h_da
HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES
fbi
FACHBEREICH INFORMATIK

Module descriptions Examination Regulations Bachelor 2021

Algorithms and data structures

English Title:	Algorithms and Data Structures
Document number:	30.7118
Language:	GERMAN
Assignment:	Bachelor 2021 - 1st semester Bachelor dual KITS 2021 - 1st semester Bachelor dual KoSI 2021 - 1st semester Bachelor KMI 2021 - 1st semester
Teaching form:	VÜ = lecture with integrated exercise
SWS:	4
CP:	5
Exam:	Written exam
Learning Objectives:	Students will gain the competencies to • Know, evaluate and apply the most important basic algorithms and data structures for sorting and searching as well as for graph-based problems. • recognize basic algorithmic problems and be able to select suitable algorithms and data structures, • be able to assess the runtime and space requirements of algorithms
Course content:	The term algorithm • Elementary data structures: one-dimensional and multidimensional fields and strings • Iteration and recursion • Sorting and search algorithms • Introduction to complexity theory (O-notation) • Advanced data structures (e.g. list, heap, stack, queue) • Hash algorithms • Binary search trees • Balanced trees • Graphs (representation and implementation alternatives) • Graph algorithms (including breadth-first and depth-first search, topological sorting, Dijkstra algorithm, A* algorithm) • Problem solving strategies (divide-and-conquer, backtracking, dynamic programming)
Literature:	Ottmann, Thomas; Widmayer, Peter (2017): Algorithms and Data Structures. 6th, revised edition. Springer-Verlag. • Sedgewick, Robert; Wayne, Kevin (2014): Algorithms. Algorithms and data structures. 4th, updated edition. Hallbergmoos: Pearson Deutschland GmbH (IT Informatics). Available online at http://lib.myilibrary.com/detail.asp?id=650968.
Department:	Computer Science

Expert group: Programming

Module responsibility: Arnim Malcherek

IT Security

English Title: IT Security

Document numbers: 30.7126 [PVL 30.7127]

Language: GERMAN

Assignment:
Bachelor 2021 - 1st semester
Bachelor dual KITS 2021 - 1st semester
Bachelor dual KoSI 2021 - 1st
semester Bachelor 2014 - 1st
semester
Bachelor dual KESS 2014 - 1st semester
Bachelor dual KITS 2014 - 1st semester
Bachelor dual KoSI 2014 - 1st semester
Bachelor KMI 2014 - 1st semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded(successful participation in the

internship) Learning objectives: Students acquire the competencies to

- Know basic concepts and the different areas of IT systems security,
- Know the safety objectives for a system design,
- understand the typical course of an attack on IT systems,
- know typical security risks for IT systems, analyze typical hazards and be able to take adequate countermeasures,
- know different evaluation schemes for IT security and are able to evaluate the security level of an IT system,
- to be able to develop an IT security strategy,
- Know the tension between usability and security.

Course content:

Basic concepts:

  - Security objectives (e.g. confidentiality, integrity, authenticity, availability, anonymization)
  - Hazard, risk, authorization
  - Attacks: e.g. spoofing, sniffing, denial of service
  - Data protection, privacy by design, legal framework

- Basics:
  - Cryptography: encryption, signature Random number generators
  - Data and instance authentication
  - Public Key Infrastructures
  - IT Forensics
- Areas and disciplines of IT security: system security, Internet security, security for ubiquitous computing, secure software development
- Phases of an attack (e.g. via the network, social engineering) and countermeasures (hardened operating systems, firewalls, intrusion detection systems)
- Security management: IT security through a structured approach, IT security as a continuous process, history, national standards

(BSI-Grundschutz), international standards (Common Criteria), separation of functional security requirement and trustworthiness requirements.

- Security and usability

Literature:

C. Eckert: IT-Sicherheit, Konzepte-Verfahren-Protokolle, Oldenbourg-Verlag, 2011

- D. Gollmann: Computer Security, John Wiley & Sons, 2010
- C. Adams, S. Llyod: Understanding PKI, Addison-Wesley, 2010
- B. Schneier, N. Ferguson, T. Kohno: Cryptography Engineering - Design Principles and Practical Applications, Wiley Publishing, 2011.
- Recent IT security publications (e.g., from conferences such as IEEE S&P, ACM CCS, Crypto).

Forms of work / resources: Seminar-stylelecture with practical

course Department: Computer Science

Expert group: IT Security

Module responsibility: Christoph Krauß

Mathematics for computer scientists 1

English Title: Math 1

Document numbers: 30.7120 [PVL 30.7121]

Language: GERMAN

Assignment:

Bachelor 2021 - 1st semester
Bachelor dual KITS 2021 - 1st semester
Bachelor dual KoSI 2021 - 1st
semester Bachelor KMI 2021 - 1st
semester
Bachelor 2014 - 1st semester
Bachelor dual KITS 2014 - 1st
semester Bachelor dual KoSI 2014 -
1st semester Bachelor KMI 2014 - 1st
semester

Teaching form: V+Ü = Lecture+Exercise

SWS: 4+2

CP: 7.5

Exam: Written exam

PVL (e.g. internship): ungraded

Learning Objectives: Students gain the skills to work through important concepts and structures of discrete mathematics and linear algebra in more advanced computer science courses. They will learn basic mathematical working methods and skills. They understand the relationship between mathematical methods and selected algorithms.

Course Content:

Boolean Algebra

- Set theory, combinatorics
- Congruence calculus, algebraic structures
- Functions, relations
- Matrices, linear systems of equations
- Vector spaces, linear mappings Eigenvalues and eigenvectors

Literature:

G. Teschl & S. Teschl: Mathematics for Computer Scientists, Vol. 1, Springer, 2013.

- M. Brill: Mathematics for Computer Science. 2nd edition, Hanser Verlag, 2005

Department: Mathematics and Natural Sciences

Subject group: Mathematics curriculum in computer science

Module responsibility: Julia Kallrath

Programming 1

English title: Programming 1

Document numbers: 30.7122 [PVL 30.7123]

Language: GERMAN

Assignment:
Bachelor 2021 - 1st semester
Bachelor dual KITS 2021 - 1st semester
Bachelor dual KoSI 2021 - 1st semester Bachelor KMI 2021 - 1st semester
Bachelor 2014 - 1st semester
Bachelor dual KITS 2014 - 1st semester Bachelor dual KoSI 2014 - 1st semester Bachelor KMI 2014 - 1st semester

Teaching form: V+P = Lecture+Internship

SWS: 4+2

CP: 7.5

Exam: practical exam

PVL (e.g. internship): ungraded (Students will work on the following during the internship. independently programming tasks. For admission, it is necessary that all practical tasks have been successfully completed by the students and certified by the lecturer).

Learning Objectives: Students will gain the competencies to

- Understand and be able to use the basic language tools of a modern programming language,
- be able to analyze and create simple programs with structured and typed program elements,
- to be able to operate a modern programming environment including debugger,
- be able to practically use basic elements of the C++ programming language and the C++ standard library (e.g. for text-oriented input and output),
- Have acquired basic knowledge and skills for understanding practical programming of information processing systems.

Course contents: structure of a program, control of the program flow.

- typed storage of values
- Functions, recursion
- first language tools from the C++ standard library (input/output, strings, first containers, exceptions)
- Pointer, References, Smart Pointer
- dynamic memory management
- user-defined types (enum, union, struct, class)
- object-oriented programming
- Relationships between classes (composition, aggregation, inheritance (ad-hoc polymorphism))
- Processing of text files, stream concept (stringstream, filestream, stream operators)

- exemplary, practical implementation of simple algorithms and data structures, such as data fields, lists, simple searching and sorting

Literature:

Breymann, Ulrich (2020): C++ programming. Learn C++ - use it professionally - use solutions: up-to-date to C++20; 6th revised edition. Munich: Hanser.

- Stroustrup, Bjarne (2014): Programming. Principles and practice using C++. 2nd ed. Upper Saddle River, NJ: Addison-Wesley. Available online at http://proquest.tech.safaribooksonline.de/9780133796759.

