Software engineering for data-intensive systems	204
2.61 Software Project Management	206
2.62 Startitup - Entrepreneurial Thinking and Business Design	208
2.63 Visual Thinking for Business	212

1 Required elective modules Bachelor - Offer

These lists only contain the compulsory elective modules offered at the Faculty of Computer Science.

Please refer to the responsible faculties for all other subjects.

1.1 Current semester: Winter semester 2024/25 - Draft

Module	Creditpoints	Credit hours
Fundamentals of ABAP ¹	8 CP	6 CH
Agile Software Development (Scrum)	5 CP	4 CH
Agile Web Applications with Python	5 CP	4 CH
Corporate Entrepreneurship	5 CP	4 CH
Data communication in the vehicle	5 CP	4 CH
Digital Business Leadership Skills	8 CP	6 CH
Electronic Trading Systems	5 CP	4 CH
Flying Robots	5 CP	4 CH
Hard- and software for the internet of things	5 CP	4 CH
University Innovation Project	5 CP	4 CH
Integrated Business Processes with SAP ERP	5 CP	4 CH
Interaction Engineering ²	5 CP	4 CH
IT Forensics	5 CP	4 CH
IT Security	5 CP	4 CH
Classic Project Management Modernized ⁵	5 CP	4 CH
Concepts of Database Technology	5 CP	4 CH
Artificial intelligence in safety-critical applications	5 CP	4 CH
Mobile Robots	5 CP	2 CH
Pattern recognition and machine learning	5 CP	4 CH
Process Intelligence	5 CP	4 CH
Programming using Databases	5 CP	4 CH
Project - Research and Transfer	10 CP	8 CH
Service Learning Project	5 CP	4 CH
Software Project Management	5 CP	4 CH
Systems programming ⁴	6 CP	5 CH

(1) WPF only for IN, TI and IA. For WI, IIS it is a compulsory module (WI: Programming 3, IIS: Programming of Information Systems).

(2) WPF only for IN, TI.

(3) WPF only for WI, IIS, TI and IA. For IN it is a compulsory module (Programming 3).

(4) WPF only for TI. This is a compulsory module for IN.

(5) WPF only for IN, WI, IIS and TI. Not for IA.

Block events

Module	Creditpoints	Credit hours
3D printing	5 CP	4 CH
Computer Games Development	5 CP	4 CH
Linux LPIC	5 CP	4 CH
Practical Robotics with Matlab	7.5 CP	6 CH
Swabia Innovation Masterclass ⁵	10 CP	8 CH
Visual Thinking for Business	5 CP	4 CH

(5) WPF two semesters

1.2 Past semester: Summer semester 2024

Module	Creditpoints	Credit hours
Agile development of a Click-Dummy Game	5 CP	4 CH
Agile Software Development (Scrum)	5 CP	4 CH
Agile Web Applications with Python	5 CP	4 CH
Business Information Systems	5 CP	4 CH
Database Applications ¹	3 CP	3 CH
Fundamentals of Data Communications ²	5 CP	4 CH
Digital Business Leadership Skills	7.5 CP	6 CH
Electronic-Commerce	7.5 CP	6 CH
Introduction to Natural Language Processing	5 CP	4 CH
Introduction to Robotics	5 CP	4 CH
Embedded Linux	7.5 CP	6 CH
Formula Student Driverless	5 CP	4 CH
Fullstack Web Development	8 CP	6 CH
Fundamentals of DevOps	5 CP	4 CH
University Innovation Project	5 CP	4 CH
Industrial Image Processing	5 CP	4 CH
Industrial Data Processing	5 CP	4 CH
Interactive Computer Graphics	7.5 CP	6 CH
IT Security	7.5 CP	6 CH
IT Sourcing and Cloud Transformation	5 CP	4 CH
Classic Project Management Modernized	5 CP	4 CH
Artificial Intelligence	7.5 CP	6 CH
Network Penetration Testing	5 CP	4 CH
Neural Networks and Deep Learning	5 CP	4 CH
NoSQL	5 CP	4 CH
Object-oriented programming with Python	5 CP	4 CH
Open Source Software Development	5 CP	4 CH
Programming of Web Applications	5 CP	4 CH
Project - Research and Transfer	10 CP	8 CH
Service Learning Project	5 CP	4 CH

Startitup - Entrepreneurial Thinking and Business Design	5 CP	4 CH
--	------	------

(1) WPF only for IN, IIS. For WI it is a compulsory module.

(1) WPF only for WI, IIS. For IN, TI it is a compulsory module.

Block events

Module	Creditpoints	Credit hours
Advanced Security Testing	5 CP	4 CH
Chance and Risk Management in Digitized Value Networks	2.5 CP	2 CH
Computer Games Development	5 CP	4 CH
Linux LPIC	2.5 CP	2 CH
Linux LPIC Advanced	2.5 CP	2 CH
Search Engine Optimization (SEO)	2.5 CP	2 CH
Visual Thinking for Business	5 CP	4 CH

2 Required elective modules - Overview

2.1 3D printing

Information about the module

engl. Name	3D printing
Code	3DDR4.WP
Coordinator	Prof. Dr. Jürgen Scholz
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	3D Printing (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying work placement to apply and deepen the knowledge acquired. In addition, the practical course supports and promotes group work and self-study.
Prerequisites	Knowledge of programming
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written examination, 60 minutes, none auxiliaries, 25%• Written assignment, 10-25 pages, with presentation of results, 5-10 minutes, 75%
Examination number	IN -, - TI - WI - IIS -

Content of the module

Coordination of the lecture content with previous knowledge / interests of the participants

- Introduction
 - What is 3D printing?
 - Areas of application
- Types of printers
 - What types are there and how do they work (FDM, SLA, DLP, SLS, MSLA, BJ, MJet, EBM)
 - What technology is used for what (also specify costs for types)
 - industrial printer vs consumer printer
- FDM printer
 - What are the differences (movable axes, delta, direct drive, multi nozzle, chamber, print bed,
 - Nozzle types, etc.)
 - which manufacturers are there, what are the differences
- Materials
 - What materials are there and what are the special features
- Slicer
 - Why do you need this
 - How does it work (only briefly address)
 - Which settings do what
 - Examples (different slicers, example objects)
 - Discuss typical problems
- Use of printers (FDM)
 - Create object (download)
 - Operating the slicer
 - Start printing
 - Setting the printer correctly
 - Maintain printer
- Different firmware
 - Marlin

- Reprap
- Clipper
- Future technologies/new approaches
 - Current developments to accelerate printing
 - Conveyor belt printing bed
 - 4/5 axis 3D printer
 - Variable Size Nozzle
 - Nozzle extruder
 - ...
- Practical part
 - Assembly of a 3D printer in group work - alternative printing of 3D objects
 - Projects on and with 3D printers

Qualification aims for the module learning objectives/skills

Students understand how an (FDM) 3D printer works. They learn how to set it up, adjust and operate it. They also learn how to create and print simple 3D objects. After successfully completing this module, students are authorized to use the faculty's 3D printers.

Reading list

Literature recommendations will be provided in the first lecture.

2.2 Fundamentals of ABAP

Information about the module

engl. Name	Fundamentals of ABAP
Code	ABAPGL6.WP
Coordinator	Prof. Dr. Stefan Bensch Dipl.-Wirt.-Inf. (FH) Christian Herdin, M.Sc.
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Fundamentals of ABAP (4 Credit hours) Practical Work Fundamentals of ABAP (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, exercises and accompanying practical training to apply and deepen the knowledge acquired. In addition, the practical course supports and promotes self-study.
Prerequisites	Software development and Programming 1 and 2
Usage possibilities	Module only for the Bachelor's degree programs in Computer Science, Computer Engineering and Interactive Media. For Information Systems and International Information Systems it is a compulsory subject (WI: Programming 3, IIS: Programming of Information Systems)
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	Electronic examination, 90 minutes, auxiliary: lecture notes, SAP Software, Office applications for text and data processing
Examination number	IN 3970339, 2970830 TI 3976559, 2976662
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Fundamentals of programming business application systems:

- Technical basis
- Basics in ABAP
- Classic reporting, classic events for page design and interactive reporting
- Data types (variables and constants) and programming structures
- Decisions
- Repetitions
- Fields and character strings
- Functions
- Complex data types

Advanced programming

- Object-oriented reporting with ABAP Objects
- References and memory management
- Events
- Interfaces
- Inheritance
- Error handling
- Global classes
- Advanced programming techniques

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Describe keywords of the ABAP programming language and their function.
- Understand source code.
- Implement given algorithms independently and efficiently.
- Develop simple algorithms independently.
- Implement requirements independently.

Reading list

Keller, Horst und Sascha Krüger. ABAP Objects: ABAP-Programmierung mit SAP NetWeaver. 3. Aufl. Bonn: SAP PRESS, 2006.

Roth, Felix. ABAP Objects: Das neue umfassende Handbuch zu Konzepten, Sprachelementen und Werkzeugen in ABAP OO. 1. Aufl. Bonn: SAP PRESS, 2016.

Schwaiger, Roland. Schrödinger programmiert ABAP: Das etwas andere Fachbuch - Dein unterhaltsamer Einstieg in ABAP. 2. Aufl. Bonn: SAP PRESS, 2014.

„SAP ERP - SAP Help Portal”. Help Portal. SAP Help Portal SAP ERP. Zugriffen 8. März 2019. https://help.sap.com/viewer/p/SAP_ERP.

2.3 Advanced Security Testing

Information about the module

engl. Name	Advanced Security Testing
Code	AST4.WP
Coordinator	Dr.-Ing. Matthias Niedermaier
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Advanced Security Testing (4 credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Integrated lecture
Prerequisites	Knowledge of IT security essential
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written examination, 60 minutes, none auxiliaries, 50%• Presentation, 20-30 minutes, 50%
Examination number	IN 3970372, 2970870 TI 3976568, 2976681 WI 3975790 IIS 9775100
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Standards for security tests
- Reporting
- Use of tools
- Excerpt not complete: Nessus, OpenVAS, Metasploit, binwalk, firmware modification kit, ZAP, Checkstyle, CCP Check, burp suite
- Creation of own scripts to highlight current IT security aspects
- Procedure for software tests
- Procedure for product tests / hardware tests
- Procedure for testing IT landscapes
- The current state of technology and research in relation to IT security is conveyed

Procedure

- The following IT security topics are covered in the lecture
 - Network security
 - Hardware tests
 - Software testing methods
- Vulnerabilities and protective measures are practically tested on current devices and software
- The students have to work on a scientific question in project groups, here topics are deepened and the state of research is taken up

Qualification aims for the module learning objectives/skills

Knowledge:

- In the lecture, the planning, procedure and completion of security tests will be discussed with practical questions. In order to keep the lecture as close as possible to professional practice, a wide range of tools/tools will be used.
- Emphasis is placed on the broadest possible variety of topics in this area. This includes detecting software vulnerabilities in source code, testing entire networks and hardware-related issues.

Skills:

- Performing classic security product tests
- Performing network security tests
- Attacks and defense on hardware
- Performing software tests

Competencies:

- Students can carry out penetration tests with the help of tools, among other things
- They can familiarize themselves with new topics in the context of secure architectures
- Students are able to fundamentally test products for their IT security level

Reading list

HUANG, Andrew Bunnie. The Hardware Hacker: Adventures in Making and Breaking Hardware. 2017.

HUANG, Andrew. Hacking the Xbox: An Introduction to Reverse Engineering. 2002.

ERICKSON, Jon. Hacking: The Art of Exploitation. No Starch Press, 2008.

Script

2.4 Agile development of a Click-Dummy Game

Information about the module

engl. Name	Agile development of a Click-Dummy Game
Code	AEKDS4.WP
Coordinator	Matthias Regner, M.Eng.
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester. Sometimes unscheduled in summer semester.
Courses	Agile development of a Click-Dummy Game (4 Credit hours
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching with theoretical and practical parts. The development of the click dummy will take place as project work in groups.
Prerequisites	Basic knowledge of software development
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written assignment, 20-30 pages, 90%• Presentation, 30 minutes, 10%
Examination number	IN 3970391, 2970889 TI 3976569, 2976711 WI 3975809 IIS 9775163
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The module provides participants with the knowledge to plan, set up and implement IT projects using agile project methods. The focus is on the practical application of the Scrum method in the context of a group project. There will be short theory units each week, which enrich the project with new agile elements until a fully-fledged Scrum process is run through at the end.

There will be an optional excursion to Munich for a Scrum-Minecraft workshop.

Theory units:

- Basics for agile concepts and Scrum
- Development of product visions
- Estimation techniques in agile projects
- Kanban & Scrumban
- Extreme programming
- Quality management in agile projects
- Scaled Scrum frameworks (SAFe, Less, Nexus, ...)
- Getting to know prototyping tools
- Gradual introduction of new Scrum elements

Group project:

- Use of a prototyping tool (e.g. Figma, Adobe XD, ...)
- Using Scrum to design a click dummy for a digital game
- Holding regular reviews and retrospectives
- Planning a sprint with the help of digital tools
- Recording to-dos in the form of user stories in the product backlog
- Maintaining a sprint backlog during development

The application of the Scrum method is clearly the focus of the module. The click dummy only serves as an illustrative object. The aim is not to develop the best click dummy, but to gain and reflect on initial experience with Scrum projects in practice.

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Assess the benefits of agile methods in projects
- Select, plan and apply agile project methods
- Apply the Scrum method in projects in practice
- Gain and reflect on experience with Scrum
- Select and apply methods for estimating effort in agile and non-agile setups
- Perform sprint planning and backlog definitions
- Apply and combine different prioritization strategies (cost/benefit, needs) in backlog management
- Name scaled agile frameworks
- Explain the functionality and benefits of the Scaled Agile Framework

Reading list

Literature will be announced during the course.

2.5 Agile Software Development (Scrum)

Information about the module

engl. Name	Agile Software Development (Scrum)
Code	SCRUM4.WP, SCRUMZ.WP
Coordinator	Dipl.-Inf. (FH) Gregor Liebermann, M.Sc.
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, every semester
Courses	Agile Software Development (Scrum) (2 Credit hours) Practical Work Agile Software Development (Scrum) (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching
Prerequisites	First experience in programming and requirements analysis
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Admission requirement for the examination	Practical work Agile Software Development (Scrum)
Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970323, 2970791 TI 3976560, 2976565 WI 3975711 IIS 9775102
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Basics:

- Classic and agile development methods
- Agile manifesto
- Iterative approach

Scrum:

- Basics and motivation
- Requirements management
- Roles and meetings
- Sprints and procedure
- Release planning

The team:

- Phases of team development
- Personality profiles
- Communication and the four-sided model
- Team building

Scrum tools and practice:

- Scrum in practice and possible problems
- Continuous integration
- Pair Programming
- CVS and SVN
- Bug tracking
- Review tools
- Digital task boards

Other agile methods:

- Extreme Programming
- Crystal
- FDD
- Excursus: Kanban
- Excursus: Design Thinking

Qualification aims for the module learning objectives/skills

Students know the advantages and disadvantages of agile development methods compared to traditional process models and can assess which methodology is suitable for which project and which is not. The basics of Scrum were learned in a practical way.

Weighting of individual performance in the final grade

In order to pass, both components must be successfully completed.

Reading list

Literature recommendations will be provided in the first lecture.

2.6 Agile Web Applications with Python

Information about the module

engl. Name	Agile Web Applications with Python
Code	PYTHON4.WP
Coordinator	Dipl.-Inf. (FH) Dipl.-Designer (FH) Erich Seifert, MA
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is offered irregularly or on demand offered.
Courses	Agile Web Applications with Python (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching to impart theoretical knowledge, practical implementation of the course work in groups supplements the lecture and and promotes teamwork and self-study, the written part of the student research project conveys the ability to evaluate the knowledge gained, the presentation promotes independent analysis and evaluation of new knowledge
Prerequisites	Programming with Python
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written assignment, 10-25 pages, 80%• Presentation, 10-25 minutes, 20%
Examination number	IN 3970329, 2970801 TI 3976562, 2976573 WI 3975721 IIS 9775104
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Agile development methods
- Test Driven Development
- Web technologies (HTML, CSS, JavaScript)
- Software architecture for web applications
- Introduction to various Python frameworks for web development

Qualification aims for the module learning objectives/skills

Students will be able to assess various frameworks for web development and can select them to suit their own projects. Agile development techniques in the web environment are known and have been deepened in a practical way. New technologies can be analyzed and evaluated independently.

Reading list

Will be announced in the first classroom session.

2.7 Business Information Systems

Information about the module

engl. Name	Business Information Systems
Code	BEINF4.WP
Coordinator	Prof. Dr. Stefan Bensch
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Business Information Systems (2 Credit hours) Practical Work Business Information Systems (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying exercises to apply and deepen the acquired knowledge. In addition, the internship supports and and promotes self-study.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Project Work
Examination number	IN 3970386, 2970884 TI 3976571, 2976706 WI 3975804 IIS 9775164
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Introduction to the topic of business information systems: operational and analytical systems (business intelligence)
- Overview of the topic: Operational information systems, architectures and developments
- Methods and structuring approaches for analysis, design and development
- Transfer to practical application examples: Development and application in the project.