Department: Computer Science

Expert group: Programming

Module responsibility: Arnim Malcherek

Technical basics of computer science

English Title: Technical Principles of Computer Science

Document numbers: 30.7108 [PVL 30.7109]

Language: GERMAN

Assignment:
Bachelor 2021 - 1st semester
Bachelor dual KITS 2021 - 1st semester
Bachelor dual KoSI 2021 - 1st
semester Bachelor 2014 - 1st
semester
Bachelor dual KESS 2014 - 1st semester
Bachelor dual KITS 2014 - 1st semester
Bachelor dual KoSI 2014 - 1st semester
Bachelor 2007 - 1st semester
KoSI 2007 - 1st semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded(successful participation in the

internship) Learning objectives: Students acquire the competencies to

- Understand the different ways numbers and alphabets are represented in calculators.
- Know simple fundamentals of electronics for passive and active components.
- Have skills in formal and programming language circuit description.
- Know methods for synthesis and analysis of circuits and their minimization.
- to know technical forms of realization of circuits.
- Know procedures and concepts for encoding digital data.
- Understand the technical constraints and limitations of current concepts for component realization.

Course Content:

Electronic fundamentals: current and voltage, active and passive devices, semiconductor technologies.

- Moore's Law, components of a computer, computer generations
- Switching algebra: Boolean postulates, complete systems, disjunctive and conjunctive normal form
- Minimization: algebraic shortening rules, graphical (Karnaugh-Veitch

diagram), and algorithmic methods (Quine and McCluskey).

- Switching networks: Adder, (de)multiplexer
- Switching mechanisms: various flip-flop types, asynchronous and synchronous switching mechanisms, counters, shift registers
- Finite automata: Moore and Mealy automata, state diagrams, state transition tables.
- Calculator arithmetic: number representations, fixed point representation, floating point representation, addition, subtraction, multiplication.
- Semiconductor memory technology: ROM, static RAM, dynamic RAM, flash, new technologies for main memory
- Mass Storage Technologies
- Programmable logic devices (e.g. PAL, CPLD, FPGA) and hardware description languages
- Information and coding: measurement of information, data compression, code protection

Literature:

Mayer, R. S.: Technical Foundations of Computer Science, Script, 2013.

- Schiffmann, W.; Schmitz, R.: Technische Informatik 1 & 2; Springer Verlag; 5th ed.; 2004/2005.
- Hoffmann, D.W.: Grundlagen der Technischen Informatik; Hanser Verlag; 3rd ed; 2013.
- Beuth, K.: Digitaltechnik; Vogel Fachbuch; 13th ed.; 2006; ISBN 978-3834330840.
- Siemers, Ch.; Sikora, A. (Eds.): Taschenbuch Digitaltechnik; Hanser Fachbuch; 2nd ed.; 2007.
- Tietze, U.; Schenk, C.; Gamm, E.: Halbleiter-Schaltungstechnik; Springer Verlag; 14th ed.; 2012.

Forms of work / resources: Seminar-like lecture with computer-aided examples as well as Lecture hall exercises, in the practical course the understanding of the material of the course is supported and deepened with the help of experiments.

Department: Computer Science

Expert group: Computer Engineering

Module responsibility: Stefan Rapp

Operating systems

English Title: Operating Systems

Document numbers: 30.7220 [PVL 30.7221]

Language: GERMAN

Assignment: Bachelor 2021 - 2nd semester
Bachelor dual KITS 2021 - 2nd semester Bachelor dual KoSI 2021 - 2nd semester Bachelor KMI 2021 - 2nd semester
Bachelor 2014 - 3rd semester
Bachelor dual KESS 2014 - 6th semester Bachelor dual KITS 2014 - 4th semester Bachelor dual KoSI 2014 - 4th semester Bachelor KMI 2014 - 3rd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. practical course): ungraded (processing of programming tasks. The tasks must be solved and successfully tested).

Course prerequisite: SPOn 2021: The module "Programming 1" must have been successfully completed. SPOn 2014: The module "Programming, Algorithms and Data Structures 1" must have been successfully completed and an examination attempt "Programming, Algorithms and Data Structures 2" must have been made.

Learning Objectives: Students will gain the competencies to distinguish between the different types of operating systems and to be able to select and use appropriate operating systems for given use cases. In addition, students should be able to implement, extend and use system-related software, analyze the behavior of operating systems and, if necessary, correct, improve and extend them, and also apply the algorithms and design principles of operating systems to the development of middleware and applications. The knowledge acquired is also the basis for entry into the development of operating system software such as device drivers.

Course content:

Architectures and operating modes

- Address spaces
- Process and thread concept, scheduling
- Synchronization
- Interprocess communication
- Jams
- File systems
- Protection mechanisms, security aspects
- Exemplary consideration of current operating systems

Literature:

Tanenbaum: Moderne Betriebssysteme, Verlag Pearson Studium, 3. akt. Edition, 2009

- Nehmer: Systemsoftware, dpunkt Verlag, 2. akt. und überarb. Auflage, 2001

Forms of work / resources: Seminar lecture and practical course in a laboratory with heterogeneous System environment.
Resources: Lecture notes of the lecturers, exercise sheets and practical course material

Department: Computer Science

Division: Operating Systems / Distributed Systems

Module responsibility: Lars-Olof Burchard

Computer Architecture

Module numbers: 30.7222 [PVL 30.7223]

Language: english

Study programme:
Bachelor 2021 - 2nd semester
Bachelor dual KoSI 2021 - 2nd semester
Bachelor 2014 - 2nd semester
Bachelor dual KESS 2014 - 2nd
semester Bachelor dual KITS 2014 -
2nd semester Bachelor dual KoSI
2014 - 2nd semester Bachelor KMI
2014 - 2nd semester

Type of course: V+P = Lecture+Practical

Weekly hours: 3+1

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded (ungraded practical exercises)

Requiredknowledge: Basicknowledge in technical principles of computer

science Learningobjectives: The students acquire the competences to

- know the basic principles of organization and architecture for the construction of computer systems.
- assess the boundary conditions and limitations of current computer systems
- understand a machine language, to apply it in a system-oriented manner and to convert high-level language constructs into machine language.
- understand the interaction of different hardware and software concepts.

Content: Introduction to the history of computers

- Computer arithmetic
- Computer organization: hardware operations, hardware operands, representation of commands, control structures
- Processor: data path, control path, microprogramming, pipelines Hardware Architectures: Von Neumann, Harvard
- Instruction set architectures using ARM processors as an example
- Concepts: Subprograms, Stacks, Indirect Addressing, Calling Standards, Implementation of high-level language programming in in assembler
- Memory organization and memory hierarchies: caches
- Superscalar architectures

Literature: Patterson, David A., Henessy, John L.; Computer Architecture, A Quantitative Approach, Morgan Kaufmann, 2017.

- Tanenbaum, Andrew, S.; Structured Computer Organization, 6th Edition, 2013.
- Furber, Steve; ARM System-on-Chip Architecture, Addison-Wesley, 2000.

Lecture style / Teaching aids: Seminaristiclecture with seminar and laboratory. Resources include lecture notes, example programs and software tools.

Faculty: Computer Science

Expert group: Computer engineering

Responsibility: Thomas Horsch

Mathematics for Computer Scientists 2

English Title: Math 2

Document numbers: 30.7124 [PVL 30.7125]

Language: GERMAN

Assignment: Bachelor 2021 - 2nd semester
Bachelor dual KITS 2021 - 2nd semester Bachelor dual KoSI 2021 - 2nd semester Bachelor KMI 2021 - 2nd semester
Bachelor 2014 - 3rd semester
Bachelor dual KITS 2014 - 3rd semester Bachelor dual KoSI 2014 - 3rd semester Bachelor KMI 2014 - 3rd semester

Teaching form: V+Ü = Lecture+Exercise

SWS: 4+2

CP: 7.5

Exam: Written exam

PVL (e.g. internship): ungraded

Learning objectives: Students gain the competencies to know concepts and structures of calculus and stochastics important for higher level computer science courses. They will learn basic mathematical working methods and skills. They will be able to apply mathematical methods from calculus and stochastics to solve problems.

Teaching content:

sequences, series
- Continuous functions, important function classes, e.g. exponential and trigonometric functions
- Differential and integral calculus for functions of a real variable
- Interpolation and approximation: polynomial interpolation according to Newton
- Descriptive statistics
- Probabilities and stochastic independence
- Random variables and their moments
- Special probability distributions
- Hypothesis testing
- Linear regression

Literature:

G. Teschl & S. Teschl: Mathematics for Computer Scientists, Vol. 1, Springer, 2013.
- G. Teschl & S. Teschl: Mathematics for Computer Scientists, Vol. 2, Springer, 2006.
- L. Fahrmeir, C. Heumann, et al, Statistics: The road to data analysis, Springer, 2016.