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Know and understand basic concepts, solutions and application areas of business information systems
- Successfully apply concepts and solutions
- Perform practical exercises and simulations (with SAP S/4HANA, SAP Analytics Cloud and SAP Datasphere)
- Independently acquire current knowledge and the state of research on business information systems

Reading list

Aßmann, Dietz, Japing, Jensen, Kästner, Rose, Scivos: ABAP Objects: SAP Data Warehouse Cloud, Rheinwerk Publishing SAP PRESS, 2023

Baars, H., Kemper, HG.: Business Intelligence & Analytics – Grundlagen und praktische Anwendungen: Ansätze der IT-basierten Entscheidungsunterstützung. Springer Vieweg, Wiesbaden, 2021

Gluchowski, P., Chamoni, P. (Hrsg.): Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen, 5. Aufl., Springer Gabler, Berlin, 2016

Kemper, H.-G., Baars, H., Mehanna, W.: ABAP Objects: Business Intelligence – Grundlagen und praktische Anwendungen: Eine Einführung in die IT-basierte Managementunterstützung, 4. Aufl., Springer Vieweg, Wiesbaden, 2016

Laudon, Kenneth C., Jane P. Laudon und Detlef Schoder: Wirtschaftsinformatik: Eine Einführung. 3. Aufl. Hallbergmoos/Germany: Pearson Studium, 2015

Roth, Felix: ABAP Objects: Das neue umfassende Handbuch zu Konzepten, Sprachelementen und Werkzeugen in ABAP OO. 1. Aufl. Bonn: SAP PRESS, 2016

2.8 Chance and Risk Management in Digitized Value Networks

Information about the module

engl. Name	Chance and Risk Management in Digitized Value Networks
Code	CHRIMG4.WP
Coordinator	Prof. Dr. Björn Steven Häckel
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	one semester; is regularly offered in the winter semester as well as in the summer semester
Courses	Chance and Risk Management in Digitized Value Networks (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Lecture with integrated exercises and discussion of practical case studies.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Written examination, 60 minutes, auxiliary: non-programmable calculator, 50% • Written assignment, 5-10 pages, 25% • Presentation, 10-15 minutes, 25%
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Driven by developments such as increasing globalization or advancing digitalization in the wake of Industry 4.0 and disruptive technologies such as AI, industrial value networks are subject to constant change. On the one hand, these developments result in promising opportunities for the companies involved, such as the opening up of new markets, the development of new, data-based products and services or the more flexible manufacturing of products. On the other hand, these developments also pose considerable risks for companies due to the complex and often non-transparent dependency relationships in networked value creation systems. In order to give students an overview of the resulting challenges for corporate management, the course deals with the following questions, among others:

- What are the key characteristics of value networks?
- What factors contribute to the transformation of value networks?
- What opportunities and risks arise from increasing networking?
- How can value networks be modeled and analyzed?
- What influence does AI have on value networks?

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Explain the basic elements of and factors influencing digitalized value networks.
- Characterize the opportunities and risks in digitalized value networks.
- Outline key challenges in the digital transformation of business models in value creation systems.
- Assess the implications of digitalization on entrepreneurial business models and the structure of value creation systems.
- Analyze the dependency structures in complex value creation systems using selected methods.
- Apply selected methods (especially centrality measures) to determine the criticality of individual actors in value networks.

Reading list

Broy, M., (2011): Integrierte Forschungsagenda Cyber-Physical Systems – Vortrag Embedded Systems, Symposium 2011.

Fraunhofer IAO (2013): Produktionsarbeit der Zukunft – Industrie 4.0.

Sydow, J. (1992): Strategische Netzwerke. Evolution und Organisation, Wiesbaden.

Fleisch E., Weinberger M., Wortmann F. (2015): Geschäftsmodelle im Internet der Dinge. Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung 67:444 - 465.

Schneider, S. und Spieth, P. (2013): Business Model Innovation - Towards an Integrated Future Research Agenda. International Journal of Innovation Management, Vol. 17, No. 1.

Osterwalder, A. und Pigneur, Y. (2011): Business Model Generation; Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt, New York.

Brandes, U. und Erlebach, T. (Hrsg.) (2005): Network Analysis – Methodological Foundations, Band 3418 der Reihe Lecture Notes in Computer Science. Springer, 2005.

Fridgen, G. und Garizy, Tira Zare (2015): Supply Chain Network Risk Analysis“, Veranstaltungsbeitrag, 23rd European Conference on Information Systems (ECIS) , 26.-29.05.2015, Münster, Germany.

2.9 Compiler

Information about the module

engl. Name	Compiler
Code	COM4.WP
Coordinator	Prof. Dr. Rolf Winter
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is offered irregularly or on demand.
Courses	Compiler (3 Credit hours) Practical Work Compiler (1 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching and accompanying practical course to apply and deepen the knowledge acquired. In addition, the practical course supports and promotes group work and self-study.
Prerequisites	Solid knowledge of a higher programming language such as JAVA or C / C++
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: 1 DIN A4 page handwritten
Examination number	IN 3970320, 2970776 TI 3976573, 2976515 WI 3975696 IIS 9775107
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

How often is a more or less small scanner or parser needed? Often adventurous on self "invented" scanners and parsers. To make it easier to jump from a few lines of code in good time, knowledge of the structure and mode of operation of compilers is important.

In this lecture, the functionality of parsers, scanners and compilers is explained. The sensible use of tools is described based on the theoretical principles.

First, the theoretical foundations of compiler construction - formal languages and automata - are developed. The focus here is on CH-2 and CH-3 languages, which are particularly relevant for compilers. Building on the theory, the practical realization of compiler construction is then discussed. The path leads to the construction of programs for lexical and syntactic analysis. Their concrete realization is illustrated using commonly used programs. A compiler is created with the help of common tools.

- Introduction
- Language theory: basics of language analysis
- Lexical analysis
- lex / flex
- The syntax analysis
- Semantic analysis
- The compiler generator YACC / BISON
- Intermediate code generation
- Code optimization

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- recognize where the areas of application of scanners and parsers lie.
- describe the function and mode of operation of scanners and parsers.
- design and create a compiler consisting of scanner and parser based on the theory of formal languages for a given task.

Reading list

A.V. Aho, R. Sethi, J.D. Ullmann: Compilerbau. Band 1 und 2, Addison-Wesley 1999

A.V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman Compilers: Principles, Techniques, and Tools. Addison-Wesley, 2007.

A.W. Appel modern compiler implementation in java, Cambridge University Press 2004
Download:
<http://eden.dei.uc.pt/~amilcar/pdf/CompilerInJava.pdf>

B. Bauer, H. Höllerer: Übersetzung objektorientierter Programmiersprachen: Konzepte, Abstrakte Maschinen Und Praktikum: Java-Compiler; Springer; 4. Auflage; 2013

S.D. Bergmann Compiler Design: Theory, Tools, and Examples; free download: <http://elvis.rowan.edu/~bergmans/> (Computer Science Department, Rowan University), 2016

H. Herold: Linux-Unix-Profitools. Addison-Wesley 1999

D. Grune, K. van Ree, H.E. Bal, C.J.H. Jacobs, K. Langendoen: Springer; 2. Auflage 2012

R.H. Güting, M. Erwig: Übersetzerbau; Springer 1999

A. Kunert: LR(k)-Analyse für Pragmatiker; Humboldt-Universität zu Berlin; Institut für Informatik / ZE Rechenzentrum (CMS) (Dissertation) 2011

Levine, J. R., Mason, T., Brown, D.: lex & yacc; O'Reilly & Associates 1995

A.J.D. Reiss. Compiler Construction using Java, JavaCC, and Yacc; Wiley, 2012.

F.J. Schmitt: Praxis des Compilerbaus; C. Hanser 1992

Wagenknecht C, Hielscher M.: Formale Sprachen, abstrakte Automaten und Compiler, Lehr- und Arbeitsbuch für Grundstudium und Fortbildung, Vieweg Teubner 2009
available for download via Springer Link!

2.10 Computer Games Development

Information about the module

engl. Name	Computer Games Development
Code	CGDEV4.WP
Coordinator	Philip McClenaghan
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course during the semester break. (February/March) and (August/September)
Courses	Computer Games Development (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical classes and workshops
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 10-30 minutes, 40%• Written assignment, 8-25 pages, 60%
Examination number	IN 3970322, 2970788 TI 3976563, 2976562 WI 3975708 IIS 9775108
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The aim of this course is to provide students with an understanding of computer game theory and design. This is not a technical course. Conceptual design and critical analysis exercises allow students to explore a range of relevant topics in order to gain the ability to look at computer games objectively and from an informed standpoint. Students present their work (in English) both verbally and in written form through presentations and analysis documents.

Qualification aims for the module learning objectives/skills

On completion of this module, the student will be able to demonstrate:

- An appreciation of the computer games industry
- An understanding of computer games design and the ability to critically evaluate computer games
- An understanding of design implementation
- The ability to create a pre-production games proposal document
- The ability to articulate course related ideas and concepts in English, both verbally and in written form

Reading list

Sylvester, T. (2013) Designing Games: A Guide to Engineering Experiences. O'Reilly

Gamasutra Website (<http://www.gamasutra.com/>)

2.11 Corporate Entrepreneurship

Information about the module

engl. Name	Corporate Entrepreneurship
Code	CES4.WP
Coordinator	Prof. Dr. Christoph Buck
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Corporate Entrepreneurship (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching for the theoretical imparting knowledge and skills in combination combined with interactive application and reflection of what has been learned in the sense of experience-based learning.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Interim presentation, 15 minutes, 30%• Final presentation, 25 minutes, 35%• Written elaboration of the final presentation, approx. 6-8 pages, 35%
Additional information on the type of examination	The presentations are group presentations. Student research project: concrete solution proposals for practical problems are to be developed and presented in group work.

Examination number	IN 3970396, 2970894 TI 3976564, 2976716 WI 3975814 IIS 9775174
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Developing, evaluating and implementing innovations in (large) companies consists of various skills that can be learned.

In this course students learn:

- the basics of corporate entrepreneurship;
- the specifics, needs and approaches of corporate entrepreneurship;
- strategies, tools and methods for entrepreneurship within companies and apply these in the context of practical problems
- opportunities, risks and challenges of corporate entrepreneurship.

Students are accompanied by an industry partner throughout the course.

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Identify and evaluate opportunities for corporate entrepreneurship within organizations
- Develop innovative solutions and create a strategy for their implementation in a company
- Create business models that can be embedded in the corporate context in terms of costs, benefits, risks and opportunities

Reading list

Osterwalder, A., Pigneur, Y. (2010): Business Model Generation Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt am Main, 2010.

Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A. (2014): Value Proposition Design.

Nambisan, S., Lyytinen, K., Majchrzak, A., Song, M. (2017): Digital Innovation Management: Reinventing Innovation Management Research in a Digital World. Management Information Systems Quarterly, 41 (1), 223–238.

Kohli, R., Melville, N.P. (2018): Digital innovation A review and synthesis. Information Systems Journal, 29 (1), 200–223.

Christensen, C. M. (2011): The innovator's dilemma: Warum etablierte Unternehmen den Wettbewerb um bahnbrechende Innovationen verlieren. Vahlen.

Kraus, R., Kreitenweis, T., & Jeraj, B. (2022): Intrapreneurship. Springer.

2.12 Fundamentals of Data Communications

Information about the module

engl. Name	Fundamentals of Data Communications
Code	DAKO4.WP
Coordinator	Prof. Dr. Rolf Winter
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Fundamentals of Data Communications (3 Credit hours) Practical Work Fundamentals of Data Communications (1 Credit hour)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar-based teaching, practical training
Prerequisites	None
Usage possibilities	WPF only for Bachelor's degree programs: Information Systems, International Information Systems and Interactive Media. For Computer Science (Bachelor) and Computer Engineering (Bachelor) it is a compulsory subject.
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: calculator
Examination number	WI 3975755 IIS 9775111
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Functionality and structure of the Internet architecture and its principles and protocols in particular:

- Application layer protocols (such as HTTP and DNS)
- Transport protocols (such as TCP and UDP)
- Routing protocols (link state and distance vector)
- Protocols of the data link layer (e.g. Ethernet)
- Operation of core components of the internet (switches, CDNs, NAT, etc.)
- Aspects of network security (e.g. packet filters)
- Key principles of the Internet (reliable data transmission, congestion control, etc.)
- Dealing with standard tools (software) in the field of networks
- Network setup, maintenance and fault diagnosis

Qualification aims for the module learning objectives/skills

Students know the key protocols of the Internet and are able to explain their tasks and functionality in detail. They know which functions of the Internet architecture are implemented how and where in the network. They also understand the complex relationships between protocols and mechanisms on the Internet.

In addition, students can also apply the knowledge they have acquired in practice when developing of networked applications or the installation and maintenance of networks. The practical course enables students to use standard tools to analyze and set up applications and networks. analyze and set up applications and networks.

Reading list

Kurose, J.F./ Ross, K.W.: Computernetzwerke, 6. Auflage, Pearson Studium, 3/2014, ca. 900 Seiten, ISBN 978-3-8689-4237-8

2.13 Data communication in the vehicle

Information about the module

engl. Name	Data communication in the vehicle
Code	DAKOFZ4.WP
Coordinator	Prof. Dr.-Ing. Thomas Kirchmeier
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Data communication in the vehicle (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based online teaching and accompanying online internship to apply and deepen the knowledge acquired.
Prerequisites	C++ knowledge is not mandatory required as long as there is a willingness to deal with it as part of the crash course.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970369, 2970867 TI 3976576, 2976678 WI 3975787 IIS 9775112
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The course uses practical examples to illustrate the basic structure and functionality of a vehicle from a data transmission perspective. Individual vehicle functions are programmed in small teams, which then interact with each other using CommonAPI and SOMEIP. This simulates vehicle data communication and addresses the following topics:

- Crash course in C++ and cmake
- Use of a C++ GUI such as wxWidgets
- Vehicle architecture
- Implementation of simple vehicle functions in C++ and its visualization
- Basic communication systems in the vehicle, from fieldbus to IP communication
- SOMEIP and ServiceDiscovery
- Data processing and interface modeling
- Trace and error analysis
- Functional safety and dealing with "insecure" communication channels
- Time synchronization in the vehicle
- Security in vehicle communication

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- highlight the framework conditions for software development in the automotive sector.
- assess the background and structure of vehicle system architecture.
- plan different communication systems in the vehicle.
- evaluate the SOMEIP protocol and ServiceDiscovery.
- create SOMEIP interfaces using Franca and generate them using COMMONAPI.
- adapt the influences of safety, security and endianness on data communication.
- modify the mechanism of time synchronization via an asynchronous network.

Reading list

Matheus, K.; Königseder, T. Automotive Ethernet, Cambridge University Press; Auflage: 2 (13. Juli 2017), ISBN: 978-1107183223.

Völker, L. COMMUNICATION PROTOCOLS FOR ETHERNET IN THE VEHICLE. IQCP Congress, 09 –11 DECEMBER 2013, STUTTGART MARRIOTT HOTEL SINDELFINGEN, <https://www.iqpc.com/media/9048/29408.pdf>

Kirchmeier, T. Design and Qualification of Automotive Ethernet - An OEM Perspective. Automotive Ethernet Congress. Munich, Germany: 4-5 February 2015.

Kirchmeier, T. Automotive Ethernet: How to handle the difference between the standard and its implementation. IEEE Ethernet & IP @ Automotive Technology Day. Paris, France: 20-21 September 2016.

Völker, L. SOME/IP SERVICE DISCOVERY, Vector Symposium 2014-05-27, http://some-ip.com/papers/2014-05-27-DrLarsVoelker-The_need_for_Service_Discovery_in_the_vehicle.pdf

Overview of additional publications to SOMEIP and Service Discovery: <http://some-ip.com/papers.shtml>

2.14 Digital Innovation

Information about the module

engl. Name	Digital Innovation
Code	DIGINN4.WP
Coordinator	Prof. Dr. Christoph Buck
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester The module will not take place in WS2024/25.
Courses	Digital Innovation (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching to impart theoretical knowledge and skills and accompanying exercises with practical examples for interactive application and reflection on what has been learned in the sense of experience-based learning.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Paper presentation, 15 minutes, 10%• Presentation A, 20 minutes, 20%• Presentation B, 20 minutes, 20%• Final presentation, 30 minutes, 50%
Examination number	IN 3970382, 2970880 TI 3976580, 2976696 WI 3975800 IIS 9775116

Content of the module

- Introduction to the topic of "Digital innovations"
- Overview of digital innovation (in contrast to traditional innovation) and digital technologies as enablers of new business models
- Methods and structuring approaches for the analysis, design and new development of (digital) innovations and value creation mechanisms
- Overview of exciting traditional and new digital innovations and business models
- Transfer to practical application examples of regional companies

Qualification aims for the module learning objectives/skills

After successfully completing the "Digital Innovations" module, students will be able to

- reproduce definitions and structuring approaches to digital innovation
- describe characteristics of digital innovations and name differences to other types of innovation
- derive influences of digitalization on traditional business models and innovations
- apply frameworks, theories and innovation methods and tools (e.g. value proposition design) to develop and analyze digital innovations themselves
- assess and compare real-world examples of digital innovations from different perspectives

Students also learn soft skills such as teamwork, structuring and presentation skills.

Reading list

Osterwalder, A., Pigneur, Y. (2010) Business Model Generation Ein Handbuch für Visionäre, Spielveränderer und Herausforderer. Campus Verlag, Frankfurt am Main, 2010.

Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A. (2014) Value Proposition Design.

Nambisan, S., Lyytinen, K., Majchrzak, A., Song, M. (2017) Digital Innovation Management: Reinventing Innovation Management Research in a Digital World. Management Information Systems Quarterly, 41 (1), 223–238.

Kohli, R., Melville, N.P. (2018): Digital innovation A review and synthesis. Information Systems Journal, 29 (1), 200–223.

Weitere Literatur gemäß gesonderter Angabe.

2.15 Digital Business Leadership Skills

Information about the module

engl. Name	Digital Business Leadership Skills
Code	DIBUS.WP
Coordinator	Prof. Dr. Norbert Gerth
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Digital Business Leadership Skills (6 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, guest lectures, best practices, individual work, presentations
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 7.5, Contact hours: 90h, Independent study: 135h, Total workload: 225h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 30 minutes, 70%• Written assignment, 6-18 pages, 30%
Examination number	IN 3970346, 2970841 TI 3976578, 2976652 WI 3975758 IIS 9775114
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- The radical nature and speed of digitalization as a new megatrend poses major challenges for all industries (keyword 'disruption').
- This is not just about incorporating new key technologies.
- Rather, fundamental approaches and methods are changing, from research and development (agile, customer-centric innovation management) to HR management (team leadership and motivation) and the way in which companies interact with their customers in the future.
- All of this presents companies with major challenges.

This event focuses on the new approaches that need to be considered here. The students are asked to work on the practical content themselves as part of their coursework. The results will then be presented and discussed by all participants.

- Companies in the digital transformation
- Opportunities of disruption for startup founders
- Key digital technologies and their business potential
- New organizational concepts of established companies (digital units) and change management
- What established companies can learn from startups?
- Agile corporate management, leadership communication & Team productivity
- Opportunities and risks of startup-industry cooperation
- Methods of customer-centric product development (e.g. design thinking; lean startup)
- Innovation selling, acceleration and growth hacking
- Digital marketing and e-commerce

Qualification aims for the module learning objectives/skills

Students of the course should through their participation ...

- understand the relevance of digitalization for companies
- recognize the opportunities of disruption for startup founders
- learn to better assess the business potential of selected key digital technologies
- gain insights into newer management and organizational concepts of the DIG age
- get to know important methods of customer-centered product development
- recognize the challenges of marketing digital innovations
- receive tips on possible solutions in the context of digital marketing and e-commerce

With this in mind, this seminar will pay particular attention to the discussion of current and practice-relevant issues.

Reading list

The relevant literature is to be researched independently by the participants.

2.16 Digital Transformation in Organizations

Information about the module

engl. Name	Digital Transformation in Organizations
Code	DTO4.WP
Coordinator	Prof. Dr. Jens Lauterbach
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Digital Transformation in Organizations (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical group work and case studies, industry talks
Prerequisites	Students should have acquired basic skills in informatics or business information systems to understand core concepts/fundamentals behind business organizations and digital technologies. Bachelor (5th semester) or master in business information systems or computer science is recommended.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 50%• Written assignment, 10-15 pages, 50%
Examination number	IN 3970377, 2970875 TI 3976579, 2976686 WI 3975795 IIS 9775115
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Digitalization is one of the megatrends of our time. We live in a time where digital technologies and their applications make astonishing progress. Cars become driverless, computers beat humans in chess and Jeopardy and 3D-printers create houses. In the first part of this course the terms digitalization and digital transformation will be defined and the foundations are laid. Specifically, the following topics will be covered:

- Digital transformation – why it is one of the biggest buzzwords but also megatrends of our time
- Digitalization and digital transformation: Definition and delimitation
- A framework for organizations, individuals, and digital technology
- Historical evolution of industry and (digital) technologies
- Key digital technologies of our time
- Influence of digital technologies on organizations

Many organizations are confronting the question of how to design and manage the digital transformation. Based on phase-models of innovation adoption, the generic transformation process will be explained. Along this process, specific tasks and challenges that an organization needs to design and manage will be introduced. Specifically, the following topics will be covered:

- Stage models for digital transformation in organizations
- Key design aspects for digital transformations
- Methods and instruments to design, manage and facilitate digital transformations

Overall, this course is aimed at giving students the opportunity to learn and practice important aspects of digital transformations in organizations, one of the most pressing topics of our time for businesses around the globe. Group work with (research) papers and case studies will be used to complement the concepts and examples from the lecture. In industry talks, practitioners will share their own experiences from digital transformation management.

Qualification aims for the module learning objectives/skills

Students that aim at learning the design and management aspects of digitalization in organizations will create and deepen their knowledge. Students will be prepared for working in digital transformation projects in business organizations. After successful participation, students particularly will:

- Understand the term and the reasons for accelerated digital transformation in organizations
- Understand the technological and conceptual foundations of digital transformation
- Remember the historical evolution of industries and (digital) technologies
- Understand the influence of digital technologies on organizations
- Understand the typical phases and tasks in digital transformations
- Analyze and evaluate design and management problems in digital transformations
- Apply methods and instruments to create solutions for real world problems in the context of digital transformation projects

Reading list

Literature recommendations will be provided in the lecture

2.17 Introduction to Natural Language Processing

Information about the module

engl. Name	Introduction to Natural Language Processing
Code	EMSV4.WP
Coordinator	Dr. Phil. Alessandra Zarcone
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Introduction to Natural Language Processing (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching and accompanying exercises to apply and deepen the acquired knowledge. In addition, the exercises support and promote self-study.
Prerequisites	Fundamentals of computer science as taught in the foundation course.
Usage possibilities	Required elective for master's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: calculator
Additional information on the type of examination	for interactive media <ul style="list-style-type: none">• Written examination, 60 minutes, auxiliary: calculator, 70%• semester-accompanying work, 30%, alternatively:<ul style="list-style-type: none">- presentation (15-30 minutes)- student research project (6-10 pages)

Examination number	IN 3970378, 2970876 TI 3976583, 2976687 WI 3975796 IIS 9775119
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Morphological and syntactic units, modeling of structures and rules
- N-gram language models
- Text classifiers: naive-Bayes and logistic regression
- Words as vectors and embeddings
- Neural language models
- Sequence labeling & Named Entity Recognition
- Pre-trained language models
- Contextual embeddings
- Chatbots and dialog systems
- Data annotation for qualitative analysis and machine learning
- Evaluation of models and tools
- Industrial applications and societal implications

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- the typical challenges of natural language processing (ambiguity, context dependency)
- describe the current methods of machine language processing
- identify the appropriate technical solution for typical use cases and apply exemplary processing methods to simple examples

Reading list

Altinok, D.: Mastering spaCy: An end-to-end practical guide to implementing NLP applications using the Python ecosystem, 2021.

Carstensen, K.: Computerlinguistik und Sprachtechnologie: Eine Einführung, 2009.

Jurafsky, D.; Martin, J.H.: Speech and Language Processing. Entwurf der 3. Auflage. Available online <https://web.stanford.edu/jurafsky/slp3/>, 2021.

Software used:

- Python:
<https://www.python.org>
- Spacy:
<https://spacy.io/>

2.18 Introduction to Robotics

Information about the module

engl. Name	Introduction to Robotics
Code	EROB4.WP
Coordinator	Prof. Dr. Michael Strohmeier
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Introduction to Robotics (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying exercises for the introduction to robotics
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: non-programmable calculator, 2 DIN A4 pages handwritten formulary
Examination number	IN 3970400, 2970898 TI 3976623, 2976724 WI 3975818 IIS 9775178
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Overview of various fields of application in robotics
- Spatial representation: coordinate systems and homogeneous transformations
- Introduction to common control architectures in robotics
- Direct and inverse kinematics for mobile robots and manipulators
- Principles of dynamics using the example of simple robots and multicopters
- Overview of sensors in robotics and their measuring principles
- Sensor fusion: complementary filters and Kalman filters
- Mapping and localization, e.g. particle filters and SLAM
- Basic path planning and obstacle avoidance techniques
- Machine learning: introduction to reinforcement learning

Qualification aims for the module learning objectives/skills

After successfully completing the module, students understand the basic principles of robotics. They will be able to analyze and design simple robot systems in terms of their kinematics, dynamics and control structures. Students will be familiar with various sensor technologies and measurement principles. They understand and can apply the basics of sensor fusion. They understand basic algorithms for mapping, navigation and obstacle avoidance. They also understand the basics of machine learning techniques and are familiar with their application in robotics.

Reading list

Hertzberg J., Lingemann K., Nüchter A. *Mobile Roboter: Eine Einführung aus Sicht der Informatik*, Springer-Verlag, 1. Ausgabe, 2012

Siciliano B., Sciavicco L., Villani L., Oriolo G. *Robotics: Modelling, Planning and Control*, Springer, 1st Edition, 2008

Siegwart R., Nourbakhsh I.R., Scaramuzza D. *Introduction to Autonomous Mobile Robots*, MIT press, 2nd Edition, 2011

Sola, J. *Quaternion kinematics for the error-state Kalman filter*, arXiv preprint, 2017

Corke P.I., Witold J., Remo P. *Robotics, vision and control*, Springer, 2011.

2.19 Electronic Trading Systems

Information about the module

engl. Name	Electronic Trading Systems
Code	ELHS4.WP
Coordinator	Prof. Dr. Arne Mayer
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is offered irregularly or on demand.
Courses	Electronic Trading Systems (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching at the beginning - Supported by case studies, group discussions and guest lectures. In the further course work in small groups, in which the students can work out the practical content themselves. These results are then presented and discussed by all participants.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 15 minutes, 60%• Written assignment, 10-15 pages, 40%
Examination number	IN 3970376, 2970874 TI 3976565, 2976685 WI 3975794 IIS 9775120
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Electronic commerce (e-commerce) as part of e-business is becoming increasingly important and is pushing traditional, direct trade relationships into the background. This module examines the underlying IT systems - from a technical, business perspective:

- Sub-areas of e-business
- Technical/technological framework conditions of the Internet economy as drivers for e-business
- Structure and components of electronic trading systems
- Specifics of electronic commerce (e-commerce) such as Platform economy, revenue models
- Technological trends
- Analysis of existing electronic trading systems in practice: Modeling/documentation of their business processes using BPML
- Implementation of electronic trading systems: In small groups, students design and implement an e-shop - with the help of existing software or themselves (if desired and with appropriate prior knowledge!)

Qualification aims for the module learning objectives/skills

With successful participation in the module, students can:

- recognize and classify the importance of e-business and its sub-areas for the economy
- analyze and understand the characteristics and necessary processes of e-commerce and electronic trading systems in particular
- acquire implementation skills for a career or start-up
- present the results obtained in a target group-oriented manner

Reading list

Will be announced in the first classroom lecture.

2.20 Embedded Linux

Information about the module

engl. Name	Embedded Linux
Code	EMLI6.WP
Coordinator	Prof. Dr. Hubert Högl
Faculty	Faculty of Computer Science
Type	Wahlpflichtmodul
Duration / Frequency	1 semester, summer semester
Courses	Embedded Linux (6 SWS)
Teaching language	The module is taught in English.
Teaching and learning methods	<ul style="list-style-type: none">• Seminaristischer Unterricht• Praktische Übungen und Projekte
Prerequisites	Kenntnisse von Linux auf dem Desktop-Rechner, vor allem das Arbeiten auf der Kommandozeile (z.B. durch Wahlpflichtfach "LPIC") und Mikrocomputertechnik (z.B. Embedded Systems I und II) sind hilfreich, aber nicht zwingend notwendig.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	Fernklausur mit Videoaufsicht, 60 Minuten, none auxiliaries
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Motive für Linux auf eingebetteten Systemen
- Typische Hardware von Embedded Linux Rechnern
- Installation des Entwicklungsrechners
- Bootloader
- Linux Kernel
- Gerätetreiber
- Schnittstellen (UART, GPIO, SPI, I2C, ADC, PWM) und ihre Programmierung
- Anwendungsprogrammierung
- Filesysteme
- Debugging
- Echtzeit

Qualification aims for the module learning objectives/skills

Die Studierenden erlangen:

- Kenntnis des GNU/Linux Entwicklungsprozesses
- Verständnis der Funktion eines Gerätes auf der Basis von Embedded Linux
- Fähigkeit, eine eigene Produktidee in der Praxis mit Embedded Linux umzusetzen

Reading list

Chris Simmonds, Mastering Embedded Linux Programming, Packt Publishing 2015.

Rodolfo Giometti, GNU/Linux Rapid Embedded Programming, Packt Publishing 2017.

Weitere Informationen auf der Homepage von Prof. Högl
<http://hhoegl.informatik.hs-augsburg.de>

2.21 Flying Robots

Information about the module

engl. Name	Flying Robots
Code	FLUGRO4.WP
Coordinator	Prof. Dr. Constantin Wanninger
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Flying Robots (2 Credit hours) Practical Work Flying Robots (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying exercises for the introduction to flight robotics.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Oral examination, 30 minutes, 50%• Project work, Demonstration / Colloquium, 50%
Examination number	
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The lecture *Flight Robotics* imparts basic knowledge about unmanned flying robots through a combination of theory and practical exercises. The following contents are covered, among others:

- Fundamentals of flight robotics
- Sensors and actuators
- Cartesian coordinates and transformations
- The Robot Operating System (ROS)
- Path planning and collision avoidance
- Software engineering course

Finally, a small project with drones will be realized.

Qualification aims for the module learning objectives/skills

The lecture should enable students to understand the basics of flight robotics and apply them in practice.

- Understanding the basics of flying robots
- Practical experience with drone control and programming

Reading list

Online documentation of the Robot Operating System (ROS), <https://www.ros.org/>

Macenski, Steven: Robot operating system 2: Design, architecture, and uses in the wild, Science robotics (2022).

Gugan, Gopi: Path planning for autonomous drones: Challenges and future directions, Drones (MDPI) (2023).

Yang, Hyunsoo: Multi-rotor drone tutorial: systems, mechanics, control and state estimation, Intelligent Service Robotics (Springer) (2016).

2.22 Formula Student Driverless

Information about the module

engl. Name	Formula Student Driverless
Code	FSD4.WP
Coordinator	Prof. Dr. Gundolf Kiefer
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module has a duration of two semesters and is offered in the winter semester and the following summer semester if there is sufficient demand.
Courses	Formula Student Driverless (4 Credit hours)
Teaching language	German, in exceptional cases (international students) and at the competition events also in English
Teaching and learning methods	Project work, seminar, seminar-style teaching, regular status meetings
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written assignment, 10-15 pages, 80%• 6 Short presentations, 10-20 minutes, 20%
Examination number	IN 3970373, 2970871 TI 3976587, 2976682 WI 3975791 IIS 9775123
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The students take responsibility for a technical or organizational part of the development of a Formula Student driverless vehicle and develop the associated components together with a student team.

The development of a vehicle usually extends over one year (winter semester followed by a summer semester) and is divided into the following phases, each of which is concluded with a short presentation:

- Development of the requirements for the subsystem and coordination in the team (requirements freeze: presentation 1)
- Creation of a design and coordination of the interfaces with the adjacent components (design freeze: presentation 2)
- Implementation / production of the subsystem (presentation of prototype: presentation 3)
- Component / subsystem tests (presentation of the test results against the requirements: presentation 4)
- Integration of the component / subsystem into the overall system and execution of the integration tests (presentation of the integration test results with focus on the component / subsystem: presentation 5)
- Support of the subsystem during the race in the vehicle (presentation of success / outlook: presentation 6)

In addition to the actual presentations, regular team meetings are held to coordinate the procedure and determine the development status.

Qualification aims for the module learning objectives/skills

Knowledge:

- Students know the structure and architecture of the overall system in an autonomous electric racing vehicle.
- They know the development process and know how to complete it on time.
- They know how to integrate themselves into an interdisciplinary team and coordinate the technical and organizational interfaces.
- You know the importance of coordinated escalation of technical, scheduling and communication problems in your own development area, as well as at the interfaces to team members, suppliers and sponsors.

Skills:

- Students can lead a subsystem through the entire development process and know how to bring it to a level of maturity on time that ensures robust and safe operation in the vehicle during the race.
- Through contact with sponsors and partners from industry and the experience gained as a result, students can present themselves and their development results in English and German.

Competencies:

- Students are able to carry out risk assessments, prepare fallback solutions and decide in good time when these need to be used.
- As part of the team leadership for a subsystem, students assess the continuous progress and degree of maturity and can make well-founded technical decisions.

Reading list

- Formula Student Driverless and Formula Student Electric regulations
- Documentation of the FSD and FSE vehicles already developed by HSA

2.23 Fullstack Web Development

Information about the module

engl. Name	Fullstack Web Development
Code	FSWD6.WP
Coordinator	Prof. Dr. Wolfgang Kowarschick
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The duration of the module is one semester. The module is regularly offered in the summer semester. The module only takes place if enough participants register.
Courses	Fullstack Web Development (2 Credit hours) Practical work Fullstack Web Development (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, practical training
Prerequisites	Knowledge of the contents of the Data Management module(IA) is very useful, but not absolutely necessary.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Written assignment (duration 110 - 150 h), 90%• Presentation, 10%
Examination number	IN 3970368, 2970866 TI 3976589, 2976677 WI 3975786 IIS 9775125
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- programming
 - Basics of the ECMAScript (JavaScript) and TypeScript languages
 - Client programming (ECAMScript-/TypeScript-based), server programming (ECAMScript-/TypeScript-based), data storage (JSON format, RDBMS)
 - Communication between client and server (e.g. REST)
 - Development of simple web systems with the help of suitable frameworks.
- Programming principles
 - Modularization
 - Asynchrony (without threads)
 - Reusability (esp. don't repeat yourself, DRY)
 - Model-View-Controller-Pattern, Observer-Pattern ...
- collaborative work using Git

Qualification aims for the module learning objectives/skills

Knowledge:

The students can

- plan and implement a web project according to given prerequisites and requirements.

Skills:

The students can

- plan and program a REST API,
- plan and implement a relational database for a web project,
- design the front end of a web application according to the requirements and implement it with a current web framework,
- structure and version the development of a web project with the help of management software,
- deploy and manage their application online with the help of cloud platforms.

Competencies:

Students are able

- to independently familiarize themselves with new web technologies in order to keep pace with the rapid developments in this field.

Reading list

Script

Vue.js-Dokumentation:

<https://vuejs.org/v2/guide>

Phoenix-Dokumentation:

<https://hexdocs.pm/phoenix/overview.html>

PostgreSQL-Dokumentation:

<https://www.postgresql.org/docs/online-resources>

Deployment:

<https://devcenter.heroku.com>

Deployment:

<https://docs.netlify.com>

2.24 Fundamentals of DevOps

Information about the module

engl. Name	Fundamentals of DevOps
Code	DEVOPS4.WP
Coordinator	Prof. Matthias Kolonko, Ph.D. (ONPU)
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Fundamentals of DevOps (4 Credit hours)
Modul area	Applications
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, practical exercises
Prerequisites	Programming 1+2
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970387, 2970885 TI 3976590, 2976707, WI 3975805, IIS 9775167
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The lecture deals with the most important elements of the infrastructure for a structured software development process. Hereby various representatives of the different categories of these support tools are discussed and the differences are highlighted. The correct use and the correct application of these tools and their interaction will be highlighted.

In detail, the following categories and tools with the corresponding representatives will be considered:

Versioning Git, SVN, CVS, ...

Bug Tracker JIRA, Mantis, Redmine, ...

Build Tools Ant, Maven, ...

Continuous Integration Jenkins, ...

The area of ITIL will also be briefly discussed here, whereby the distinction between the above-mentioned bug trackers and ticket systems will be emphasized.

During the event, the different tools and their interlinking will also be interlocking will also be applied in practice. The systems will be installed, configured and tested with simple code examples.

Participants should also independently recognize and compare the advantages and disadvantages of application of these tools and compare them.

Qualification aims for the module learning objectives/skills

After successfully completing the module, participants will be able to

- name the current tools in the above categories.
- describe the advantages and disadvantages of the different tools.
- use the tools discussed in the lecture correctly.
- develop an integrated approach to the development of a software project using the different tools.