Department: Mathematics and Natural Sciences

Subject group: Mathematics curriculum in computer science

Module responsibility: Julia Kallrath

Object-oriented analysis and design

English Title: Object-Oriented Analysis and Design

Document numbers: 30.7206 [PVL 30.7207]

Language: GERMAN

Assignment:

Bachelor 2021 - 2nd semester
Bachelor dual KITS 2021 - 2nd
semester Bachelor dual KoSI 2021 -
2nd semester Bachelor KMI 2021 -
2nd semester
Bachelor 2014 - 2nd semester
Bachelor dual KESS 2014 - 2nd
semester Bachelor dual KITS 2014 -
2nd semester Bachelor dual KoSI
2014 - 2nd semester Bachelor KMI
2014 - 2nd semester
Bachelor 2007 - 2nd semester
KoSI 2007 - 2nd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (successful participation in the internship)

Course prerequisite: SPOn 2021: The module "Programming 1" must have been successfully completed.

SPOn 2014: An examination attempt "Programming, Algorithms and Data Structures 1" must have been made.

Learning objectives:

Students acquire the competencies to master the basic principles of object orientation and can apply them in analysis, design and programming.
The results from analysis and design can be expressed as UML diagrams and specified in a case tool. The UML model can then be implemented in code. The students know basic quality aspects and important rules of "good design" (e.g. cohesion, freedom from redundancy, design patterns).
The knowledge and skills acquired with the help of the module are fundamental for computer science education ("core computer science"). Thus, this module forms an important basis for various other modules resp.
Courses such as "Databases", project "System Development", courses with focus on application development as well as the practical phase and bachelor thesis.

Course content:

Classification of OOAD in software engineering (central terms).
- Principles of object orientation and model building
- Phases in the development of object-oriented systems: object-oriented analysis, design, programming
- UML (basics, notation, semantics, important diagrams, modeling rules)
- Use of modeling and development tools
- Basic aspects of software quality
- Rules of "good design" for a design model

Literature:

Balzert, Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, Spektrum Akademischer Verlag, 2012.
- Chris Rupp et al, UML 2 glasklar: Praxiswissen für die UML-Modellierung, Carl Hanser Verlag GmbH & Co, 2012.
- Bernd Oestereich, Stefan Bremer, Analyse und Design mit der UML: Objektorientierte Softwareentwicklung, Oldenbourg Wissenschaftsverlag, 2013.
- Karl Eilebrecht, Gernot Starke, Patterns kompakt - Entwurfsmuster für effektive Software-Entwicklung, Springer Vieweg, 2013.

Forms of work / resources:

Seminar-like lecture, small groups in the practical course, use of a Modeling tool, lecture hall exercises with supplemental examples, exam examples, presentation slides.

Department: Computer Science

Professional group: Software Engineering

Module responsibility: Frank Bühler

Programming 2

English title: Programming 2

Document numbers: 30.7128 [PVL 30.7129]

Language: GERMAN

Assignment:

Bachelor 2021 - 2nd semester
Bachelor dual KITS 2021 - 2nd
semester Bachelor dual KoSI 2021 -
2nd semester Bachelor KMI 2021 -
2nd semester
Bachelor 2014 - 2nd semester
Bachelor dual KITS 2014 - 2nd semester

Bachelor dual KoSI 2014 - 2nd semester
Bachelor KMI 2014 - 2nd semester

Teaching form: V+P = Lecture+Internship

SWS: 4+2

CP: 7.5

Exam: practical exam

PVL (e.g. internship): ungraded (Students will work on the following during the internship. independently programming tasks. For admission, it is necessary that all practical tasks have been successfully completed by the students and certified by the lecturer).

Course prerequisite: SPOn 2021: The module "Programming 1" must have been successfully completed. SPOn 2014: An examination attempt "Programming, Algorithms and Data Structures 1" must have been made.

Learning Objectives: Students will gain the competencies to

- understand typical language tools of a modern programming language in depth and be able to apply them practically at an advanced level,
- Be able to analyze and create more complex programs with advanced program elements,
- be able to practically use important elements of the C++ programming language and the C++ standard library,
- To have acquired fundamental knowledge and skills for deeper and advanced understanding of practical programming of information processing systems,
- to be able to use a modern version management tool fundamentally.

Course contents: Consolidation and extension of selected programming techniques relevant in practice, such as.

- Processing of structured text files and binary files
- Generic programming, templates (parametric polymorphism)
- Simple graphical user interfaces
- Unit tests
- Use of libraries
- other language tools from the C++ standard library (e.g. containers, iterators, stream iterators)
- exemplary, practical implementation of advanced algorithms and data structures, such as search trees, hash tables, index or pointer tables, graphs, path search
- Outlook on other programming languages and paradigms, e.g.
  - Event oriented programming
  - Functional programming
  - Declarative programming
  - Regular expressions
  - Lambda functions

Literature: U.Breymann: C++ Programmieren, 6th edition; Hanser; 2020

- B.Stroustrup: Introduction to Programming with C++; Pearson Studium; 2010

Department: Computer Science

Expert group: Programming

Module responsibility: Arnim Malcherek

Computer architecture

English Title:	Computer Organization and Design
Document numbers:	30.7106 [PVL 30.7107]
Language:	GERMAN
Assignment:	Bachelor 2021 - 2nd semester Bachelor dual KoSI 2021 - 2nd semester Bachelor 2014 - 2nd semester Bachelor dual KESS 2014 - 2nd semester Bachelor dual KITS 2014 - 2nd semester Bachelor dual KoSI 2014 - 2nd semester Bachelor KMI 2014 - 2nd semester Bachelor 2007 - 1st semester KoSI 2007 - 1st semester
Teaching form:	V+P = Lecture+Internship
SWS:	3+1
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded (successful participation in the internship)
Required prerequisite knowledge:	Basic bachelor's level knowledge in Technical Foundations of Computer Science.

Learning Objectives:

Students will gain competency in the following knowledge and skills:

- Knowledge: Students will
  - Know the basic organizational and architectural principles for building computer systems.
  - understand the interaction of different hardware and software concepts.
- Skills: Students will
  - can assess the constraints and limitations of current computer systems
  - are able to understand a machine language, to apply it in a system-oriented way and to convert high-level language constructs into machine language.

Course Content:

Introduction to the history of computers

- Calculator Arithmetic
- Computer organization: operations of the hardware, operands of the hardware, representation of commands, control structures
- Processor: data path, control path, microprogramming, pipelines
- Hardware architectures: Von Neumann, Harvard
- Instruction set architectures using the example of ARM processors
- Concepts: Subroutines, stacks, indirect addressing, calling standards,
- Conversion of high-level language constructs into assembler
- Exception handling
- Memory organization and memory hierarchies: caches

Literature:

Patterson, David A., Henessy, John L.; Computer Organization and Design; Spektrum Akademischer Verlag; 3rd ed. 2005.

- Tanenbaum, Andrew, S.; Computer Architecture. Structures - Concepts - Fundamentals; Pearson Studies; 5th ed. 2005.
- Furber, Steve; ARM Computer Architectures for System-on-Chip Design; mitp-Verlag, Bonn; 1st ed. 2002.

Forms of work / resources: Seminar-like lecture with computer-aided examples as well as Lecture hall exercises, in the practical course the understanding of the material of the course is supported and deepened with the help of experiments.

Department: Computer Science

Expert group: Computer Engineering

Module responsibility: Thomas Horsch

Databases

English Title: Databases

Document numbers: 30.7326 [PVL 30.7327]

Language: GERMAN

Assignment:
Bachelor 2021 - 3rd semester
Bachelor dual KITS 2021 - 4th
semester Bachelor dual KoSI 2021 -
4th semester Bachelor KMI 2021 - 3rd
semester
Bachelor 2014 - 3rd semester
Bachelor dual KESS 2014 - 4th
semester Bachelor dual KITS 2014 -
4th semester Bachelor dual KoSI 2014
- 4th semester Bachelor KMI 2014 -
3rd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (Weekly completion of internship assignments. All Internship tasks must be successfully completed in order to successfully pass the PVL).

Course Prerequisite: SPOn 2021: The modules "Programming 2", "Object-oriented analysis and Design",
"Algorithms and Data Structures" must have been successfully completed.
SPOn 2014: The module "Programming, Algorithms and Data Structures 1" must have been successfully completed and an examination attempt "Programming, Algorithms and Data Structures 2" must have been made.