Reading list

Literature will be announced in the course.

2.25 Hard- and software for the internet of things

Information about the module

engl. Name	Hard- and software for the internet of things
Code	HARSO.WP
Coordinator	Prof. Dr. Volodymyr Brovko
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Hard- and software for the internet of things (2 Credit hours) Practical work Hard- and software for the internet of things (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, exercises, practical training
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: 1 DIN A4 page handwritten
Examination number	IN 3970347, 2970842 TI 2976653 WI 3975759 IIS 9775126
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Microcontrollers: typical components/use/programming in C and Python
- Typical interfaces (GPIO, UART, I2C, SPI), signal level, compatibility.
- Typical sensors (temperature, humidity, distance, acceleration, movement, ...)
- Typical actuators (servo, relay, DC motor, ...)
- MQTT protocol in Internet of Things / Raspberry Pi as MQTT broker / microcontroller WeMos D2 as MQTT client.
- Power supply in autonomous systems
- Example implementation of a sensor network

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to:

- Know how the microcontroller works and its interfaces.
- Program microcontrollers using the C programming language.
- Understand the operation of typical sensors and actuators.
- Create communication between multiple devices using the MQTT protocol in a network.
- Implement a simple data acquisition system with some sensors based on a simple microcontroller.

Reading list

Banzi, Massimo, 2015. Arduino für Einsteiger: 160/ST 170 B219 A6. ISBN: 978-3-95875-055-5,3-95875-055-9

Kofler, Michael, 2016. Raspberry Pi: 160/ST 160 K78(3).

Engelhardt, Erich F., 2016. Sensoren am Raspberry Pi: 160/ST 160 S294. ISBN: 978-3-645-60490-1

Hüning, Felix, 2016. Sensoren und Sensorschnittstellen: 160/ZQ 3120 H887. ISBN: 978-3-11-043854-3,3-11-043854-2,978-3-11-043855-0,978-3-11-042973-2.

Boyd, Bryan, 2014. Building Real-time mobile solutions with MQTT and IBM Message-Sight: ISBN: 978-0-7384-4005-7.

2.26 University Innovation Project

Information about the module

engl. Name	University Innovation Project
Code	HIP.WP
Coordinator	Prof. Dr. Alexander von Bodisco
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	ein Semester, jeweils im Winter- und Sommersemester
Courses	University Innovation Project (4 Credit hours)
Teaching language	The module is taught in German and English language.
Teaching and learning methods	Students work in small groups to develop IT solutions on a practice-oriented topic for an IT or interdisciplinary interdisciplinary project. The aim is to project process as realistically as possible with all facets as realistically as possible. The project topics are selected by authorized examiners of the Faculty of Computer Science and include a practical part (software/hardware), a documentation (student research project) and a presentation. The practical part (software and, if applicable, hardware) must be described as part of the student research project. The presentation usually takes place as part of a project project day or a seminar. The coordination with the project creator takes place regular face-to-face meetings and via electronic channels. The work is not necessarily bound to the lecture period.
Prerequisites	Solid knowledge of the most important areas of computer science, such as algorithms and data structures, programming, databases, data communication, software engineering and operating systems. The acquired knowledge should already have been practically applied in a team project.
Usage possibilities	Required elective for bachelor's degree programs Computer Science, International Information Systems, Computer Engineering, Information Systems
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 10-20 Seiten, 80%• Presentation, 10-20 minutes, 20%
Examination number	IN 3970401, 2970899 TI 3976624, 2976725 WI 3975819 IIS 9775179
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Students carry out independent small IT projects in groups or expand/support ongoing IT or interdisciplinary projects from computer science-related degree programs. The students' tasks include project management, software development and, depending on the project, independent familiarization with interdisciplinary topics.

In moodle you will find the current topics that are currently on offer:
<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Plan and execute software tasks in a team in terms of time, effort and resources.
- Apply software development methods in practice.
- Learn new software techniques independently and select suitable methods.
- Prepare interdisciplinary topics in self-study and develop questions.
- Document project results in an understandable and appealing way.

Reading list

Literature recommendations will be provided in the lecture.

2.27 Industrial Image Processing

Information about the module

engl. Name	Industrial Image Processing
Code	INDBV4.WP
Coordinator	Prof. Dr. Peter Rösch
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Industrial Image Processing (4 Credit hours)
University responsible for the module	TH Augsburg
Teaching language	The module is taught in German.
Teaching and learning methods	Participants work on content in self-study using textbooks and publications supported by instructional videos and instructions created by the lecturer. In the attendance part, students implement selected procedures and apply them to images from practice.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 90 minutes, none auxiliaries
Examination number	IN 3970389, 2970887 TI 3976593, 2976709 WI 3975807 IIS 9775169

Content of the module

Image processing is indispensable in automated industrial production, especially for quality assurance. During the course, students learn about the methods of industrial image processing and create their own applications using freely available tools and libraries.

- Fundamentals of image processing
- Image acquisition
- Image preprocessing
- Position recognition
- Label identification
- Presence control
- Measurement
- Surface inspections

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Describe common methods of industrial image processing verbally.
- Select and apply suitable tools from a program library to solve an image processing task.
- Systematically evaluate various given machine vision components for effectiveness and efficiency.
- Develop solutions for image processing tasks of medium complexity independently.

Reading list

C. Demant, B. Streicher-Abel, A. Springhoff: Industrielle Bildverarbeitung, 3. Auflage, Springer (2011)

W. Burger, M.J. Burge: Digitale Bildverarbeitung, 3. Auflage, Springer (2015)

R. C. Gonzalez, R. E. Woods: Digital Image Processing, 4th Ed., Pearson (2018)

J. Howse, J. Minichino: Learning OpenCV 4 Computer Vision with Python 3, 3rd Ed., Packt Publishing (2020)

scikit-image, Online-Dokumentation,
<http://scikit-image.org/docs/stable>

2.28 Industrial Data Processing

Information about the module

engl. Name	Industrial Data Processing
Code	INDIV4.WP
Coordinator	Prof. Dr.-Ing. Thomas Kirchmeier
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Industrial Data Processing (4 Credit hours)
University responsible for the module	TH Augsburg
Teaching language	The module is taught in German.
Teaching and learning methods	<p>The module is taught hybrid (face-to-face and online) and with flipped classroom as a teaching concept; means:</p> <ul style="list-style-type: none">• 2 SWS: The lecture content (instructional videos, presentations, assignments, etc.) is to be completed independently by the next lecture (completion time: 1 week).• 2 SWS: On the day of the lecture, questions about the lecture content will be discussed and, if necessary, further connections will be explained.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 90 minutes, Assistance: 1 self-written DIN A4 page (no copy)
---	---

Examination number	IN 3970402, 2970900 TI 3976625, 2976726 WI 3975820 IIS 9775180
--------------------	---

Grading	According to § 20 of the APO in the currently valid version.
---------	--

Content of the module

The lecture on industrial information processing looks at the derivation of knowledge from various data in relation to industrial applications, because data alone does not lead to recommendations for action per se. Only modeling from data with corresponding contextual knowledge enables prediction and thus control of systems and future decisions. Just as important as the modeling itself is the evaluation of results using stochastics to increase the significance and to exclude random influencing factors. Existing software tools will not be used for the time being, as a corresponding basic knowledge is required for their application. The partial aspects of the following topics required for the lecture are taught successively using simple examples:

- Stochastics (descriptive and evaluative statistics and probability theory)
- Numerical algorithms for the computer-aided solution of ordinary differential equations
- Ordinary differential equations with initial and boundary value problems
- Mathematical optimization algorithms (least square, recursive least square, gradient descent methods)
- Stability considerations

The example models and tasks are implemented in Python. With regard to the models, only the basic approaches of regression equations and neural networks are considered. In addition to static data processing, approaches to dynamic and adaptive control of unknown systems are also discussed (MRAC - model reference adaptive control).

Qualification aims for the module learning objectives/skills

At the end of the event, the participants should

- be able to determine independent dependent data and labels,
- have a basic understanding of how to create models from data,
- apply cost functions and learning algorithms to adapt model parameters,
- be able to analyze model and adaptation stability of learning algorithms,
- assess model results and statements with the help of stochastics,
- be able to adaptively control simple dynamic and non-linear processes.

Reading list

J. D. Kelleher, B. Tierney: Data science, MIT Press, Cambridge, 2018.

J. Cleve, U. Lämmel: Data Mining. De Gruyter: Berlin, 2020.

J. Starmer: The StatQuest Illustrated Guide To Machine Learning. StatQuest Publications, 2022.

G. C. Goodwin, K. S. Sin: Adaptive Filtering Prediction and Control. Dover Books on Electrical Engineering, Dover. 1984

G. Schulz: Regelungstechnik, Mehrgrößenregelung - Digitale Regelungstechnik - Fuzzy-Regelung. Oldenbourg, 2002.

H.-J. Reinhardt: Numerik gewöhnlicher Differentialgleichungen, Anfangs- und Randwertprobleme, De Gruyter, 2012.

2.29 Information technology and the environment

Information about the module

engl. Name	Information technology and the environment
Code	INUM4.WP
Coordinator	Prof. Dr. Jürgen Scholz
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Information technology and the environment (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	<p>In group work, the knowledge gained is then specified and prepared for an INFO-Shop. Based on these results, small tasks are developed for teams of 2-4 people and worked on as part of a project.</p> <p>At the end of the semester, a computer science & environmental fair is planned, in which each project group sets up its "exhibition stand" and presents the results to interested parties.</p>
Prerequisites	Basic knowledge of computer science (Programming, Introduction to Computer Science)
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	<p>Portfolio exam:</p> <ul style="list-style-type: none">• Elaboration, 35%• Presentation, 15%• Participation in the overall project, 50%
---	---

Examination number	IN 3970393, 2970891
	TI 2976713
	WI 3975811
	IIS 9775170

Grading	According to § 20 of the APO in the currently valid version.
---------	--

Content of the module

The challenge of protecting our environment (air pollution, global warming, ...) affects everyone. Another aspect is to reduce our dependence on imports - especially of fossil fuels. Everyone has a responsibility to bear, including us as technical computer scientists, computer scientists and business computer scientists. What opportunities are there for us computer scientists to make our contribution? What can we achieve? This is the motto of the event "Informatics and the environment".

First, an overview of the subject area of computer science and the environment is given. This includes an introduction to the physical/electrotechnical relationships from a practical perspective. This is not a physics lecture, but a summary of what you need to know to be able to work here as a computer scientist. So it's not Einstein's theory of relativity, but a pragmatic summary according to Jürgen Scholz.

After this introduction, we quickly move on to practical topics, where students work on smaller topics themselves in small teams. The students research the topic in question based on the material provided. They work on the topic and create a poster for an information store based on their findings. In the "Info-Shop", the students show the results of their team to the other teams using the poster. If possible, the preparation for the Info-Shops and the Info-Shop should take place on the same day.

Concrete, semester-spanning project topics are derived from the Info-Shop work and topics, which are also developed in teams. The semester project can range from practical, tangible topics (building a small circuit that saves energy, programs, apps) to theoretical evaluations. A list of suggestions for topics will be provided. The only requirement: the topic context of the lecture must be recognizable in the topic and the processing.

If possible, the results will be presented on a larger scale (possibly on the project day).

Special feature:

In addition to - and as part of - the event, presentations by speakers from industry and authorities are planned, which will show some of the approaches already being pursued by industry in the various areas.

At the end of the semester, a computer science & environment fair is planned in which the students will present their projects to other interested parties.

The documentation of the teams' results will be bound together in a document at the end of the semester.

Qualification aims for the module learning objectives/skills

The student learns about the areas in which computer science has an impact on the environment.

The student has the ability to carry out theoretical or practical projects, from the design to the construction of small devices, software or economic assessments or environmental impact assessment systems, etc.

He is sensitized to the environmental aspect of his work in his life as a computer scientist.

Reading list

Literature recommendations will be provided in the lecture.

2.30 Integrated Business Processes with SAP ERP

Information about the module

engl. Name	Integrated Business Processes with SAP ERP
Code	SAPERP4.WP
Coordinator	Dipl.-Ing. Harald Röser
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Integrated Business Processes with SAP ERP (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, exercises, practicals
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 90 minutes, none auxiliaries
Examination number	IN 3970321, 2970782 TI 3976543, 2976555 WI 3975702 IIS 9775129
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Overview of the components of an ERP system and the basics of key logistics processes and their integration.

Qualification aims for the module learning objectives/skills

Students should be able to do the following:

- describe the core functions of SAP ERP
- name the components of a business process
- explain the individual process steps
- the organizational levels used in the business process
- and list the master data
- recognize the integration points of a process

Reading list

Will be announced at the beginning of the first course of the module.

2.31 Interaction Engineering

Information about the module

engl. Name	Interaction Engineering
Code	INTENG4.WP
Coordinator	Prof. Dr. Michael Kipp
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Interaction Engineering (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	The course includes a series of lectures by the lecturer. Students will give oral presentations and work on assignments at home, both individually and in teams. Students will also work on a final team project which engages them in scientific thinking, practical implementation and critical reflection.
Prerequisites	The requirements for this course are solid programming skills, prior experience with working scientifically, a good command of the English language (reading, writing and speaking) and an interest in working both analytically and creatively to develop novel interaction methods.
Usage possibilities	Required elective for bachelor's degree programs: Computer Science and Computer Engineering
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 15 minutes, 25%• Project work, 50%• Written assignment, 15-20 pages, 25%
---	--

Examination number IN 3970326, 2970796
TI 3976567, 2976571

Grading According to § 20 of the APO in the currently valid version.

Content of the module

In the course students will learn about fundamental concepts of human-computer interaction and various research areas that try to improve traditional ways of human-computer interaction by including touch, gesture, facial and bodily actions to make the interaction more intuitive, natural and efficient.

Students will also get to know and apply methods to evaluate interactive systems objectively (measurable aspects) and subjectively (user feedback).

Qualification aims for the module learning objectives/skills

Knowledge

- Fundamentals of human-computer interaction
- Touch interaction
- Gestural interaction
- Tangible interaction
- Proxemic, spatial, full-body interaction
- Cross-device interaction

Skills

- Understanding and presenting a research publication
- Implementing a running prototype of an interactive system
- Applying evaluation methods for an interactive system
- Critically discussing research publications
- Working in a team

Competencies

- Understanding and further developing a research topic
- Informally evaluating a prototype

Reading list

- B. Buxton, S. Greenberg, S. Carpendale, N. Marquardt (2012)** Sketching User Experiences: The Workbook, Morgan Kaufmann, 262 pages.
- B. Albert, T. Tullis (2013)** Measuring the User Experience, 2. Edition, Morgan Kaufmann, 301 pages.
- J. Butler, K. Holden, W. Lidwell (2010)** Universal Principles of Design, Rockport Publishers, 272 pages.

2.32 Interactive Computer Graphics

Information about the module

engl. Name	Interactive Computer Graphics
Code	IACOGR6.WP
Coordinator	Prof. Dr. Peter Rösch
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The duration of the module is one semester. The module will be offered in the summer semester if there are enough registrations
Courses	Interactive Computer Graphics (6 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, exercises Programming languages and interfaces used: Python (panda3d and WorldViz Vizard) OpenGL Shading Language (GLSL) JavaScript (babylon.js)
Prerequisites	Linear algebra (matrices, vectors, transformations)
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h Online participation in the attendance part is possible.

Exam

Type of exam / required course achievements	Written examination, 90 minutes, none auxiliaries
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Summary

The performance of current hardware makes it possible to run sophisticated interactive graphics applications not only on specially equipped computers, but increasingly also on mobile devices. At the same time, 3D content can be presented directly in the web browser without installing specific software, so that the importance of computer graphics, e.g. for the visualization of complex content or for the presentation of products, will continue to grow.

The course is divided into three parts. First, basic methods and algorithms of computer graphics are introduced and practically applied using the platform-independent OpenGL interface, whereby the graphics hardware is also controlled directly with your own shader programs.

Equipped with these basics, we enter "virtual reality" and use the 3x2m projection screen in the 3D visualization lab in combination with an optical tracking system to interact with stereoscopically displayed 3D models. The "WorldViz Vizard" software used reduces the programming effort considerably and allows the user to concentrate on setting up the scene, physics simulation and interaction.

Finally, the WebGL interface is introduced and used to display 3D content platform-independently in the web browser.

- Geometry - objects and transformations
- Virtual camera, projections
- Lighting and shadows
- Textures and advanced surface effects
- Interaction with the user
- Shader programming
- Stereoscopic output
- 3D tracking
- Physics simulation
- Interactive 3D graphics in the web browser

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Define basic terms in computer graphics.
- Explain algorithms for displaying scenes.
- Combine components from libraries to create computer graphics applications of medium complexity.
- Evaluate source code, especially with regard to efficiency.
- Implement interactive computer graphics applications independently.

Reading list

T. Akenine-Möller et al.: Real-Time Rendering, 4th Ed., CRC Press (2018)

D. Wolff: OpenGL 4 Shading Language Cookbook, 3rd Ed., Packt Publishing (2018)

J.D. Foley, A. van Dam, S.K. Feiner: Computer Graphics – Principles and Practice, Addison Wesley, 3rd Ed., Pearson (2014)

R. J. Rost, J. M. Kessenich, B. Lichtenbelt: OpenGL Shading Language, 3rd Ed., Addison Wesley (2009)

2.33 IT-Consulting

Information about the module

engl. Name	IT-Consulting
Code	ITC4.WP
Coordinator	Prof. Dr. Stephan Zimmermann
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	IT-Consulting (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching with accompanying exercises and case studies to apply and deepen the knowledge acquired. In addition the exercises support self-study.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 10-25 Seiten, 60%• Presentation, 10-30 minutes, 40%
Examination number	IN 3970379, 2970877 TI 3976595, 2976688 WI 3975797 IIS 9775131
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Consulting skills are key requirements for anyone who introduces and develops information systems and digital technologies. The consulting industry itself is a billion-dollar business and attracts many university graduates. However, in-house consultants who provide consulting services in their own companies are also in demand. In the context of digital transformation, IT consulting is therefore a major topic for the future:

- In the analysis and introduction of innovative information technologies,
- in the integration of information systems and business processes and
- in the management of IT in the company.