Learning Objectives: Students will gain the competencies to,

- Be able to transform a UML class diagram into a relational data model (both manually and with a CASE tool) and understand ER models.
- Be able to implement a database schema using SQL-DDL and be able to insert, query and modify data using SQL-DML.
- To be able to implement integrity constraints with the help of constraints
- Be able to practically apply database rights concepts
- Be able to implement database application logic in an application program as well as using an OR mapper.
- Know and be able to appropriately apply concepts of transaction management and database index structures.
- Know basic concepts of NoSQL database systems

Course content: Conceptual data modeling based on the UML class diagram.
and the ER model

- Relational data modeling
- SQL-DDL, SQL-DML, system catalog
- Access to databases from an application
- Use of OR mappers
- Transaction concept
- Internal data organization: indexes (B-trees, hash procedures)
- NoSQL databases

Literature:

A. Heuer, K.-U. Sattler, G. Saake. Datenbanken: Konzepte und Sprachen, 6. Edition mitp 2018;

- A. Kemper, A. Eickler: Database Systems. Eine Einführung, Oldenbourg, 10th edition 2015;
- B. Müller, H. Wehr: Java Persistence API 2 : Hibernate, EclipseLink, OpenJPA und Erweiterungen, Hanser, 2012;
- C. J. Date, An Introduction to Database Systems, Addison Wesley 2004;

Department: Computer Science

Professional group: Databases

Module responsibility: Peter Muth

Human Computer Interaction

Document number: 30.7328

Language: GERMAN

Assignment: Bachelor 2021 - 3rd semester
Bachelor dual KITS 2021 - 4th semester Bachelor dual KoSI 2021 - 4th semester Bachelor KMI 2021 - 3rd semester
Bachelor 2014 - 3rd semester
Bachelor dual KITS 2014 - 4th semester Bachelor dual KoSI 2014 - 4th semester Bachelor KMI 2014 - 3rd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Examination: Project Report

Course Prerequisite: The module "Programming 2" must have been successfully

completed. Required Prior Knowledge: Basic knowledge at bachelor's level in object-oriented Analysis and design and basic knowledge of statistics.

Learning objectives: Students apply a user-centered development method and work through the associated phases in a small team. In doing so, they research target group-specific requirements and usage contexts, develop prototypes using suitable frameworks, and then evaluate these using heuristic and empirical testing methods. After successfully completing the module, they will have acquired the competencies to

- the phases, tools and methods of the user-centered to be able to explain software development.
- To be able to elicit target group-specific requirements for software systems by means of qualitative methods of user research (e.g., with partially standardized interviews).
- Psychological and social concepts of interaction between people and

to describe the computer with adequate terms and to be able to classify it in the context of human-technology interaction.
- in a user-centered design process gui-based
  Applications in a concrete application context using MCI standards
  in a problem-adequate manner and in relation to various
  develop user groups and apply relevant software design patterns (e.g. event handling, MVC).
- GUIs and interaction patterns in a concrete application context based on evaluate and assess MCI standards and, if necessary, develop design recommendations.

Course content:

User Research
- User-centered development methods
- Requirements analysis
- Design and perceptual-psychological design principles
- Gender and diversity aspects
- HCI Standards
- Prototyping techniques
- GUI design and interaction patterns
- Usability evaluation methods
- Development of a prototype (e.g. app development with Android Studio, AR/VR with Unity or others).

Literature:

Donald Norman, The Design Of Everyday Things (2013).
- Alan Dix, Janet Finlay, Gregory D. Abowd, Russell Beale, Human-Computer Interaction (2007).
- Everett N. McKay, UI is Communication: How to Design Intuitive, User Centered Interfaces by Focusing on Effective Communication (2013).

Department: Computer Science

Professional group: Multimedia and graphics

Module responsibility: Ute Trapp

Informatics and society

English Title: Information Technology and Society

Document number: 30.7514

Language: GERMAN

Assignment:

Bachelor 2021 - 3rd semester
Bachelor dual KITS 2021 - 4th semester Bachelor dual KoSI 2021 - 4th semester Bachelor 2014 - 4th semester
Bachelor dual KESS 2014 - 6th semester Bachelor dual KITS 2014 - 6th semester Bachelor dual KoSI 2014 - 6th semester Bachelor KMI 2014 - 4th semester

Teaching Form: S = Seminar

SWS: 2

CP: 2.5

Examination: Lecture, participation and, if applicable, a written paper; details will be determined at the beginning of the course.

Learning Objectives: Students acquire the competencies to understand the conditions, effects, and consequences of informational action and design in society.

to analyze, understand and judge.
They should learn the basics for perceiving their own responsibility towards those affected by the use of information technology and for translating this into individual and collective, socially effective and responsible action.

Course content: The course is not oriented towards fixed course content but rather takes into account some aspects from the following exemplary catalog of topics, depending on the topicality of the subject and the interests of the teachers conducting the course and the students:

- New perspectives on computer science; social and cultural history of the Data processing, computer science as a science, philosophy of science of computer science
- Areas of application of ICT technologies: production, healthcare, education, ...
- Overarching effects and requirements for action, requirements for action, labor market and occupational structure, "women and information technology", thought and communication structures
- Perspectives for a socially oriented computer science: work analysis and software development, software ergonomics, AI and expert systems, computer networks and distributed systems.
- Computer science between theory and practice: technology assessment, ethics and computer science, professional practice, social situation and awareness of computer scientists and computer scientists

Literature: Primarily recent journal articles;

- J. Friedrich et al: Informatics and Society, Spektrum, 1994
- A. Grunwald: Technology Assessment; Berlin, 2010
- G. Stamatellos: Computer Ethics, A global perspective, Sudbury, 2007
- J. Weizenbaum: Power of the Computer - Powerlessness of Reason, 2000

Forms of work / resources: The topics compiled at the beginning of the seminar will be presented through presentations by the students and subsequently discussed in the seminar. The number of participants is limited.
Supplementary materials: video, film, etc. Presentations on special topics

Department: SuK Accompanying Studies

Division: Social and Cultural Aspects of Computer Science

Module responsibility: Bettina Harriehausen

Information Technology and Society

Module number: 30.7516

Language: english

Study programme: Bachelor 2021 - 3rd semester
Bachelor dual KITS 2021 - 4th semester
Bachelor dual KoSI 2021 - 4th semester

Type of course: S = Seminar

Weekly hours: 2

Credit Points: 2.5

Exam: Oral presentation, active participation, and a written presentation; details will be fixed at the beginning of the seminar.

Learning objectives: The students shall analyze, understand and evaluate the terms, effects and implications of the actions and designs of computer scientists in society. They shall acquire the basic concepts to realize their personal responsibility with regard to those who are affected by the use of computer science technologies and they shall learn to implement an individual as well as common socially effective and responsible behavior.

Content:

The course is not based on fixed teaching content but it takes aspects into consideration, which show a specific timeliness and topicality or are of special interest to either the students or the lecturers. These may be aspects from the following catalogue of topics:

- New views of Computer Science; social and cultural history of data processing, Computer Science as a Science, philosophy of Computer Science
- Scope of Computer Science technologies: production, health science, education,...
- Comprehensive effects and action requirements, action requirements, job market and structure of the profession, "women in computer science", structures of mental activity and communication
- Perspectives of a socially oriented Computer Science: work analysis and software development, software ergonomics, A.I. and expert systems, computer networks and distributed systems
- Computer Science between theory and practice: estimation of the consequences of technology, ethics and computer science, job practice, social situation and awareness of computer scientists

Literature:

Primarily current literature:

- J. Friedrich et al: Informatics and Society, Spektrum, 1994
- A. Grunwald (2010): Technology Assessment; Berlin
- G. Stamatellos (2007): Computer Ethics, A global perspective, Sudbury.
- J. Weizenbaum (2000): Power of the Computer - Powerlessness of Reason

Lecture style / Teaching aids:

The topics given out at the beginning of the semester will be presented by students' presentations and will be discussed in the seminar group. The number of participants is limited.
Additional material: video, film to specific topics

Faculty: SuK Accompanying Studies

Expert group: Social and cultural aspects of computer science

Responsibility: Bettina Harriehausen

Computer networks

English title: Computer networks

Document numbers: 30.7330 [PVL 30.7331]

Language: GERMAN

Assignment:

Bachelor 2021 - 3rd semester
Bachelor dual KITS 2021 - 4th
semester Bachelor dual KoSI 2021 -
4th semester Bachelor 2014 - 2nd
semester
Bachelor dual KESS 2014 - 4th
semester Bachelor dual KITS 2014 -
4th semester Bachelor dual KoSI 2014
- 2nd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. practical course): ungraded (performance of laboratory experiments and projects, documentation as laboratory report or protocol).

Prerequisite: The module "Programming 1" must be successfully completed. RequiredPrevious

knowledge: Mathematics 2 (Statistics)

Learning Objectives:

Knowledge: Students will know the basic structure and Structure of computer networks and the main communication functions and protocols of the Internet.