In this module, the techniques, personal skills and challenges of IT consultants are examined and applied:

- Fundamentals, structures and objectives of business and IT consulting
- Service offerings in the field of IT consulting
- Phases in the IT consulting process: project acquisition, market research, project management, business analysis, presentation of results
- Analytical methods and techniques in IT consulting projects (e.g. hypothesis-based problem-solving, ideation & design thinking, business model analysis, business process reengineering & process modeling, information systems analysis, requirements engineering, solution design, ...)
- Methods of IT consulting: management skills, research and analysis techniques, workshop, conference and meeting design, moderation techniques, presentation, slide deck visualization
- Profile of the IT consultant: know-how, social & team skills

Qualification aims for the module learning objectives/skills

After successful participation in the module, students can:

- Classify the objectives, processes and challenges of IT consulting companies.
- Discuss the tasks and methods in IT consulting.
- Perform and adapt project management, business analysis and consulting methods with regard to IT consulting projects.
- Analyze and model corporate issues in the use of information systems and technologies.
- Conduct workshops, conferences and meetings in consulting projects.
- Plan and organize consulting assignments based on case studies.

Reading list

Cadle, James; Paul Debra; Turner Paul (2014): Business Analysis Techniques – 99 Essential Tools for Success (2. Auflage). BCS, The Chartered Institute for IT

Conn, Charles; McLean Robert (2018): Bulletproof Problem Solving. Hoboken, New Jersey: John Wiley & Sons, Inc.

Hamilton, Pamela (2016): The Workshop Book – How to design and lead successful workshops. Pearson

Lippold, Dirk (2020): Grundlagen der Unternehmensberatung (2. Auflage). Berlin/Boston: De Gruyter

Weiss, Alan (2021): The Consulting Bible (2. Auflage), Wiles

Williams, Robin (2017): Non-Designer's Presentation Book, The: Principles for effective presentation design, 2nd Edition, Peachpit Press

2.34 IT Forensics

Information about the module

engl. Name	IT Forensics
Code	ITFORE4.WP
Coordinator	Prof. Dr. Kay Werthschulte
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Basics of IT Forensics (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching and accompanying work placement to apply and deepen the knowledge acquired. In addition, the practical course supports and promotes self-study.
Prerequisites	Lecture IT Security desirable but not an exclusion criterion
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 90 minutes, none auxiliaries
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Introduction to digital forensics
- Procedure models
- Securing digital traces
- Analysis of digital traces
- Hard disk forensics
- Windows forensics
- Memory forensics
- Network forensics
- Mobile forensics
- Malware analysis
- Presentation of evidence in court
- Legal aspects

Qualification aims for the module learning objectives/skills

The Digital Forensics lecture deals with securing, analyzing and presenting digital evidence after an incident. Students are given an overview of forensic procedures, IT attacks and the underlying technologies.

As this is an integrated lecture, what is learned is applied directly in the lecture, thus achieving a close link between theory and practice.

After the lecture, participants should be able to determine whether an attack has taken place and know how to secure, analyze and properly present digital evidence in court.

Reading list

Dan Farmer, Wietse Venema: Forensic Discovery, Addison-Wesley Longman, Amsterdam; Auflage: illustrated edition (13. Januar 2005)

Brian Carrier: File System Forensic Analysis, Addison-Wesley Longman, Amsterdam (7. April 2005)

Harlan Carvey: Windows Forensic Analysis DVD Toolkit, Second Edition, Syngress; 2 edition (June 11, 2009)

Lee Reiber: Mobile Forensic Investigations, McGraw-Hill Education, Auflage: 2., 2019

2.35 IT Security

Information about the module

engl. Name	IT Security
Code	ITSICH4.WP
Coordinator	Prof. Lothar Braun Prof. Dr.-Ing. Dominik Merli
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The duration of the module is one semester. The module is offered regularly in both the winter semester and the summer semester.
Courses	IT Security (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying exercises and presentations to apply and deepen the acquired knowledge
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Fundamentals of IT security
 - Basic terms
 - Relevant standards
 - Typical attacks
 - Security processes
 - Analysis of threats and risks
- Cryptographic basics
 - Symmetric encryption
 - Hash functions
 - Asymmetric cryptography
 - Key management
 - Security protocols
- Application-related security
 - Embedded systems
 - Networks
 - Web applications

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- explain the basic concepts of IT security.
- describe typical attacks.
- apply the methodology of threat and risk analysis to a scenario.
- present the basics of cryptographic algorithms.
- implement simple cryptographic applications.
- analyze simple security properties of networks, devices and web applications.
- plan simple security measures for networks, devices and web applications.

Reading list

- A. Shostack:** "Threat Modeling: Designing for Security", Wiley, 2014
- M. Howard, S. Lipner:** "The Security Development Lifecycle", Microsoft Press, 2006
- C. Paar, J. Pelzl:** "Understanding Cryptography: A Textbook for Students and Practitioners", Springer, 2010
- C. Eckert:** "IT-Sicherheit: Konzepte - Verfahren - Protokolle", Oldenbourg, 2012
- M. Ruef:** "Die Kunst des Penetration Testing", C & L, 2007

2.36 IT Sourcing and Cloud Transformation

Information about the module

engl. Name	IT Sourcing and Cloud Transformation
Code	ITSCT4.WP
Coordinator	Prof. Dr. Arne Mayer
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	IT Sourcing and Cloud Transformation (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar-based instruction at the beginning - Supported by case studies, group discussions and guest lectures. In the further course, work in small groups, in which the students work out the practice-relevant content themselves.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: non-programmable calculator
Examination number	IN 3970380, 2970878 TI 3976596, 2976689 WI 3975798 IIS 9775133
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Offshoring and outsourcing as well as the change from classic IT models to the cloud are a 'must have' for organizations in high-wage countries like Germany. This stems not only from an economic point of view, but also against the background of the permanent shortage of IT specialists. As a result, complexity and demands on the IT of organizations increase significantly. In this module - with a strong focus on relevant, current problems - students are prepared for opportunities and challenges in their future professional life.

The following blocks are covered:

- Off- and nearshoring (regional IT sourcing)
- Outsourcing (external IT sourcing)
- Transformation to the Cloud / Everything as a Service
- Low code platforms as game changers in software development

Qualification aims for the module learning objectives/skills

With successful participation in the module, students can:

- Understand the challenges in today's information management
- Be familiar with and discuss the IT measures and technologies mentioned
- Generate solution proposals for current problems and create implementation approaches

Reading list

Will be announced in the first lecture.

2.37 Classic Project Management Modernized

Information about the module

engl. Name	Classic Project Management Modernized
Code	KLPRO.WP
Coordinator	Prof. Dr. Wolfgang Kowarschick
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Classic Project Management Modernized (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching with the use of Worksheets to deepen the acquired knowledge.
Prerequisites	None
Usage possibilities	Compulsory elective subject for all Bachelor's degree programs, except IA
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written assignment, 10 pages
Examination number	IN 3970371, 2970869 TI 3976598, 2976680 WI 3975789 IIS 9775135
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

At the beginning of the course, the key terms of critical chain project management are defined: Project objectives, project participants, management tasks (people management, risk management, planning, control) and project success. After an introduction to risk management, the course of the project is examined in more detail: phases and processes, waterfall and spiral model, V-model XT. Building on this, various estimation methods and their advantages and disadvantages are presented. Common planning techniques are then discussed: Work Breakdown Structures, network diagrams, bar charts, cost planning. A key topic is the critical chain method (instead of the critical path) and the associated buffer management (as a very important component of risk management). Finally, the topics "Project control based on buffer management" and "Earned value analysis" are discussed.

Parallel to the classic planning and control topics, the importance of people management is repeatedly emphasized throughout the semester. Important aspects here are: leadership styles, teamwork, motivation and avoiding pressure

Qualification aims for the module learning objectives/skills

Knowledge:

- Students know the key terms and objectives of critical chain project management.
- Students know the differences between classic and agile project management.
- They are aware that explicit buffer management can be used profitably in both areas.
- They are aware that agile project management can only be used in certain areas of a project that is not exclusively based on software development.
- They are also aware of the typical management errors that are often responsible for the failure of a project.
- Students are familiar with the documentation architecture of V-Modell XT.

Skills:

- Students can successfully carry out media projects as project collaborators.
- Students can contribute to the planning of a project so that all project objectives (duration, costs, functionality, quality) are likely to be met. In particular, they can apply the principles of explicit buffer management profitably.
- Students can assess project risks, take suitable precautionary measures and, if necessary, appropriate countermeasures.
- They can create project documentation according to the specifications of V-Modell XT.
- They can adapt V-Modell XT specifications to specific projects (tailoring).

Competencies:

- Students can justify the decisions they make as project team members.
- They can categorize and evaluate a variety of project techniques.

Reading list

A very comprehensive script and digital documents will be provided for the lecture.

2.38 Concepts of Database Technology

Information about the module

engl. Name	Concepts of Database Technology
Code	KDBT4.WP
Coordinator	Prof. Dr. Michael Predeschly
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Concepts of Database Technology (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying work placement to apply and consolidate the knowledge acquired. In addition, the practical course supports and promotes self-study.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970397, 2970895 TI 3976545, 2976717 WI 3975815 IIS 9775175
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The lecture introduces various concepts that are used in different types of databases. Different architectures are presented.

One focus of the course is the storage of data. The following topics are covered:

- Storage structures and access paths
- buffer management
- insertion strategies
- Indexes

A second central aspect is dedicated to the consistency of databases using:

- Transactions
- Concurrency control
- Serializability
- recovery
- Schema migration

In addition, the topic of query optimization is considered both algebraically and algorithmically.

Finally, concepts of data protection and data security in databases are examined.

Both theoretical basics are taught and their application in practice is demonstrated and implemented.

Qualification aims for the module learning objectives/skills

Students gain an overview of the subject area of various database technologies. After successfully completing the module, students acquire the following skills

- Knowledge of the architectural principles important for the implementation of database systems
- Understanding of data structures and algorithms and the ability to compare, analyze, evaluate and implement them
- In-depth understanding of the design and internal structures of a complex software system
- Optimization of the operation of database systems
- Planning a database system and its secure operation
- Concepts and techniques of data protection and data security

Reading list

A bibliography will be provided during the course.

2.39 Artificial intelligence in safety-critical applications

Information about the module

engl. Name	Artificial intelligence in safety-critical applications
Code	KISICH4.WP
Coordinator	Dr. Marc Zeller (Siemens AG, München)
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Artificial intelligence in safety-critical applications (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching and accompanying exercises
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: calculator
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Fundamentals of functional safety
 - Basic terms
 - Safety engineering life cycle
 - Risk analysis and safety classification
 - Safety verification and certification
- Safe software development in different industrial domains
 - Safety concepts and error analysis methods
 - Test and verification methods for secure software
 - Relevant standards and their practical application
- Secure and robust artificial intelligence (AI)
 - AI and ML = Software 2.0
 - Relevant standards
 - Safety Of The Intended Functionality (SOTIF)
 - Analysis methods of AI/ML models with regard to robustness, uncertainty and transparency
 - Out-of-Distribution Detection and Runtime Monitoring
 - Iterative and agile development (MLOps) and security

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- explain basic concepts of functional safety for software, artificial intelligence (AI) and machine learning (ML)-based systems
- describe aspects of functional safety and SOTIF aspects of autonomous systems in different industrial domains
- apply methods of risk and error analysis and derive requirements for the safety of the system
- apply test and verification methods for safe software
- present the basics of robustness, uncertainty and transparency analyses on AI/ML models
- create safety concepts for the development and operation of safe and robust autonomous systems

Reading list

Books:

Laprie, Jean-Claude: Dependability: Basic concepts and terminology. Springer Vienna, 1992.

Koopman, Phil: How Safe is Safe Enough?: Measuring and Predicting Autonomous Vehicle Safety. Carnegie Mellon University, 2022.

Wolfgang Ertel: Grundkurs Künstliche Intelligenz – Eine praxisorientierte Einführung, Springer Verlag, Wiesbaden.

Standards:

- Automotive (ISO 26262-6)
- Railway (EN 50128, EN 5065, SIRF)
- Avionics (DO-178C)
- Medical Devices (IEC 62304)
- Industry Automation (ISO 13849)
- Artificial Intelligence (EU AI Act, UL4600, VDE-AR-E_2842-61-5)

2.40 Lean IT & Enterprise Architecture

Information about the module

engl. Name	Lean IT & Enterprise Architecture
Code	LEANIT4.WP
Coordinator	Prof. Dr. Stephan Zimmermann
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Lean IT & Enterprise Architecture (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Lecture and seminar lessons with laboratory exercises and case studies to apply the knowledge acquired. In addition, the exercises support self-study.
Prerequisites	The requirements for this course are a basic command of the English language, and an interest in better managing IT organizations and enterprise architectures.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Project work, 15-30 pages
Examination number	IN 3970394, 2970892 TI 3976600, 2976714, WI 3975812, IIS 9775171
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

IT in companies is becoming more and more important and complex. A numerous and increasing number of applications, systems and IT services used in business processes and delivered by IT organizations substantiates this development.

Lean IT and Enterprise Architecture Management (EAM) help companies to address related challenges. While Lean IT uses lean principles to develop and manage IT products and services with the central concern to eliminate waste in the context of IT that adds no value for the customer or user, EAM describes the management practice to transform the IT landscape by defining, communicating, and using a coherent set of strategies and guidelines.

In this course students will learn about the fundamental concepts of lean IT and enterprise architectures, and how these two topics connect. They also get to know techniques to develop strategies, analyze waste and work in value streams, and build business, information system and technology architectures.

Students will play several lean games to increase their lean mindset and solve several case studies regarding enterprise architecture challenges in practice. Supported by the novel “The Phoenix Project” they will have an additional touchpoint to practical challenges.

Knowledge focus:

- Lean IT concepts (value, waste, value streams, pull, flow)
 - Value stream mapping
 - The Four Types of Work
 - Kanban-Boards
- Enterprise Architecture concepts: Business, Information System and Technology Architecture
 - Business Capability Management
 - IT Portfolio Management
 - The Open Group Architecture Framework (TOGAF)
 - Visualization of IT landscapes

Qualification aims for the module learning objectives/skills

After successful participation in the module, the students can:

- illustrate waste, work, and Kanban in a lean IT context
- apply value stream mapping for IT services & products
- demonstrate competencies with the application of EA methods and IT landscape modelling
- apply business capability management and IT portfolio techniques
- illustrate enterprise architecture frameworks
- solving practical case studies and scenarios
- articulate course related ideas and concepts in English.

Reading list

Ahlemann, F., Stettiner, E., Messerschmidt, M., Legner, C. (2012): Strategic Enterprise Architecture Management Challenges, Best Practices, and Future Developments, Springer-Verlag Berlin Heidelberg.

Kim, Gene; Behr, Kevin; Spafford, George (2013) : The Phoenix Project – A novel about IT, DevOps and helping your business win, IT Revolution Press.

Lankhorst M. (2013) : Enterprise architecture at work: Modelling, communication, and analysis. Springer, Berlin.

Peppard J., Ward J. (2016) : The strategic management of information systems: Building a digital strategy. Wiley, Chichester, West Sussex.

The Open Group (2018), The Open Group Architectural Framework (TOGAF) Version 9.2. The Open Group, Reading, UK.

2.41 Linux LPIC

Information about the module

engl. Name	Linux LPIC
Code	LINLP14.WP
Coordinator	Dieter Thalmayr
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course in both the winter and summer semesters. Note: The course WPF Linux LPIC is offered as a block course on 6 days. The examination takes place outside the usual examination period at an appropriate distance from the block course. A Saturday is planned as the examination day.
Courses	Linux LPIC (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, exercises
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The content of Linux LPIC is based on the topics that the Linux Professional Institute has defined for the LPI 101 and 102 exams:

- System architecture
- Installation and package management
- GNU and UNIX commands
- Devices, Linux file system, file system, hierarchy standard
- Data management and rights concept
- Simple administrative tasks
- Learning a Linux editor
- Package management
- GNU and UNIX commands
- Shells and basics of script programming
- Administrative tasks
- Network basics
- Setting up a network service
- Security

Qualification aims for the module learning objectives/skills

Participants should gain an insight into how GNU/Linux works, as well as learn advanced operation and basic administration of Linux computers. At the end of the block, participants can optionally take a "Linux Professional Institute" (LPI) exam to confirm their knowledge with a "LPIC" certificate, which is highly regarded in the industry.

Reading list

Training material from tuxcademy:

www.tuxcademy.org (free of charge)

Harald Maassen, LPIC-1. Sicher zur erfolgreichen Linux Zertifizierung, Galileo Computing, latest edition.

(will be announced in the lecture)

Further information:

<http://hhoeigl.informatik.hs-augsburg.de/hhweb/lpic>

<http://www.lpice.eu/de>

2.42 Artificial Intelligence

Information about the module

engl. Name	Artificial Intelligence
Code	METHKI6.WP
Coordinator	Prof. Dr. Thomas Rist
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Artificial Intelligence (4 Credit hours) Practical work Artificial Intelligence (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	<p>Seminar-based teaching to convey the theoretical foundations of conceptual solution approaches, which are worked on together with the students on the basis of selected problems.</p> <p>In an accompanying practical course, students work independently on a specific application task (e.g. from the field of machine learning and data mining, robotics, expert systems, game AI).</p> <p>The type of teaching is lecture and practical course.</p>
Prerequisites	Fundamentals of computer science at undergraduate level and familiarity with a programming language (e.g. Java, Python, C++)
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	ePortfolio
---	------------

Examination number	IN -, - TI -, - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Basics

- AI concept from a scientific, technical and social perspective
- strong versus weak AI
- Modeling intelligent capabilities in technical systems

Problem solving as a search task

- Modeling of application problems
- Search methods (e.g. A*, MinMax, strategy games)
- Modeling with constraints and constraint solver

Knowledge-based systems

- Rule systems, expert systems
- Logical reasoning, logic calculi, SAT solvers
- Knowledge representation, ontological modeling
- Procedures for action planning
- Probabilistic reasoning, Bayesian networks, fuzzy inference

Learning systems, machine learning and data mining

- Basic concepts: supervised / unsupervised learning, symbolic / non-symbolic approaches
- Clustering, classification, knowledge discovery
- reinforcement learning
- Neurocomputing and artificial neural networks
- Approaches to deep learning

Outlook on current and emerging fields of research

Qualification aims for the module learning objectives/skills

Software tools and libraries are presented for the topics covered, with which practical problems can be solved.

- have a sound overview of common AI methods and AI techniques,
- know typical fields of application in which AI techniques are used,
- are able to work on selected problems with suitable AI methods.

Reading list

Stuart Russel, Peter Norvig: Künstliche Intelligenz Pearson Studium – IT, Gebundene Ausgabe, 2012.

Wolfgang Ertel: Grundkurs Künstliche Intelligenz: Eine praxisorientierte Einführung. 4. Auflage, Springer Verlag 2016.

Jürgen Cleve, Uwe Lämmel: Data Mining. De Gruyter Studium, Taschenbuch 2014.

Peter Buxmann, Holger Schmidt (Hrsg.): Künstliche Intelligenz: Mit Algorithmen zum wirtschaftlichen Erfolg, Springer Gabler, 2018.

Further literature (including current specialist publications) will be recommended during the lecture as appropriate to the topics discussed.

2.43 Mobile Robots

Information about the module

engl. Name	Mobile Robots
Code	MOBRO2.WP
Coordinator	Prof. Dr. Constantin Wanninger
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Mobile Robots (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	
Prerequisites	
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 2, CP credits: 5, Contact hours: 30h, Independent study: 120h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 5 minutes, 3 slides, 25%• Written examination, 45 minutes, (closed book), 50%• Project work, presentation / documentation / UML-models, 25%
Examination number	IN -, - TI - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The elective module combines theoretical lectures with a written examination and practice-oriented projects in block format, to ensure a comprehensive understanding of the topics covered. Participants acquire both in-depth specialist knowledge and practical experience, which prepares them optimally for future challenges. The following topics are covered, among others:

- Basics of mobile robotics
- Electronics fundamentals for mobile robots
- Robot navigation
- Control algorithms
- Autonomy and decision making

Finally, a small project with mobile robots is being realized.

Qualification aims for the module learning objectives/skills

This compulsory elective module provides students with in-depth specialist knowledge in the field of mobile robotics and the ability to apply this independently and creatively to a given project. They learn to analyze complex issues and develop innovative solutions that they implement as part of the project. In doing so, they strengthen their self-organization, teamwork and communication skills.

Reading list

Online documentation of the Arduino platform, <https://www.arduino.cc/>

Banzi, Massimo: Getting Started with Arduino. Maker Media, Inc. (2022).

Purdum, Jack: Arduino C, Springer (2012).

Maier, Helmut: Grundlagen der Robotik, VDE Verlag GmbH (2016).

2.44 Pattern recognition and machine learning

Information about the module

engl. Name	Pattern recognition and machine learning
Code	MKML4.WP
Coordinator	Prof. Dr.-Ing. Alexandra Teynor
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Pattern recognition and machine learning (2 Credit hours) Practical work Pattern recognition and machine learning (2 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying practical course to apply and deepen the acquired knowledge
Prerequisites	Adequate math skills (linear algebra, statistics)
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970344, 2970837 TI 3976548, 2976602 WI 3975752 IIS 9775140
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Basics of pattern recognition
- Preprocessing and feature extraction
- Performance measures
- Simple classifiers (e.g. minimum distance classifiers)
- Probabilistic classifiers
- Unsupervised learning / clustering
- Neural networks
- Deep learning approaches

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- understand relevant basic techniques of pattern recognition
- to select, extract and/or combine suitable features for further processing
- select and apply suitable classifiers for given classification problems
- apply clustering algorithms for meaningful grouping of data
- compare the performance of pattern recognition systems based on recognized performance characteristics

Reading list

R. Duda et al., „Pattern classification”, Wiley, 2000

C. M. Bishop, „Pattern recognition and Machine learning”, Springer, 2006

T. Hastie et al.: „The Elements of Statistical Learning“, Springer 2011

Aurelien Geron: Hands-On Machine Learning with Scikit-Learn & TensorFlow, O'Reilly, 2017

2.45 Network Penetration Testing

Information about the module

engl. Name	Network Penetration Testing
Code	NETP.WP
Coordinator	Dr. Lothar Braun
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Network Penetration Testing (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Lecture, exercise, student research project
Prerequisites	Knowledge of <ul style="list-style-type: none">• IT security• Networks• Linux is an advantage (but not necessary)
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: 1 DIN A4 page handwritten
Examination number	IN 3970358, 2970855 TI 3976602, 2976666 WI 3975773 IIS 9775141
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Planning penetration tests for networks
- Creation of reports
- Information gathering in the network
 - Techniques for detecting machines and services in networks with common tools
 - Investigation of attack surfaces of network services
 - Identification of potential vulnerabilities in network services
- Attacks on network services
 - Password attacks
 - Attacks on web applications
 - Analysis, customization and use of exploits
 - Buffer overflow exploits
 - Development of scripts to carry out attacks

Qualification aims for the module learning objectives/skills

Students acquire knowledge about the implementation of penetration tests in computer networks.

Students learn how to use techniques to obtain information in the network. They know the relevant techniques for identifying vulnerabilities.

Students learn the techniques for carrying out attacks to demonstrate vulnerabilities found and are able to apply these using known tools. They are able to give recommendations for action to eliminate the vulnerabilities.

Reading list

Georgia Weidman: Penetration Testing: A Hands-On Introduction to Hacking, No Starch Press, 2014

Google Hacking for Penetration Testers, Third Edition, Syngress, Dezember 2015

Script

2.46 Neural Networks and Deep Learning

Information about the module

engl. Name	Neural Networks and Deep Learning
Code	NNDL4.WP
Coordinator	Prof. Dr. Michael Kipp
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Neural Networks and Deep Learning (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Lecture with practical components and weekly tasks to apply and deepen the acquired knowledge.
Prerequisites	Basics of programming and mathematics as taught in the first two semesters of computer science courses
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Information Systems, Computer Science, Computer Engineering: Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h Interactive Media: Credit hours: 4, CP credits: 8, Contact hours: 60h, Independent study: 180h, Total workload: 240h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: calculator
Additional information on the type of examination	for Interactive Media Portfolio exam: <ul style="list-style-type: none">• Written examination, 60 minutes, auxiliary: calculator, 70%• Presentation, 10 minutes, 30%

Examination number	IN 3970367, 2970865 TI 3976603, 2976676 WI 3975785 IIS 9775142
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Introduction to machine learning
- Fundamentals of neural networks (feedforward networks)
- Training and evaluation (backpropagation, hyperparameters, optimization)
- Creation, training and evaluation of neural networks in Python (Tensorflow/Keras)
- Convolutional networks using the example of image recognition
- Network architectures
- Recurrent neural networks (GRU and LSTM) using the example of speech processing
- Transformer networks

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Describe the structure and function of neural networks mathematically
- Distinguish between different types and architectures of neural networks and their areas of application
- Pre-process the data for given data sets in an environment such as Jupyter Notebook, select, generate, train and evaluate suitable networks
- Using standard libraries such as TensorFlow, Keras or PyTorch to solve data-based problems with the help of hyperparameter tuning, visualization and systematic evaluation

Reading list

- M. Kipp (2023):** Neuronale Netze und Deep Learning, Onlineskript unter <https://michaelkipp.de/deeplearning>
- F. Chollet (2021):** Deep Learning With Python, 2nd Edition. Manning Publications.
- R. Schwaiger, J. Steinwendner (2019):** Neuronale Netze programmieren mit Python. Rheinwerk Computing.
- M. Ekman (2021):** Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow. Addison-Wesley.

2.47 NoSQL

Information about the module

engl. Name	NoSQL
Code	NoSQL4.WP
Coordinator	Prof. Dr. Michael Predeschly
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	NoSQL (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching and accompanying work placement to apply and deepen the knowledge acquired. In addition, the practical course supports and promotes self-study.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 75%• Presentation, 15 minutes, 25%
Examination number	IN 3970383, 2970881 TI 3976549, 2976697 WI 3975801 IIS 9775143
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The lecture focuses on developments in the field of NoSQL databases. Different types of NoSQL databases and their respective special features are discussed.

In addition to practical work with different NoSQL systems, the underlying theoretical concepts are in the foreground.

Qualification aims for the module learning objectives/skills

Students receive an overview of the subject area of NoSQL databases. After successfully completing the module, students will be able to

- recognize the necessity of NoSQL databases and to assess the usefulness of their use.
- differentiate between various NoSQL databases and classify them according to their respective purpose
- design and install a NoSQL database
- submit queries to a selected NoSQL database in all stages of a CRUD cycle

Reading list

A bibliography will be provided during the course.

2.48 Object-oriented programming with Python

Information about the module

engl. Name	Object-oriented programming with Python
Code	OPPYTH4.WP
Coordinator	Prof. Dr.-Ing. Thomas Kirchmeier
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Object-oriented programming with Python (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, exercise, practical course
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Electronic examination, 120 minutes, all auxiliaries, eigener Laptop, inkl. Chat-GPT
Examination number	IN 3970403, 2970901 TI 3976626, 2976727 WI 3975821 IIS 9775181
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

In general, in order to understand a programming language and to be able to use it independently, experience is necessary. This should be achieved through the numerous programming tasks and control questions (Moodle quizzes). One or more instructional videos are provided for each lecture section (45 minutes on average). These must be worked on independently (do not simply watch them, but write down the most important points, for example) and then the control questions (Moodle quiz) must be answered. Finally, the transfer task is completed. In the instructional videos, an address manager is built up step by step. The programming aspects learned in the process must be transferred to a stock exchange manager. This trains the reading and understanding of foreign code, the transfer to your own task and the interpretation of error messages from the Python interpreter during programming. These three skills are also required when searching for problems on the internet and transferring them. The following aspects are addressed in terms of content:

- Data structures and nested structures
- Object-oriented programming
- Inheritance and composition
- Application of Python libraries
 - os (operating system)
 - sys (system)
 - datetime (date and time)
 - urllib (data acquisition from the Internet)
 - argparse (program arguments)
 - logging (data logger for debugging)
 - time, threading (more efficient program design)
- Application of third-party modules via virtual environment (venv)
 - numpy (math library)
 - matplotlib (plots and visualization)
 - pandas (data management)
 - read, write from text files

Qualification aims for the module learning objectives/skills

Knowledge:

- Programming language Python
- Object-oriented programming
- Data structures and structuring concepts
- Organization and file structure of small software projects
- Application of various Python libraries

Skills:

- Assessing the structuring approach for data
- Reading and interpreting code
- Extension of own and other people's code
- Interpretation of program errors and analysis of the cause of errors

Competencies:

- Reading in raw data and pre-processing it
- Structuring data in terms of the task for targeted further processing
- Processing and evaluating the data by creating functions
- Further structuring of functions and data in objects

Reading list

Online documentation of the Python programming language. URL: <https://docs.python.org>

Eric Matthes: Python Crash Course. 3. Aufl. No Starch Press, Incorporated, 2019.
ISBN: 9781718502703

Johannes Ernesti und Peter Kaiser: Python 3. Rheinwerk Verlag GmbH, 2015. 1126 S. ISBN: 3836291290.

URL: https://www.ebook.de/de/product/44876051/johannes_ernesti_peter_kaiser_python_3.html

Thomas Theis: Einstieg in Python. Ideal für Programmieranfänger geeignet. Galileo Press GmbH, 2014. ISBN: 9783836228619.

Michael Weigend: Python 3 - Lernen und professionell anwenden. mitpVerlag, 2016.
ISBN: 9783826694561.

2.49 Open Source Software Development

Information about the module

engl. Name	Open Source Software Development
Code	OSSWE4.WP
Coordinator	Prof. Dr. Hubert Högl
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Open Source Software Development (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, exercises, practical course
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Remote exam with video supervision, 60 minutes, none auxiliaries
Examination number	IN 3970317, 2970742 TI 3976604, 2976526 WI 3975662 IIS 9775144
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Historical development of free software
- The GNU project
- Open source software
- Production of free software
- Legal aspects of free software
- Important free projects
- The open source principle in other areas

Qualification aims for the module learning objectives/skills

- Understanding the historical development of free software to "open source" software.
- Insights into the typical tools for developing free software.
- Knowledge of collaborative techniques to participate in a free project.
- Ability to start your own free project.
- Overview of free programs from the most important areas.
- Knowledge of open source licenses.

Reading list

Volker Grassmuck, Freie Software zwischen Privat- und Gemeineigentum, Bundeszentrale für politische Bildung.

<http://freie-software.bpb.de>

Karl Fogel, Producing Open Source Software. How to Run a Successful Free Software Project, O'Reilly 2005, 302 pages. Das Buch erschien unter der Creative Commons Attribution-ShareAlike Lizenz und ist somit auch frei erhältlich.

<http://producingoss.com>

Open-Sources, Voices from the Open-Source Revolution, O'Reilly 1999.

<http://oreilly.com/openbook/opensources/book/>

Joseph Feller, Perspectives on Free and Open Source Software, MIT Press, 2005.

<https://mitpress.mit.edu/books/perspectives-free-and-open-source-software>

Material von der Website "Teaching Open-Source"

<http://teachingopensource.org>

Jono Bacon, The Art of Community, O'Reilly, 2nd edition, 2012

<http://www.artofcommunityonline.org>

Greg Wilson, The Architecture of Open-Source Applications

<http://aosabook.org/en/index.html>

Notes: [Homepage of the event:](#)

<http://elk.informatik.hs-augsburg.de/hhweb/oss/index.html>

2.50 Practical Robotics with Matlab

Information about the module

engl. Name	Practical Robotics with Matlab
Code	PRRO.WP
Coordinator	Prof. Dipl.-Ing. Georg Stark
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course in the winter semester and in the summer semester.
Courses	Practical Robotics with Matlab (4 SWS) Practical work Practical Robotics with Matlab (2 SWS)
Teaching language	The module is taught in German and English
Teaching and learning methods	Seminar-style teaching and accompanying practical course with group-related programming exercises. Their close interlinking ensures in-depth learning of the acquired knowledge.
Prerequisites	Basics of mathematics and programming
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN - TI - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Introduction to robotics

- Definitions, practical robotics
- Robot classes and their areas of application

Robot Mathematics I

- Simple geometric elements
- Linear mappings

MATLAB programming techniques I

- Simple methods of robot mathematics
- Introduction to the ROBOMATS function library

Modeling and implementation of simple kinematic models

- Forward transformation
- Back transformation

Introduction to the application programming of a modern industrial robot controller

Future development

Practical course

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Describe the individual types of robots and their areas of application,
- Develop simple MATLAB programs,
- Implement kinematic robot models with the help of MATLAB,
- Develop simple robot application programs,
- Explain the industry requirements for modern robot controllers
- Transfer and apply the methods of practical robotics to general mechatronic systems

Reading list

Verwendete Literatur

Stark G.: Robotik mit Matlab. Hanser, 2009.

http://www.hs-augsburg.de/stark/robotik_mit_matlab/

This book should be obtained, as the lecture is largely based on it.

Further reading

Introduction to robotics, applications

Craig, J. J.: Introduction to Robotics. Pearson Education, 2005.

Haun, M.: Handbuch Robotik. Programmieren und Einsatz intelligenter Roboter. Springer, 2007.

Hesse, S.: Grundlagen der Handhabungstechnik. Hanser, 2006.

Fundamentals of robot mathematics

Hoffmann, A.; Marx, B.; Vogt, W.: Mathematik für Ingenieure. Pearson Education, 2005.

Papula, L.: Mathematik für Ingenieure und Naturwissenschaftler Bd. 1/2. Vieweg, 2001

Papula, L.: Mathematische Formelsammlung für Ingenieure und Naturwissenschaftler. Vieweg, 2006.

Programming with MATLAB, error handling and optimization

Beucher, O.: Matlab und Simulink. Grundlegende Einführung für Studenten und Ingenieure in der Praxis. Pearson Education, 2006.

Schweizer, W.: MATLAB kompakt. Oldenbourg, 2006.

Stein, U.: Einstieg in das Programmieren mit Matlab. Hanser, 2007.

Kinematic structure, path control

Corke, P.: Robotics, Vision and Control. Springer, 2017.

Siegert, H.-J.; Boncione, S.: Programmierung intelligenter Roboter. Springer 1996.

Vidyasagar, M.; Spong, M.W.; Hutchinson, S.: Robot Modeling and Control. John Wiley & Sons, 2006.

Weber, W.: Industrieroboter. Methoden der Steuerung und Regelung. Hanser, 2002.

2.51 Process Intelligence

Information about the module

engl. Name	Process Intelligence
Code	PRCINT4.WP
Coordinator	Prof. Dr. Wolfgang Kratsch
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Process Intelligence (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching and accompanying internship in which case studies are worked on in small groups over the semester
Prerequisites	Basic programming knowledge is an advantage
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 10-25 Seiten, 60%• Presentation, 20-30 minutes, 40%
Additional information on the type of examination	
Examination number	IN 3970398, 2970896 TI 3976551, 2976718 WI 3975816 IIS 9775176
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The module „Process Intelligence“ teaches students basic concepts and advanced techniques in the field of data-driven process management. The students learn how to analyze, optimize and automate business processes using technologies such as process mining, predictive process monitoring, context-aware process mining and robotic process automation.

Qualification aims for the module learning objectives/skills

After successfully completing this module, students will be able to

- Classify technologies from the field of process intelligence to optimize processes and apply them within a limited framework
- Identify process weaknesses using process mining and identify potential for improvement based on evidence
- Develop prediction models for process sequences using machine learning
- Automate simple processes using RPA software
- Develop Python-based process intelligence solutions yourself using standard libraries such as PM4Py, SKlearn or Keras

Reading list

Van Der Aalst, W. (2016): Process Mining. *Data science in action*. Springer Berlin Heidelberg.

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018): *Fundamentals of business process management* (Vol. 2). Heidelberg: Springer.