- Skills: Students will be able to determine the performance limits of telecommunication systems and measure important performance variables of IP-based networks. They can name the relevant parameters and functions and assign them to the protocol stack.
- Competencies: Students will understand the design scope and the essential design decisions in the development of telecommunications systems. They understand the interaction of the functions and protocols involved. You will be able to narrow down their performance and assess it based on relevant factors.

Course Content:

Fundamentals of computer networks: basic concepts, network architecture, OSI, hybrid and TCP/IP reference model

- Direct connection networks: hardware building blocks and coupling element, broadcast domains and collision domains
- Coding, generation of frames,
- Error detection, reliable transmission
- Multiple access in selected local area networks: Ethernet with Carrier Sense Multiple Access with Collision Detection (CSMA/CD), and WLAN with Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)
- Packet switching: switching and forwarding, bridges and LAN switches
- Internetworking: IPv4 and IPv6 addressing, IPv4 subneting, ARP, ICMP with PING and traceroute, DHCP and DNS
- Routing: computer networks as graphs, routing algorithms, distance vector routing and RIP
- Transport protocols: UDP, TCP flow control, congestion control and options
- Optional:
  - Connecting lines, structured cabling
  - Link State Routing and OSPF
  - Selected protocols of the application layer (HTTP, ...)
  - Architecture and implementation of Internet services
  - Socket API

Literature:

Larry L. Peterson and Bruce S. Davie, "Computer Networks: A Systems-Oriented Introduction," 3rd edition (2003) or higher, dpunkt.verlag.

- Andrew S. Tanenbaum, "Computer Networks," 4th edition (2003) or higher, Pearson Publishing.
- William Stallings, "Data and computer communications," Pearson Publishing, 2014.
- James F. Kurose and Keith W. Ross, "Computer Networks: The Top-Down Approach," Pearson Publishers.
- Christian Baun, "Computernetze kompakt (IT kompakt)", Springer-Verlag

Department: Computer Science

Professional group: Telecommunications

Module responsibility: Michael Massoth

S Catalog (SuK)

English title:	Catalog S (SuK)
Document number:	29.2/3xxxx
Language:	GERMAN
Assignment:	Bachelor 2021 - 3rd semester Bachelor KMI 2021 - 1st semester Bachelor 2014 - 1st semester Bachelor KMI 2014 - 4th semester Bachelor 2007 - 1st semester KoSI 2007 - 1st semester
Teaching Form:	Z = Supplementary course
SWS:	2
CP:	2.5
Examination:	Presentation and/or presentation plus discussion; written examination and/or written examination and discussion; will be announced at the beginning of the respective course.
Learning objectives:	The interdisciplinary competencies are intended to enable students to deal competently and critically with their own professional tasks and their own professional field and subject area in the context of society as a whole, to act in a future-oriented and responsible manner in a democratic and social constitutional state, and to engage in interdisciplinary cooperation and intercultural communication. The interdisciplinary competencies include competencies with professional field (key competencies) as well as those without direct professional reference (Studium Generale). Methodological competence, structuring and orientation knowledge from various disciplines for coping with future professional and social requirements in the professional field of computer science are taught and trained.
Course Content:	Selection from topics 1-4: 1. Work, occupation, self-employment (AB&S) 2. Culture and Communication (K&K) 3. Politics and Institutions (P&I) 4. Knowledge Development (K&I) (incl. techniques of scientific work and presentation techniques)
Literature:	s. Work forms / resources: Lecture and/or seminar Presentations on application areas (written + lecture), overhead, beamer
Department:	SuK Accompanying Studies
Division:	Social and Cultural Aspects of Computer Science

Software Engineering

Document numbers:	30.7318 [PVL 30.7319]
Language:	GERMAN
Assignment:	Bachelor 2021 - 3rd semester Bachelor dual KoSI 2021 - 4th semester Bachelor KMI 2021 - 3rd semester Bachelor 2014 - 3rd semester Bachelor dual KESS 2014 - 4th semester

Bachelor dual KITS 2014 - 4th semester Bachelor dual KoSI 2014 - 4th semester Bachelor KMI 2014 - 3rd semester
Bachelor 2007 - 3rd semester
KoSI 2007 - 3rd semester

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. practical course): ungraded (successful participation in the practical course

(testat))Prerequisite: SPOn 2021: The modules "Algorithms and Data Structures" must be

taken,

"Programming 2" and "Object-Oriented Analysis and Design" must be successfully completed.
SPOn 2014: The module "Programming, Algorithms and Data Structures 1" must have been successfully completed and an examination attempt "Programming, Algorithms and Data Structures 2" must have been made.

Prerequisite knowledge: Basic bachelor's level knowledge of object-oriented analysis and design.

Learning objectives: After completing the module, students will have acquired the competencies to work in a modern SW development project. They understand the importance and necessity of software engineering and how the different techniques from the module OOAD interact in a project. In addition, students will be able to apply basic techniques and methods (e.g., requirements analysis, architecture design, testing) for the various phases. Current process models can be compared and evaluated.
In addition, methods of technical project management (e.g. quality, test, configuration and risk management procedures) are learned from the perspective of the software developer. Graduates of the module are able to work independently in a project in different project roles and apply the common procedures.

Course Content: Fundamentals of software engineering (classification and terms)
Methods and techniques of the software life cycle:

- Requirements analysis (e.g. functional specification, functional and non-functional Requirements, content and language analysis, effort estimation, prioritization).
- Architecture and design (e.g., architectural styles, view model, design patterns, frameworks, interfaces)
- Implementation (programming guidelines)
- Testing (e.g., testing and testing procedures, testing strategies).
- Current procedure and process models (agile and classic)
- Technical management, such as
  - Software Metrics
  - Configuration and build management
  - Test Management
  - Continuous Integration
  - Risk Management
  - Change Management
- Application of a selection of the techniques in the practical course.

Literature:

Balzert, Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, Spektrum Akademischer Verlag, 2012.

- Sommerville, Software Engineering, Pearson Studies, 2012.
- Dan Pilone et al, Software Development from Head to Toe: A Book on the

	Participation and Understanding, O'Reilly, 2008. - Eric Freeman et al, Design Patterns from Head to Toe, O'Reilly, 2005.
Forms of work / resources:	Seminar-like lecture, small groups in the practical course, use of a Modeling tool, lecture hall exercises with supplementary examples, exam examples, presentation slides.
Department:	Computer Science
Professional group:	Software Engineering
Module responsibility:	Frank Bühler

Theoretical computer science

English Title:	Theoretical Computer Science
Document numbers:	30.7332 [PVL 30.7333]
Language:	GERMAN
Assignment:	Bachelor 2021 - 3rd semester Bachelor dual KITS 2021 - 4th semester Bachelor dual KoSI 2021 - 4th semester Bachelor KMI 2021 - 3rd semester
Teaching form:	V+Ü = Lecture+Exercise
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. practical course):	ungraded (submission of 50% correctly solved exercises).
RequiredPrior knowledge:	Basic knowledge at bachelor's level in mathematics, algorithms and data structures, and programming knowledge in mathematics and programming.
Learning Objectives:	Students will gain the competencies to - Develop an understanding of basic concepts, terms, and relationships from the subfields of automata theory and formal languages. - Develop an understanding of basic proof methods. elicit the ability to conduct simple proofs independently. - Gain knowledge of the performance of different descriptive tools and develop the ability to use the descriptive tools independently. - Maintain knowledge of the relationship between the performance and algorithmic controllability of different means of description. - Develop an understanding of non-deterministic machine models and their importance.
Course content:	Basic concepts: Words, alphabets, relations, operations over relations. - Formal languages: The word problem, relation to general decision problems, efficient versus non-efficient solution algorithms for the word problem. - Formal languages and automata theory: deterministic and non-deterministic finite automata, application of finite automata, equivalence of deterministic and non-deterministic finite automata, minimization algorithm, deterministic and non-deterministic basement automata. - Formal languages and grammars: Chomsky hierarchy, efficient

enumerable languages and the word problem (undecidability of the word problem), context-sensitive grammars and the word problem (relation to the complexity class NP), right-linear grammars, closure properties, regular expressions (incl. use in scripting languages), closure properties, right-linear languages and the word problem (finite automata), context-sensitive grammars and the word problem, context-free grammars and the word problem.
(Chomsky normal form, CYK algorithm), applications of context-free languages (syntax of programming languages, XML-based languages and languages describing communication protocols), context-free languages and non-deterministic basement automata, deterministic basement automata.

Literature: Hromkovic, J.: Theoretical Computer Science, Teubner Verlag, Stuttgart, 2002.