2.52 Programming using Databases

Information about the module

engl. Name	Programming using Databases
Code	DBP4.WP
Coordinator	Prof. Matthias Kolonko, Ph.D. (ONPU)
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Programming using Databases (4 Credit hours)
Modul area	Applications
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, practical exercises
Prerequisites	Database Management Systems, Programming 1+2 The lecture Database Applications is recommended.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970384, 2970882 TI 3976552, 2976698 WI 3975802 IIS 9775147
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The lecture deals with the possibilities of connecting relational databases to the business logic, which can be created in different programming languages.

The following basic access options are examined in more detail:

- Direct access via embedded SQL
- Access via an individual API
- Access via existing frameworks such as object-relational mapping (ORM) or Data Transfer Objects (DTO)

The basic possibilities and concepts are highlighted with a focus on the highlighted using the Java programming language. In addition other current programming languages are also examined, demonstrated and compared. (PHP, Python, C/C++, ...)

The participants should also be shown the correct structure within the software architecture by discussing the advantages and disadvantages. Here also security aspects are also taken into account.

Concepts for the use of "Polyglot Persistence" are also presented as part of the lecture, to demonstrate the possibilities of diversifying data storage.

The content discussed will be practiced by the students themselves as part of an accompanying and practiced by the students themselves.

Qualification aims for the module learning objectives/skills

After completing the module, participants will be able to

- distinguish and describe the options for connecting databases.
- use the various database connection options.
- analyze the requirements for the database connection.
- implement a suitable database connection.
- recognize the possibilities of Polyglot Persistence.

Reading list

Literature recommendations will be provided in the lecture.

2.53 Programming with Python

Information about the module

engl. Name	Programming with Python
Code	PROGPY6.WP
Coordinator	Prof. Dr. Peter Rösch
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Programming with Python (4 SWS) Practical work Programming with Python (2 SWS)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, practical course, partly with teamwork.
Prerequisites	<ul style="list-style-type: none">• Object-oriented software development.• Fundamentals of vector calculus and analysis.
Usage possibilities	WPF only for Bachelor programs: Information Systems, Computer Engineering and Interactive Media. For Computer Science (Bachelor) this is a compulsory subject (Programming 3).
Total workload and its constituent parts	Credit hours: 6, CP credits: 8, Contact hours: 90h, Independent study: 150h, Total workload: 240h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	TI 3976553, 2976599 WI 3975746 IIS 9775148
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Summary

Customers expect high-performance, easy-to-use solutions for increasingly complex tasks, with the acceleration of market cycles leading to enormous time and success pressure for many software projects.

Modern software engineering concepts promise a remedy, but only lead to success if basic software construction methods that have been known for decades are used. The problem to be solved must first be understood and systematically analyzed before alternative solution approaches can be developed and their feasibility demonstrated using prototypes if necessary. As a rule, one approach is pursued further, which ultimately leads to the final product.

An important task in this process is the selection of suitable programming languages, whereby different languages can be used in the individual phases of the project. In order for a developer or project manager to choose the "right" language for a sub-problem, they should have gained experience with several languages and have an overview of their advantages and disadvantages.

The course introduces Python as a representative of the object-oriented scripting languages. The syntax of this language is so simple and the extension libraries are so powerful that developers can concentrate on the task at hand when implementing advanced concepts without being distracted by inconsistencies or tricky language constructs.

As part of the exercises, tasks from the areas of mathematics and simulation are dealt with in order to develop and practically apply central techniques of software construction.

Efficient software development with Python

- Python – Introduction
- Interactive software development with Jupyter Notebooks
- Automation of tests
- Systematic optimization
- Graphical user interfaces

Advanced programming techniques with Python

- Concurrency
- Design patterns
- Integration of heterogeneous components
- Scientific applications
- Distributed applications

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Describe the syntactic constructs of the programming language Python programming language.
- Classify given source code in terms of efficiency and quality. classify.
- Optimize applications in terms of the consumption of computing time and memory consumption.
- Compare the implementations of algorithms of medium complexity in different programming languages.
- Solve tasks by skillfully combining existing components. of existing components.
- Break down problems of medium complexity into sub-problems.
- Develop, test and implement software components for solving problems of medium complexity yourself, test and document them.

Reading list

Johannes Ernesti, Peter Kaiser: Python3 – Das umfassende Handbuch, 5. Auflage, Rheinwerk Computing (2017)

Bernd Klein: Einführung in Python 3, Hanser (2014)

Mark Pilgrim: Python 3 – Intensivkurs, Springer (2010)

Dusty Phillips: Python 3 Object-Oriented Programming, 3. Auflage, Packt Publishing (2018)

Eric Freeman, Elisabeth Freeman: Entwurfsmuster von Kopf bis Fuß, O'Reilly (2015)

Mark Summerfield: Rapid GUI Programming with Python and Qt - The definitive Guide to PyQt Programming, Prentice Hall (2015)

Python-Homepage: <https://www.python.org/>

2.54 Programming of Web Applications

Information about the module

engl. Name	Programming of Web Applications
Code	PWA4.WP
Coordinator	Prof. Dr. Anja Metzner
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Programming of Web Applications (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-based teaching, project work, presentations (Note: If the number of participants is low, the lecture can also be held in directed reading format with reduced attendance time).
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Project work, 75%• Written assignment, 5-15 pages, 25%
Examination number	IN 3970381, 2970879 TI 3976605, 2976690 WI 3975799 IIS 9775149
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Web-based systems have become an important economic factor. This lecture provides an overview of the programming of web applications with current frameworks.

Basic topics:

- Architectures of web-based systems (including MPA, SPA, progressive web apps)
- Markup and scripting languages of the web (e.g. HTML, CSS; JavaScript)
- Overview of current frameworks and libraries (frontend, backend)
- Programming web applications (with relevant development tools, IDEs, build tools, validation, debugging and testing)

Exemplary selection of topics for students to specialize in:

- Database connection options
- Advanced web technologies (e.g. AJAX: asynchronous data transfer, REST, Web-sockets, GraphQL, cross-platform development of mobile apps)
- Web design and UX/UI (e.g. responsive design, CSS frameworks, principles, accessibility)
- Security topics (e.g. HTTPS, authentication, security vulnerabilities)

Introduction:

With the help of short lectures by professors on web architecture, scripting languages, current frameworks, a relevant collection of materials and project support, students are enabled to program the web and to continue studying on their own.

Choice of scripting languages and frameworks:

Students will be able to understand and program the scripting languages of the web, in particular HTML, JavaScript, CSS **and at least one framework** of their choice. **The choice of possible scripting languages and frameworks depends on the student's degree program. Excluded are those that can already be studied in the curriculum or in other approved elective subjects of the respective degree program.**

Project work and presentation:

With the help of project experiments, students learn how to program their own web projects. The techniques used are presented to all participants in student project presentations, so that a broad insight into many current frameworks and libraries is gained. Finally, each presentation is documented as a student research project and (voluntarily) made available to all participants.

Qualification aims for the module learning objectives/skills

Students are then able to read and understand the programming of web applications and create web projects themselves. This gives them the basics to work as a full-stack programmer.

1 Understanding the basics of web programming

- Students can explain the basic concepts and technologies of the web (HTML, CSS, JavaScript, HTTP/HTTPS).
- Students know the architecture of the internet and can describe how web servers and clients work.
- Students can explain the significance of web standards (W3C) and know why they are important.
- Students are familiar with current best practices for web development, such as responsiveness and accessibility.

2 Applying web programming

- Students can create structured and semantically correct HTML documents.
- Students can use CSS to create and customize the layout and design of web pages.
- Students can write basic JavaScript programs that enable interactivity on websites.
- Students understand the DOM (Document Object Model) and can use JavaScript to manipulate it.
- Students are able to install and use a technology stack of their choice for programming.
- Students are able to use at least one current framework and integrate JavaScript libraries if required.
- Students learn how to use different browsers for programming.
- Students are able to systematically identify and rectify errors in web applications.
- Students learn about specialization topics such as database connection, JSON or AJAX.

3 Teamwork and project management

- Students can work together in teams to plan and implement complex web projects.
- Students are familiar with agile project management methods and can apply these in web development (e.g. Scrum, Kanban).

4 Documentation and presentation

- Students can document their projects and the development process in writing and present them in presentations.
- Students are able to communicate technical problems and solutions clearly and comprehensibly.

5 Technology transfer

- Students are able to transfer their knowledge to new technologies and frameworks by applying basic principles of web development.

Reading list

Philip Ackermann: Webentwicklung: Das Handbuch für Fullstack-Entwickler in neuer Auflage, Rheinwerk Computing, 2023.

David Flanagan, Jens Olaf Koch , et al.: JavaScript - Das Handbuch für die Praxis, O'Reilly, 2021.

Cybellium Ltd, Kris Hermans: Mastering Back-End Development, Independently published, 2023.

Kiet Huynh: Front-End Web Development: Techniques and Trends, Independently published, 2023.

Sebastian Springer: Node.js: The Comprehensive Guide, Rheinwerk Computing, 2022.

2.55 Project Jupyter

Information about the module

engl. Name	Project Jupyter
Code	PRJU4.WP
Coordinator	Prof. Dr. Nik Klever
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Project Jupyter (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	<p>The event is divided into four parts:</p> <ul style="list-style-type: none">• Part 1 - Introduction to the applications of Project Jupyter and exercises (1st block 2 days)• Part 2 - Brainstorming and brainstorming of topics for student research projects from e.g. the following areas (2nd block 2 days)• Part 3 - Implementation of the student research project topics (online approx. 11 weeks)• Part 4 - Presentation of the student research projects (3rd block 1 day)
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Exercises, 20% • Description Brainstorming and idea generation, 10% • Written assignment, 10-40 pages, 70%
Examination number	IN 3970374, 2970872 TI 2976683 WI 3975792 IIS 9775150
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Project Jupyter now comprises a number of applications. It emerged from the IPython Notebook, which subsequently became the Jupyter Notebook, which is now widely used for data science and AI applications in particular, Jupyter Notebook, which is now widely used for data science and AI applications in particular.

Jupyter Notebook

In recent years, Jupyter Notebook has become increasingly popular not only for computer scientists but also for natural scientists, economists and engineers. Why is that? One of the reasons is that Jupyter Notebook makes it easy to combine a wide variety of materials such as normal text, images, graphics with HTML, LaTeX, SVG graphics and especially this with programming code of different programming languages such as Python, Java, JavaScript, C++, R, Scala, and others. One particular advantage is that the user interface of a Jupyter Notebook Server for creating a Jupyter Notebook is solely a browser.

JupyterLab

The further development of the Jupyter Notebook is the JupyterLab, which is an extended web-based interactive development environment for Jupyter Notebooks, program code or data. JupyterLab is more flexible than Jupyter Notebook, as the user interface can be configured and arranged by the user. This means that a large number processes in the fields of data science, scientific computing and machine learning can be supported. JupyterLab is also expandable and modular via plugins and components.

JupyterHub

Jupyter Notebook and JupyterLab are single-user web servers that are easy to install and run on any computer. The extension of these single-user web servers for companies, organizations, universities, work teams, etc. to a multi-user web server is done by the JupyterHub Server. There are also corresponding extensions for the JupyterHub Server, such as nbgrader, an automated distribution and code checking framework based on Jupyter Notebook and JupyterHub.

Viola

The latest addition to Project Jupyter is Voilà, an application that converts Jupyter Notebooks into a standalone web application in such a way that only the program code from the Jupyter Notebook that has been approved by the Jupyter Notebook owner is visible and usable for the users. This sharing will be controlled via a secure and customizable interactive dashboard.

Qualification aims for the module learning objectives/skills

The students can classify, understand, configure and use the individual applications from Project Jupyter. Furthermore, they should be able to improve or even further develop individual applications in the form of plugins or patches.

Reading list

Further information at

<https://klever.hs-augsburg.de/nb/OWL/>

2.56 Project - Research and Transfer

Information about the module

engl. Name	Project - Research and Transfer
Code	FUT.WP
Coordinator	Prof. Dr. Alexander von Bodisco
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	ein Semester, jeweils im Winter- und Sommersemester
Courses	Project - Research and Transfer (4 Credit hours)
Teaching language	The module is taught in German and English.
Teaching and learning methods	Students work in small groups to develop IT solutions to a current research topic. The aim is to get to know application-oriented research, as well as the transfer and the associated problems in a realistic way. The projects undertaken have a clear practical relevance and are typically carried out within the framework of funding/third-party projects or in cooperation with companies. The project topics are assigned by authorized examiners from the Faculty of Computer Science and include a practical part (software/hardware), documentation (student research project) and a presentation. The practical part (software and possibly hardware) must be described as part of the student research project. The presentation usually takes place as part of a project day or a seminar. Coordination with the project creator takes place in regular personal meetings and via electronic channels. The work is not necessarily tied to the lecture period.
Prerequisites	Solid knowledge of the most important areas of computer science, such as algorithms and data structures, programming, database systems, data communication, software engineering and, if applicable, operating systems. The acquired knowledge should already have been practically applied in a team project.
Usage possibilities	Required elective module for the Bachelor's degree programs in Computer Science, International Information Systems, Computer Engineering, Information Systems

Total workload and its constituent parts	Credit hours: 8, CP credits: 10, Contact hours: 120h, Independent study: 180h, Total workload: 300h
--	--

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Project work, 10-20 Seiten, 80% • Presentation, 10-20 minutes, 20%
Examination number	IN 3970404, 2970902 TI 3976627, 2976728 WI 3975822 IIS 9775182
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The research and transfer project offers students the opportunity to apply theoretical knowledge in practice while developing innovative solutions to real-world challenges. The project focuses on research, teamwork and the transfer of results into practice. The students' tasks include project management, software development, independent familiarization with research topics, the preparation of research results and their presentation with regard to practical application.

In moodle you will find the current topics that are currently on offer:
<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Plan and execute software tasks in a team in terms of time, effort and resources.
- Apply software development methods in practice.
- Learn new software techniques and apply suitable methods.
- Develop research topics independently.
- Document project results in an understandable and appealing way.

Reading list

Is determined individually for each project and is based on current scientific research in the selected field.

2.57 RFID and NFC technology

Information about the module

engl. Name	RFID and NFC technology
Code	RFID3.WP
Coordinator	Prof. Dr. Volodymyr Brovko
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is offered regularly in both the winter and summer semesters.
Courses	RFID and NFC technology (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, exercises, practical course
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 3, CP credits: 5, Contact hours: 45h, Independent study: 105h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, auxiliary: 1 DIN A4 page handwritten
Examination number	IN 2970806 TI 2976589 WI ...
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

- Basics of RFID technology. Coding and modulation in RFID. Anti-collision in RFID.
- Memory card architecture. Smart card architecture. Java Cards.
- Autonomous RFID and NFC systems: technical basics. Software design.
- NDEF on memory card and MIFARE.
- Architecture of mobile NFC devices.
- NFC on ADROID System

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Know the technical basics of RFID technology.
- Understand memory and smart card architecture.
- Know the architecture of an autonomous RFID and NFC system.
- Program a simple NFC read/write device based on a microcontroller.

Reading list

Josef Langer, Michael Roland: Anwendungen und Technik Von Near Field Communication (NFC), Springer-Verlag, 2010 - 265 Seiten

Klaus Finkenzeller: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC, Carl Hansen Verlag München, 2012

Wolfgang Rankl, Wolfgang Effing: Handbuch der Chipkarten, Carl Hansen Verlag München, 2008

2.58 Swabia Innovation Masterclass

Information about the module

engl. Name	Swabia Innovation Masterclass
Code	SIM8.WP
Coordinator	Prof. Dr. Christoph Buck (THA) Prof. Dr. Bayer/ Prof. Dr. Daniel Schallmo (HNU) Prof Dr. Erik Lehmann (UniA) Prof. Dr. Bernd Lüdemann-Ravit (HKE)
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The duration of the module is two semesters. The Swabian Innovation Masterclass is offered regularly starting in the winter semester.
Courses	Schwaben Innovation Masterclass (10 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, workshop units, best practices, team/group work, presentations, maximum number of participants 6
Prerequisites	Both modules must be taken, it is not possible to attend only one Masterclass. A total of 10 ECTS can be acquired. Travel and accommodation costs for the journey to Neu-Ulm, Kempten and Bergamo will be covered by the joint project.
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Per semester: SWS: 4, CPs: 5, total 8 SWS/ 10 CPs Attendance time: 30 h, self-study (preparation/ follow-up): 120 h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Partial examination 1: Presentation, 5-10 minutes (70%) and written elaboration, 3-8 pages (30%), 25%% • Partial examination 2: Presentation, 5-10 minutes (70%) and written elaboration, 3-8 pages (30%), 25%% • Partial examination 3: Presentation, 5-25 minutes (70%) and final report, 10-14 pages (30%) , 25%% • Partial examination 4: Presentation, 15-25 minutes (70%) and pitch slides, 7-10 slides (30%) , 25%%
Examination number	
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

This course consists of four chapters: future (&) trends, social impact, business ideation and international entrepreneurship. Chapter 1 is offered at Augsburg University of Applied Sciences, Chapters 2, 3 and 4 as external courses at the universities of Kempten and Neu-Ulm and the University of Augsburg, including a stay abroad in Bergamo (Italy).

Chapter 1: Future (&) Trends (THA)

Chapter 1 is divided into two phases: trend and scenario analysis. As part of the trend analysis, students examine current developments in the field of digital technologies. They record the status quo and identify emerging trends. An interdisciplinary approach enables students to look at the topics from different perspectives and to analyze economic, social, political, legal and ecological framework conditions. In the second phase, scenario analysis, students build on the results of the trend analysis and develop their own trend analysis and develop their own ideas for innovative products or services. They deal with the possible effects of the identified trends and develop scenarios for future developments.

Teaching methods include lectures and workshops led by external and internal lecturers from academia and industry. In addition, students receive regular coaching and feedback on their work. In addition, the students have the opportunity to discuss and further develop their results and ideas in plenary sessions as part of a joint interim and final presentation.

Chapter 2: Social Impact (HKE)

In Chapter 2, students are introduced to the diversity of social challenges at global, regional and local level. They learn the concepts of impact thinking, including scalability, sustainability and impact measurement in social innovation. They gain practical insights by analyzing case studies of successful projects. In the "Technology for Social Impact" area, digital technologies such as blockchain, artificial intelligence and the Internet of Things are discussed, as well as their potential applications for tackling social challenges. The project development and management phase enables students to put their knowledge into practice by developing technology-based solutions in interdisciplinary teams, accompanied by mentors and experts. The final challenge involves presenting the results to an interdisciplinary jury and reflecting on the process.