- Schöning, U.: Theoretische Informatik - kurz gefasst, Spektrum Akademischer Verlag, Heidelberg, 1997.
- Wegener, I.: Theoretische Informatik - eine algorithmenorientierte Einfu&#776;hrung, Teubner Verlag, Stuttgart, 1999

Department: Computer Science

Expert group: Theoretical Computer Science

Module responsibility: Steffen Lange

Introduction to Business Informatics

English Title: Introduction to Business Informatics

Document number: 30.7114

Language: GERMAN

Assignment: Bachelor 2021 - Elective S_5/6 Catalog - 4th semester.
Bachelor dual KoSI 2021 - elective S_5/6 catalog - 6th semester
Bachelor KMI 2021 - 4th semester
Bachelor 2014 - 1st semester
Bachelor dual KoSI 2014 - 1st semester
Bachelor KMI 2014 - 1st semester
Bachelor 2007 - Specialization WI: Business Informatics
Bachelor 2007/2004/2002/99 - Elective courses from the computer science
area KoSI 2007 - Specialization WI: Business Informatics
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Teaching form: V+Ü = Lecture+Exercise

SWS: 3+1

CP: 5

Exam: Written exam

Learning Objectives: Students acquire an overview of selected approaches, systems, methods and content of business information systems and acquire the competencies to be able to use and evaluate these independently and problem-related on simplified examples - for example, profitability analyses and
-calculations, business process analyses and models.
In doing so, the students also become familiar with the subject matter and basic concepts of business administration in business information systems, specifically the typical structure and usual functioning of companies and the corresponding business management concepts (e.g. economic efficiency principle), and are able to discuss these critically.
Building on basic knowledge about companies, students will be able to

	Discuss the fundamentals of business application systems and the concept of integrated information processing in enterprises. Interfaces to other subfields of computer science, business administration and other related disciplines, and their significance for business informatics are understood so that students can reproduce interdisciplinary knowledge, discuss it critically and transfer it independently to simple problems in business informatics and thereby apply it to solve these questions.
Course content:	Basic contexts and subject matter of business administration. • Selected operational functional areas and performance processes • Basic concept and methods of model building (data and process models) • Integrated operational information processing • Operational application systems to support operational functions • Industry-oriented application systems • Market, industry and labor market IT • Selected topics in business informatics
Literature:	Bea, F. X., Dichtl, E., and Schweitzer, M. (eds.), Allgemeine Betriebswirtschaftslehre, vol. 1: Grundfragen, Stuttgart, 9th ed. 2009. • Hansen / Neumann: Wirtschaftsinformatik 1, 10th edition, Stuttgart, 2009 • Holey / Welter / Wiedemann: Wirtschaftsinformatik, 2nd edition, Ludwigshafen, 2007 • Laudon / Laudon: Management Information Systems, 13th Edition, Prentice Hall 2013. • Mertens, Bodendorf, König et al.: Grundzüge der Wirtschaftsinformatik, Heidelberg, 11th ed. 2012 • Wöhe, Döring: Einführung in die Allgemeine Betriebswirtschaftslehre, 25th edition, Munich 2013 • Laudon, K.; Laudon, J.; Schoder, D.: Wirtschaftsinformatik - Eine Einführung. Pearson Studies. New edition 2015 (3rd, completely revised edition).
Work forms / resources:	Seminar-style lecture, script, supplementary examples, case studies
Department:	Computer Science
Professional group:	Information Systems
Module responsibility:	Urs Andelfinger

Development of web-based applications

English Title:	Development of Web-Based Applications
Document numbers:	30.7400 [PVL 30.7401]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective S_5/6 Catalog - 4th semester. Bachelor dual KoSI 2021 - elective S_5/6 catalog - 6th semester Bachelor KMI 2021 - 4th semester Bachelor 2014 - 4th semester Bachelor dual KoSI 2014 - 4th semester Bachelor KMI 2014 - 4th semester Bachelor 2007 - 4th semester KoSI 2007 - 4th semester
Teaching form:	V+P = Lecture+Internship
SWS:	3+1

CP:	5
Exam:	Written exam
PVL (e.g. practical course):	ungraded (successful participation in the practical
course)Prerequisite:	SPOn 2021: The modules "Algorithms and Data Structures" must be
taken,	"Programming 2" and "Object-Oriented Analysis and Design" must be successfully completed. SPOn 2014: The modules "Programming, Algorithms and Data Structures 1" and "Programming, Algorithms and Data Structures 2" must have been successfully completed.
Required Prior Knowledge:	Basic bachelor's level knowledge in user-centered Software development and databases
Learning Objectives:	Students will acquire the skills to develop a web application that will • contains static and dynamically generated content, • includes an appealing and usable design, • collects, checks and transmits data on the client side, • evaluates and processes the transmitted data on the server side, • integrates a database for storing the data, • meets current standards, • implements basic safety checks • is maintainable as software.
Course content:	HTML basics, hyperlinks, forms, validation. • Formatting and layout with CSS, layout concepts • Requirements of mobile devices • Client-side programming with JavaScript and HTML document object model • AJAX, JSON • Web server configuration, access protection, • Server-side object-oriented programming with PHP • Database connection • Communication via HTTP, sessions • System architecture • Safety aspects
Literature:	Stefan Münz, Clemens Gull, "HTML 5 Handbook", 2nd edition, Franzis Verlag GmbH, 2012. • Eric Freeman and Elisabeth Robson, "HTML5 Programming from Head to Toe," O'Reilly; 2012. • Mark Lubkowitz, "Programming and designing websites", Galileo Computing, 2007 • Carsten Möhrke, "Better PHP programming", Galileo Computing, 2009
Forms of work / resources:	Seminar lecture and practical course Script, supplementary examples, old exam questions
Department:	Computer Science
Professional group:	Multimedia and graphics
Module responsibility:	Ute Trapp

Microprocessor Systems

English Title: Microprocessor Systems

Document numbers: 30.7204 [PVL 30.7205]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective S_5/6 Catalog - 4th semester.
Bachelor dual KoSI 2021 - elective S_5/6 catalog - 6th semester
Bachelor 2014 - 3rd semester
Bachelor dual KESS 2014 - 4th
semester Bachelor dual KoSI 2014 -
4th semester Bachelor 2007 - 2nd
semester
KoSI 2007 - 2nd semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (successful participation in the internship)

Prerequisite: SPOn 2021: The modules "Programming 1" and "Computer Architecture" must be successfully completed.
SPOn 2014: The module "Programming, Algorithms and Data Structures 1" must have been successfully completed and an examination attempt must have been made in the module "Programming, Algorithms and Data Structures 2".

Prerequisite knowledge: Basic undergraduate level knowledge of computer architecture, technical principles of computer science, and programming.

Learning Objectives: Students will gain competencies to

- Understand the interaction of hardware and software concepts of a computer with its environment.
- Understand the structure of simple embedded systems and be able to develop them
- gain a profound understanding of information and data processing in real-time systems

Course content: Deepening of system-oriented programming with high-level languages (C/C++) and machine-oriented languages (e.g. ARM instruction set)

- Introduction to development environments for embedded systems
- Practical teaching of processors and peripherals in the form of modern microcontrollers with communication interfaces, timer and counter components, analog/digital converters and power management
- Basics of hardware abstraction
- Real-time capabilities in real system environments

Literature: Furber, Steve; ARM Computer Architectures for System-on-Chip Design; mitp-Verlag, Bonn; 1st ed.; 2002.
A.N. Sloss, D. Symes, C. Wright; ARM System Developer's Guide. Designing and Optimizing System Software, Morgan Kaufmann Series in Computer Architecture and Design, 2004.
J. Yiu: The Definite Guide to the ARM Cortex-M3 and Cortex-M4 Processors, Newnes Publishing, 2013.

Forms of work / resources: Seminar-like lecture with computer-aided examples as well as Lecture hall exercises, in the practical course the understanding of the material of the course is supported and deepened with the help of experiments.

Department: Computer Science

Expert group: Computer Engineering

Module responsibility: Thomas Horsch

Distributed systems

English Title: Distributed Systems

Document numbers: 30.7404 [PVL 30.7405]

Language: GERMAN

Assignment:
Bachelor 2021 - elective S_5/6 catalog - 4th semester
Bachelor dual KITS 2021 - 6th semester.
Bachelor dual KoSI 2021 - elective S_5/6 catalog - 6th semester
Bachelor KMI 2021 - 4th semester
Bachelor 2014 - 4th semester
Bachelor dual KITS 2014 - 6th
semester Bachelor dual KoSI 2014 -
6th semester Bachelor KMI 2014 - 4th
semester
Bachelor 2007 - 4th semester
KoSI 2007 - 6th semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (successful participation in the internship)

Course prerequisite: completed. SPOn 2021: The modules "Programming 2" and "Computer Networks" must be

be successfully completed.
SPO 2021 KMI: The modules "Programming 2" and "Multimedia Communication" must be successfully completed.
SPOn 2014: The modules "Programming, Algorithms and Data Structures 1" and "Programming, Algorithms and Data Structures 2" must have been successfully completed.

Required previous knowledge: Contents of the courses algorithms, operating systems, software Engineering and databases.

Learning Objectives:

Students will gain competencies in the development of network communications as well as the structure, architecture and basic algorithms from the field of distributed systems and will be able to apply them in a meaningful way.

- Students will master distributed systems fundamentals, be able to design, implement, and apply a distributed system infrastructure, understand and apply middleware for distributed systems, and design and implement simple distributed applications.
- The knowledge acquired can be used in the administration and development of distributed systems.

Course Content:

Characteristic properties of distributed systems

- Computer communication
- Basic technologies and design patterns for distributed processing
- Time in distributed systems, synchronization
- Distributed transactions and concurrency control
- Replication, consistency and fault tolerance in distributed systems
- Distributed file systems and name services
- Case studies middleware [e.g. Web Services, Message-oriented Middleware].

Literature:

Tanenbaum, Steen: Distributed Systems, Pearson Studium Publishing, 2nd ed. Edition, 2007

- Coulouris, Dollimore, Kindberg: Distributed Systems, Prentice Hall, 5th Edition, 2011.

Forms of work / resources:

Seminar lecture and practical course in a laboratory with heterogeneous System environment.
Resources: Lecture notes of the lecturers, exercise sheets and practical course material

Department: Computer Science

Division: Operating Systems / Distributed Systems

Module responsibility: Lars-Olof Burchard

Visual Computing

Document numbers: 30.7416 [PVL 30.7417]

Language: GERMAN

Assignment:

Bachelor 2021 - Elective S_5/6 Catalog - 4th semester.
Bachelor dual KoSI 2021 - elective S_5/6 catalog - 6th semester
Bachelor KMI 2021 - 4th semester
Bachelor 2014 - 4th semester

Teaching form: V+P = Lecture+Internship

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. practical course): ungraded(Successful participation in the practical course. Practical and theoretical tasks related to the contents of the course must be completed. Further details will be announced at the beginning of the course).

Prerequisite: Module "Programming 1" must be successfully completed.

Required prerequisite courses: Basic undergraduate level knowledge of linear algebra and.
technical basics of computer science

Learning Objectives:

Students will gain competencies to

- Understand how graphics systems, as well as image processing systems, work internally and be able to work with them,
- to master the basic features of graphic programming in order to be able to model and animate 2D and 3D scenes themselves, e.g. for demonstration and simulation purposes,
- to process digital image data [e.g. with regard to analyzability], to process them in a targeted manner [e.g. for computer vision applications] and to be able to save them in a targeted manner with regard to the respective further use,
- know current image generation and image output techniques [e.g. also 3D output],
- Know current rendering and visualization techniques and master the algorithms that are fundamental to them,
- understand the structure of digital images and color models and be able to assign them to the different application areas or questions,
- Know data formats of graphical data processing and understand the underlying compression methods,

- Master the mathematical fundamentals of graphical data processing.

Course Content: Introduction and overview of the entire field and related areas.

- Special features of graphical data
- Digital images, object and image space
- Color models
- Elementary image editing and processing
- Image compression and file formats
- Graphical objects and their creation, graphical programming
- Mathematical basics, geometric transformations
- Rendering techniques, visualization
- Acquisition and output of digital images, equipment technology

Literature: Hughes J.F. et al, "Computer Graphics Principles and Practice," Addison Wesley;

- Nischwitz A. et al, "Computer Graphics and Image Processing: Volume I: Computer Graphics: 1", Vieweg+Teubner;
- Nischwitz A. et al, "Computer Graphics and Image Processing: Volume II: Image Processing: 2", Vieweg+Teubner;
- Strutz T., "Image Data Compression", Vieweg+Teubner;
- Gortler S. J., "Founda0ons of 3D Computer Graphics", MIT Press.

Department: Computer Science

Professional group: Multimedia and graphics

Module responsibility: Elke Hergenröther

Scientific work

English title: Scientific work

Document number: 30.7418

Language: GERMAN

Assignment: Bachelor 2021 - 4th semester
Bachelor KMI 2021 - 4th semester

Teaching form: VÜ = lecture with integrated exercise

SWS: 4

CP: 5

Exam: Written exam

Prerequisite: The module "Mathematics 2" must be successfully completed.

RequiredPrevious knowledge: It is recommended tohave already gained an in-depth insight into the various topics of computer science. For this, the basic studies (the first three semesters) should have been successfully completed.

Learning objectives:Students acquire the competencies to apply different methods to deal with typical problems in computer science. They will be able to describe them using correct technical terms and definitions and give examples of their application. In addition, students can explain when and why a method can be applied to an existing problem. In addition, students know and understand the basic rules of scientific ethics and scientific work. They are able to explain, interpret and apply these to their work.

Students are able to analyze a typical problem in computer science and break it down into individual parts in order to understand and abstract the structure of the problem. They are able to describe the problem precisely in their own words and select suitable solution methods in a well-founded manner. Students are further able to apply methods of scientific work to various problems in computer science.

Course content:

Ethics of Science

- Methods to cope with typical problems in information technology
  - Literature research and identification of the state of the art
  - Formal or conceptual and argumentative-deductive analysis
  - Case studies
  - Prototyping
  - Laboratory and field experiments
  - Simulations
  - Reference modeling
  - Grounded Theory
  - Qualitative or quantitative cross-sectional analysis
- Formalities of the practical phase and the final thesis
- Scientific writing
  - Structure of a scientific Work
  - Working with sources/literature
  - Correct citation

Literature:

Raj Jain: The Art of Computer Systems Performance Analysis.

- Averill M. Law: Simulation, Modeling, and Analysis
- Prototyping-Oriented Software Development: Concepts and Tools
- Donald E. Knuth: The Art of Computer Programming
- Morten Hertzum: Usability Testing: A Practitioner's Guide to Evaluating the User Experience
- Balzert, H., Schröder, M. and Schäfer, C. Wissenschaftliches Arbeiten: Ethics, content & form of wiss. Work, tools, sources, project management, presentation

Department: Computer Science

Module responsibility: Dean of Studies

Project System Development

Module number: 30.7526

Language: english

Study programme: Bachelor 2021 - 5th
semester Bachelor 2014 -
5th semester
Bachelor KMI 2014 - 5th semester
Bachelor 2007 - 5th semester

Type of course: Pro = Project

Weekly hours: 4

Credit Points: 7.5

Exam: The project-specific criteria will be posted at the beginning of

eachproject.Prerequisite for booking: PO 2021: -
PO 2014: The modules "Programming / Algorithms & Data Structures 1" and "Programming / Algorithms & Data Structures 2" have to be passed successfully.

Required knowledge: Project-specific knowledge from mandatory courses of the first 4 semesters.

Learning objectives: The students can work on a specific question in one of the areas of Computer Science. They are proficient in a structured approach and can present their results in an adequate form.
They apply their so far acquired knowledge and expand and deepen

- their academic competences in at least one area of Computer Science,
- their competences in Software Engineering and Project Management,
- key competences, such as cooperation and team skills, presentation techniques and competences,
- strategies for knowledge acquisition.

Content: Regarding the content, the project group will work independently on current relevant questions. The content covers the deepening and application of the knowledge of at least one area of Computer Science, as well as the deepening and application of knowledge in Software Engineering and Project Management. At the end of the semester, all projects shall be presented in an adequate way, ideally open to the university.

Literature: Will be announced in each project.

Faculty: Computer Science

Responsibility: Dean of Students

Project Management

English Title: Project Management

Document number: 30.7506

Language: GERMAN

Assignment:
Bachelor 2021 - 5th semester
Bachelor dual KITS 2021 - 4th
semester Bachelor dual KoSI 2021 -
4th semester Bachelor KMI 2021 - 5th
semester
Bachelor 2014 - 4th semester
Bachelor dual KESS 2014 - 2nd
semester Bachelor dual KITS 2014 -
2nd semester Bachelor dual KoSI
2014 - 2nd semester Bachelor KMI
2014 - 2nd semester
Bachelor 2007 - 5th semester
KoSI 2007 - 4th semester

Teaching format: V = Lecture

SWS: 2

CP: 2.5

Exam: Written exam

RequiredPrior Knowledge: Prior knowledge from required courses in previous semestersLearning

Objectives: Students will gain the competencies of the following knowledge and.
Skills:

- Knowledge: The students know the general life cycle of projects as well as essential processes of project management and can explain them. The students can classify risk management as a permanent task and they know the basic principles of agile project management.
- Skills: Students will be able to use key planning documents in the create and deploy the course of projects, and they can use the

Document, analyze and control project progress. To do this, they can use basic techniques such as the earned value method.