Chapter 3: Business Ideation (HNU)

The Masterclass offers students a practical approach to entrepreneurial challenges, especially in the context of innovation and start-up initiatives. The students develop a sound understanding of the importance of innovation. They acquire established methods and practical tools from various fields, including creativity techniques, design thinking and lean startup, to generate customer and demand-oriented innovations. At the same time, they are given the opportunity to develop their own innovation projects in collaboration with regional companies in order to gain a realistic insight into entrepreneurial processes.

Chapter 4: International Entrepreneurship (UA)

The main aim of the module is for students to follow on from Chapter 3 in a course last-

ing several days to deepen their knowledge of the international aspects of entrepreneurship and innovation. They will learn international skills as part of a summer school. This starts at the University of Augsburg and ends with an excursion to the University of Bergamo in Italy. The focus is on learning experiences with other cultures. A special focus of the chapter is the guest lectures by internationally renowned guest lecturers from Indiana University Bloomington, USA. They are intended to impart knowledge, experiences and views from a scientific perspective in english language. Practical elements are also taught, with managers from companies in Augsburg region who provide insights and share their knowledge.

Qualification aims for the module learning objectives/skills

After successful participation in **chapters 1 and 2**:

- Students will be able to understand the challenges of collaboration in interdisciplinary project teams.
- They are also able to apply trend and future research methods in a project team and jointly write a trend report on current developments in the field of digital technologies.
- Students can carry out a status quo analysis, identify trends and future developments and form ideas for future products or services.
- The module aims to provide students with practical skills in the application of research methods, in scientific writing, in the discussion and presentation of topics and in interdisciplinary collaboration.
- Through the presentation and discussion of case studies of successful projects, they gain insights into best practices. Furthermore, they will be familiarized with various digital technologies to tackle social problems and have the opportunity to deepen these in workshops and guest lectures.
- Through project development and management in interdisciplinary teams, they develop technology-based solutions to specific social problems and present their results to a jury.
- Their learning process is supported and deepened through reflection and documentation.

Chapter 3: Business Ideation

Subject-related skills:

- Students are enabled to apply concepts such as Lean Startup and Design Thinking to manage and successfully implement innovation processes.
- They gain an in-depth understanding of business planning and learn to use tools effectively to design agile and successful start-up processes.
- The focus is also on identifying demand potential, target groups and competitive advantages.

Methodical competencies:

- Students learn methods for brainstorming and developing business models as well as for market analysis in order to make well-founded decisions in the corporate context.

- A special focus is placed on the application of design thinking, a creative problem-solving method that aims to develop user-centered solutions.
- In addition, you will be introduced to the Design Sprint Lean Startup methodology to quickly and efficiently conceptualize, prototype and test solutions.

Social skills

- Students develop the ability to make effective decisions in a team and learn to understand innovative solutions and present them convincingly to company representatives.
- The exchange with fellow students and cooperation in interdisciplinary teams promote the understanding of different perspectives and the development of solution approaches.

Personal skills

- Students are encouraged to reflect on the consequences of their decisions and develop their personal skills in entrepreneurship and risk assessment. Through practical work on innovation projects, they strengthen their problem-solving skills and gain self-confidence in their abilities.

Chapter 4: International Entrepreneurship

By participating in this course, students should gain an in-depth understanding of international entrepreneurship for the development of innovations. After successfully completing this module, students will know the basics of entrepreneurship and will be able to

Subject-related competencies

- understand international entrepreneurship,
- internationalize their own start-up ideas and business models,
- take an international and intercultural perspective on key social, ecological and economic changes in society, the environment and the economy
- project innovative ideas onto the international market

Methodical skills

- handle complex start-up projects in a goal-oriented manner,
- carry out systematic needs and action analyses in a social context from different perspectives

Interdisciplinary competencies

- apply multi-perspective thinking,

- recognize and promote opportunities for social, economic and ecological improvements from different perspectives,
- implement innovative solutions in the context of international entrepreneurship.

Key competencies

- reflect on strategies for setting up a business,
- develop and justify strategic considerations independently,
- think and work in an interdisciplinary way,
- solution-oriented and intercultural communication.

Reading list

Chapter 1: Will be announced at the beginning of the semester

Chapter 2:

- Chang, Ann Mei: Lean Impact, How to Innovate for Radically Greater Social Good, San Francisco, 2018.
- Impact Measurement – Wirkung- und Wirkungsmessung Sozialer Innovationen.
- Kursbuch Wirkung, Praxishandbuch für alle, die Gutes noch besser tun wollen. Social Reporting Standard, Leitfaden zur wirkungsorientierten Berichterstattung
- u.v.m

Chapter 3 und 4:

Audretsch, David: Everything in Its Place: Entrepreneurship and the Strategic Management of Cities, Regions, and States. New York: Oxford University Press (2015).

Audretsch, David; Lehmann, Erik: The seven secrets of Germany. Economic Resilience in an Era of Global Turbulence. New York: Oxford University Press (2016).

Schallmo, D.: Design Thinking erfolgreich anwenden, Springer Verlag, Wiesbaden (2017).

Pijl, P. v. d., Lokitz, J., Solomon, L., Pluijm, E. v. d., Lieshout, M. v., Schallmo, D.: Design a better business: Neue Werkzeuge, Fähigkeiten und Mindsets für Strategie und Innovation, Vahlen Verlag, München (2018).

Schallmo, D.: Jetzt Design Thinking anwenden, Springer Verlag, Wiesbaden (2018).

Schallmo, D.: Jetzt digital transformieren. So gelingt die erfolgreiche Digitale Transformation Ihres Geschäftsmodells, Springer Verlag, Wiesbaden (2016)

Brown, T.: Change by Design. Harper Business (2009).

Curedale, R.: Design Thinking. Design Community College (2013).

d.school: Bootcamp Bootleg. Hasso Plattner Institute, Stanford (2010).

Liedtka, J. & Ogilvie, T.: Designing for Growth. Columbia Business School (2011).

Plattner, H.; Meinel, Ch. & Weinberg, U.: Design Thinking. Innovation lernen, Ideenwelten öffnen. München (2009).

Stickdorn, M. & Schneider, J.: This is service design thinking. BIS publishers (2014).

2.59 Service Learning Project

Information about the module

engl. Name	Service Learning Project
Code	SLP.WP
Coordinator	Prof. Dr. Alexander von Bodisco
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	ein Semester, jeweils im Winter- und Sommersemester
Courses	Service Learning Project (4 Credit hours)
Teaching language	The module is taught in German and English.
Teaching and learning methods	Students work in small groups to develop individual IT solutions in the field of service learning for a real civil society partner. In addition to the classic project skills, the aim is to train communication with the project partner and to align a project with a specific service. The project topics are assigned by authorized examiners from the Faculty of Computer Science and include a practical part (software/hardware), documentation (student research project) and a presentation. The practical part (software and, if applicable, hardware) must be described as part of the student research project. The presentation usually takes place as part of a project day, a seminar or a demonstration at the project partner's premises. Coordination with the topic creator/project partner takes place in regular personal meetings and via electronic channels. The work is not necessarily tied to the lecture period.
Prerequisites	Solid knowledge of the most important areas of computer science, such as algorithms and data structures, programming, database systems, data communication, software engineering and, if applicable, operating systems. The acquired knowledge should already have been practically applied in a team project.
Usage possibilities	Required elective module for the Bachelor's degree programs in Computer Science, International Information Systems, Computer Engineering, Information Systems

Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h
--	---

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • Project work, 10-20 Seiten, 80% • Presentation, 10-20 minutes, 20%
---	---

Examination number	IN 3970405, 2970903 TI 3976628, 2976729 WI 3975823 IIS 9775183
--------------------	---

Grading	According to § 20 of the APO in the currently valid version.
---------	--

Content of the module

Students work in groups to carry out small IT projects in the field of service learning largely independently. The students' tasks include project management, software development, independent familiarization with interdisciplinary topics and project orientation with regard to the individual requirements of the respective target group.

In moodle you will find the current topics that are currently on offer:
<https://moodle.hs-augsburg.de/course/view.php?id=7942>

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Align projects with the requirements of real civil society partners.
- Plan and carry out project tasks in a team in terms of time, effort and resources.
- Apply software development methods in practice.
- Prepare interdisciplinary topics in self-study.
- Develop questions and solutions in dialog with project partners.

Reading list

Project-specific literature will be announced by the supervisor before the start of the project.

2.60 Software engineering for data-intensive systems

Information about the module

engl. Name	Software engineering for data-intensive systems
Code	SDS4.WP
Coordinator	Dr. Jan-Philipp Steghöfer
Faculty	Faculty of Liberal Arts and Sciences
Type	Required elective module
Duration / Frequency	one semester, winter semester The registration takes place by e-mail directly with Mr. Steghöfer: jan-philipp.steghoefer@tha.de
Courses	Software engineering for data-intensive systems (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	
Prerequisites	
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	<ul style="list-style-type: none">• Experience report (to be handed in the group)• Individual interviews• Peer assessment• Source code analysis
Examination number	IN -, - TI - WI - IIS -
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The data and models that data scientists work with are always used in the context of an application. In particular, it is necessary to embed data and the machine learning models based on it embedded in an application. This application must meet the requirements of the user and have an appealing and user-friendly interface. It must also be easy to roll out, update and operate.

Using a specific application that students develop in collaboration with external stakeholders, this course teaches the basics of application development, in particular the basics of software architecture and Scrum as an agile software development process. There is also a focus on operating software in containers with the help of Docker and Kubernetes. Computer science and data science students work together in a team, supporting each other with their respective specialist knowledge and interacting with external partners. Collaboration with external partners also requires the collection of requirements and regular feedback loops.

Qualification aims for the module learning objectives/skills

Upon successful completion of the course, students will be able to

- describe the context of working with data and models and the importance of embedding them in an application,
- understand agile software development processes, in particular Scrum, and demonstrate how these can be used in the development of data science applications,
- apply their programming skills to develop a data science application that incorporates data and machine learning models,
- work effectively with external partners to gather requirements, obtain regular feedback and ensure the successful delivery of the application,
- demonstrate an understanding of the principles of software architecture and its role in developing scalable and maintainable data science applications, and
- demonstrate their ability to package applications in Docker containers and deploy, maintain and operate them using Kubernetes.

Reading list

Literature recommendations will be provided in the lecture.

2.61 Software Project Management

Information about the module

engl. Name	Software Project Management
Code	SWPJMG.WP, SWPMG4.WP
Coordinator	Dipl.-Wirt.-Inf. (FH) Andrea Obermeyer, MBA
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, winter semester
Courses	Software Project Management (4 Credit hours)
Teaching language	The module is taught in German.
Teaching and learning methods	Seminar-style teaching, exercise groups, presentation of special content by Master's students
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Written examination, 60 minutes, none auxiliaries
Examination number	IN 3970330, 2970802 TI 3976555, 2976576 WI 3975722 IIS 9775154
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

The course combines theoretical content with practical exercise components, student teaching elements and detailed case studies from software project management. The following key areas are covered:

- Introduction to project management: tasks, interfaces, project phases and project organization
- Process models, software lifecycles and development methods (agile vs. conventional)
- Project types
- Project planning: feasibility studies, requirements engineering
- Effort estimation
- Project monitoring/controlling
- Leadership: corporate culture, leadership, team building
- Soft and social skills for project teams and employees
- Risk management
- Case studies on selected example projects

Qualification aims for the module learning objectives/skills

After successfully completing the Software Project Management module, students can:

- Understand what modern software project management is
- Understand how small and large, technical and business-oriented software projects are organized and led to success or failure
- Select and apply methods, techniques and tools for project management
- Understand team dynamics
- Understand which soft and social skills should be developed for this purpose

Reading list

Literature recommendations will be provided in the lecture.

2.62 Startitup - Entrepreneurial Thinking and Business Design

Information about the module

engl. Name	Startitup - Entrepreneurial Thinking and Business Design
Code	START4.WP
Coordinator	Prof. Dr. Christoph Buck
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	1 semester, summer semester
Courses	Startitup - Entrepreneurial Thinking and Business Design (2 Credit hours) Praktical work Startitup - Entrepreneurial Thinking and Business Design (2 SWS)
Teaching language	The module is taught in German and English.
Teaching and learning methods	Seminar-style teaching and accompanying internship to apply and deepen the acquired knowledge. In addition, the internship supports and promotes self-study.
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none"> • 3 presentations, 20-30 minutes each, 75% • Written assignment, 8-12 pages, 25%
Additional information on the type of examination	The presentations are group presentations. Written assignment: a business project developed during the course is to be prepared as a business plan in group work.

Examination number	IN 3970395, 2970893 TI 3976611, 2976715 WI 3975813 IIS 9775172
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Entrepreneurship is one of the most vibrant disciplines today and can be learned. In this course:

- Students develop their own business idea and think it through from A-Z.
- Students learn a systematic and structured approach to innovation and entrepreneurship (structuring value creation, potential analyses, rapid prototyping, etc.)
- Students apply numerous innovation methods and innovation tools (value proposition canvas, business model canvas, UX design, etc.)
- Presentation skills are actively promoted by having to present progress on a regular basis

Qualification aims for the module learning objectives/skills

After successfully completing the module, students will be able to

- Think through an innovative business idea (profit-oriented or non-profit-oriented) from A to Z (BYO - bring your own, DYO - develop your own)
- Apply innovation methods independently
- Recognize and evaluate business potentials
- Develop innovation approaches in a structured manner

Reading list

Aulet, Bill (2013): Disciplined entrepreneurship: 24 steps to a successful startup. John Wiley & Sons.

Nambisan, Satish, et al. (2017): "Digital innovation management." MIS quarterly 41.1. 223-238.

Osterwalder, Alexander; and Pigneur, Yves (2010): Business model generation: a handbook for visionaries, game changers, and challengers. Vol. 1. John Wiley & Sons.

Osterwalder, Alexander (2015): Value proposition design: How to create products and services customers want. John Wiley & Sons.

2.63 Visual Thinking for Business

Information about the module

engl. Name	Visual Thinking for Business
Code	VISTH.WP
Coordinator	Philip McClenaghan
Faculty	Faculty of Computer Science
Type	Required elective module
Duration / Frequency	The module is regularly offered as a block course during the semester break. (February/March) and (August/September)
Courses	Visual Thinking for Business (4 credit hours)
Teaching language	The module is taught in English.
Teaching and learning methods	Seminar format, practical classes and workshops
Prerequisites	None
Usage possibilities	Required elective for bachelor's degree programs
Total workload and its constituent parts	Credit hours: 4, CP credits: 5, Contact hours: 60h, Independent study: 90h, Total workload: 150h

Exam

Type of exam / required course achievements	Portfolio exam: <ul style="list-style-type: none">• Presentation, 10-25 minutes, 40%• Written assignment, 10-15 pages, 60%
Examination number	IN 3970353, 2970849 TI 3976558, 2976659 WI 3975767 IIS 9775160
Grading	According to § 20 of the APO in the currently valid version.

Content of the module

Companies in the modern business world are turning to new ways of working such as Design Thinking and Lean Start-Up to keep pace with constantly evolving marketplaces and technological advancements. The visual tools and methods of Visual Thinking support these new working practices by making information, ideas, concepts and processes visible and thus accessible to all.

Visual Thinking extends the verbal and written language using visualization methods that enable the graphic representation of ideas and complex content. In the new world of design thinking, agile innovation, lean start-up, etc., this is essential.

This course is suitable for all students who want to think through new ideas, complex content and procedures in a structured manner and communicate effectively in their professional life.

Qualification aims for the module learning objectives/skills

The students should develop the following skills during the course:

- Basic understanding of the theoretical aspects of visual thinking and visual communication.
- Application and further development of visual storytelling methods.
- The ability to communicate ideas and complex content visually.
- The ability to independently use visual thinking in a business environment.

Reading list

Will be announced in the first lecture.

Index

- 3D printing , 8
- Advanced Security Testing , 16
- Agile development of a Click-Dummy Game , 20
- Agile Software Development (Scrum) , 24
- Agile Web Applications with Python , 28
- Artificial Intelligence , 144
- Artificial intelligence in safety-critical applications , 132
- Business Information Systems , 30
- Chance and Risk Management in Digitized Value Networks , 32
- Classic Project Management Modernized , 124
- Compiler , 36
- Computer Games Development , 40
- Concepts of Database Technology , 128
- Corporate Entrepreneurship , 42
- Data communication in the vehicle , 48
- Digital Business Leadership Skills , 56
- Digital Innovation , 52
- Digital Transformation in Organizations , 60
- Electronic Trading Systems , 70
- Embedded Linux , 72
- Flying Robots , 74
- Formula Student Driverless , 76
- Fullstack Web Development , 80
- Fundamentals of ABAP , 12
- Fundamentals of Data Communications , 46
- Fundamentals of DevOps , 84
- Hard- and software for the internet of things , 86
- Industrial Data Processing , 94
- Industrial Image Processing , 90
- Information technology and the environment , 98
- Integrated Business Processes with SAP ERP , 102
- Interaction Engineering , 104
- Interactive Computer Graphics , 108
- Introduction to Natural Language Processing , 64
- Introduction to Robotics , 68
- IT Forensics , 116
- IT Security , 118
- IT Sourcing and Cloud Transformation , 122
- IT-Consulting , 112
- Lean IT & Enterprise Architecture , 136
- Linux LPIC , 140
- Mobile Robots , 148
- Network Penetration Testing , 152
- Neural Networks and Deep Learning , 154
- NoSQL , 158
- Object-oriented programming with Python , 160
- Open Source Software Development , 164
- Pattern recognition and machine learning , 150
- Practical Robotics with Matlab , 168
- Process Intelligence , 172
- Programming of Web Applications , 180
- Programming using Databases , 174
- Programming with Python , 176
- Project - Research and Transfer , 188
- Project Jupyter , 184
- RFID and NFC technology , 192
- Service Learning Project , 202

Software engineering for data-intensive
systems , 204

Software Project Management , 206

Startitup - Entrepreneurial Thinking
and Business Design , 208

Swabia Innovation Masterclass , 194

University Innovation Project , 88

Visual Thinking for Business , 212