- Competencies: Students will be able to work competently on projects -. in both classic and agile projects.

Course content:

According to the objective of the bachelor's degree program, academic professionals the focus of the learning objectives is on the operational fundamentals of project management. Aspects of personnel management are addressed, but not in depth.

- Project organization in the company (organizational structure, process organization)
- Necessary framework conditions for project initiation (budget, resources, deadlines, legal requirements)
- Involvement of service providers and consultants with a focus on service contracts, contracts for work and services, SLA as well as negotiation bases (focus also on employee leasing, bogus self-employment, liability, warranty)
- Project management, controlling and reporting during project execution
- Communication within the project, to the client and to the public
- Documentation (project file, operating concept)
- Risk management in the project, from problem identification to decision making to problem solving
- Special methods and procedures in project work such as cost/benefit analysis, earned value analysis, estimation procedures,
  Logical framework, milestone trend analysis, decision table technique.
- Moderation and presentation
- Dealing with resistance and conflicts
- Project completion, transfer to the line, post-calculation, lessons learned

Literature:

Frank Habermann, Karen Schmidt: Project Design: Thinking Tools for visually shaping new ventures. Becota GmbH, 2017. Electronic documents at https://overthefence.com.de/ (project canvas approach).

- Jenny, B. 2014. Project management: The knowledge for the professional, (3rd ed.). Zurich: vdf Hochschulverlag AG.
- Litke, H.-D., Kunow, I. and Schulz-Wimmer, H. 2015. project management, (2nd edition). Freiburg: Haufe-Lexware.
- Project Management Institute, A guide to the project management body of knowledge, 5th ed, Project Management Institute Publishing, 2012.
- Jörg Preußig: Agiles Projektmanagement, Freiburg: Haufe Verlag, 2018
- Peter Siwon: The human side of project success. Dpunkt Publishing House 2011
- Spitczok von Brisinski, N., Vollmer, G. and Weber-Schäfer, U. 2014. Pragmatic IT Project Management: Leading Software Development Projects Based on the PMBOK® Guide, (2nd ed.). Heidelberg: dpunkt-Verlag.
- Tiemeyer, E., Beims, M., Bergmann, R. and Ebert, C. 2018. handbook. IT project management: process models, management tools, good practices, (3rd ed.). Munich: Carl Hanser.
- Holger Timinger: Schnellkurs Projektmanagement, Wiley Verlag 2015.

Forms of work / resources:

Seminar-like lecture with intensive involvement of the students to practice the most important project management techniques. Applicable electronic aids (e.g. spreadsheets, protocol forms, etc.) are provided and used. Accompanying the lecture is a case study of project structure and execution.

Department: Computer Science

Professional group: Information Systems

Module responsibility: Urs Andelfinger

Project System Development

English Title: Project System Development

Document number: 30.7504

Language: GERMAN

Assignment: Bachelor 2021 - 5th semester
Bachelor KMI 2021 - 5th semester
Bachelor 2014 - 5th semester
Bachelor KMI 2014 - 5th semester
Bachelor 2007 - 5th semester

Teaching Form: Pro = Project

SWS: 4

CP: 7.5

Examination: The project-specific evaluation criteria will be announced at the beginning of the respective course.

Document Requirement: SPOn 2021: -
SPOn 2014: The modules "Programming, Algorithms and Data Structures 1" and "Programming, Algorithms and Data Structures 2" must have been successfully completed.

Required prerequisite courses: Project-specific prerequisite courses from required courses in the first four semesters.

Learning objectives: The students acquire the competences to work on a problem in a subarea of computer science in a project team. They master a structured approach and can present their results in a suitable form.
They apply the knowledge they have acquired up to that point and expand and deepen

- their professional competencies in at least one subfield of computer science,
- their competencies in software engineering and project management,
- Key competencies such as cooperation and teamwork skills, presentation and moderation skills,
- Knowledge acquisition strategies

Course content:In terms of content, the project group works independently on current issues relevant to practice. The subject matter includes, among other things, the deepening and application of knowledge in at least one sub-field of computer science as well as the deepening and application of knowledge in software engineering and project management. At the end of the semester, all projects are to be presented in a suitable form, preferably to a university public.

Literature: Will be announced in the respective course.

Department: Computer Science

Module responsibility: Dean of Studies

Bachelor module

English Title: Bachelor Module

Document numbers: 30.8920 [Bachelor thesis 30.8900; colloquium 30.8910]

Language: GERMAN

Assignment: Bachelor 2021 - 6th semester

Bachelor 2014 - 6th semester

Teaching Form: Pro = Project

CP: 15

Examination: Written paper (75%) and presentation (25%).

Document Requirement: SPO 2021: The following admission requirements must be met for admission to the bachelor's module:
a. the first study section (the first three semesters with 90 CP) has been successfully completed
b. the module "Scientific Work in Computer Science" has been successfully completed
c. the practical module has been successfully completed
d. all other compulsory modules of semesters 4 and 5 have been successfully completed except for one module
e. at least one examination attempt has been completed in the compulsory module that has not yet been passed

SPO 2014: Admission to the Bachelor module is granted if the candidate has passed all compulsory modules from the first and second stage of studies with the exception of 10 CP. At least one examination attempt must be completed in each of the compulsory modules not yet passed with a maximum of 10 CP.

Learning Objectives: The student is able to work independently on a problem of the subject, which may be related to the practical phase, within a given period of time using scientific methods and knowledge of the subject. This includes structuring the task, compiling the necessary resources and working on the basis of a time schedule.
Upon completion of the module, students will be familiar with the basic concepts and methods of scientific work as they apply to the preparation of the bachelor's thesis.

Department: Computer Science

Module responsibility: Dean of Studies

Bachelor thesis with colloquium

English Title: Bachelor Thesis with Defense

Document numbers: 30.89100 [bachelor's thesis 30.8900; bachelor's thesis AE 30.8901; bachelor's thesis TI. 30.8902; Bachelor's thesis TK 30.8903; Bachelor's thesis WI 30.8904; Colloquium on the Bachelor thesis 30.8910]

Language: GERMAN

Assignment: Bachelor 2007 - 6th semester

Teaching Form: Pro = Project

CP: 15

Examination: Colloquium, single weighted PVL

(e.g. internship): Bachelor thesis,triple weighted

Record Prerequisite: Successfully completed preparatory seminar 30.7510, practical phase 30.7600 begun.

Course Content: This is the final module. It consists of the bachelor thesis and the associated colloquium. The bachelor thesis can be assigned to a specialization

	in which case special document numbers are used, see above.
Department:	Computer Science

Practice module

English Title:	Practice Module
Document number:	30.7608
Language:	GERMAN
Assignment:	Bachelor 2021 - 6th semester Bachelor KMI 2021 - 6th semester Bachelor 2014 - 6th semester Bachelor KMI 2014 - 6th semester
Teaching Form:	Pro = Project
SWS:	1
CP:	15
Examination:	Written paper and presentation (successfully attended)
Prerequisite:	SPOn 2021: Admission to the practical module will be granted
by the	
	Examination Committee if the practical project meets the requirements of the module description, the module "Scientific Work in Computer Science" has been successfully completed and proof has been provided that all compulsory modules from the first three semesters have been successfully passed. SPOn 2014: Admission is granted by the examination board if the topic meets the requirements of the module description and the module 30.7512 "Scientific Work in Computer Science 2" or 81.7522 "Communication and Media" in the variant Communication and Media in Computer Science has been successfully completed.

Learning Objectives:The goal of the practicum phase is for students to become familiar with the duties of a computer scientist through their own hands-on, engineering-related activities. These include:

- Necessary framework for project initiation (budget, resources, deadlines, legal requirements)
- Providing an overview of the technical, organizational and economic interrelationships of the company and its social structures
- Acquisition of personal experience in a professional field characterized by technical, organizational and economic issues and the typical work processes and interrelationships there.
- Deepening knowledge of contemporary working methods for solving tasks (e.g. project management, team and group work, moderation)

The practical phase is intended to enable the application of the knowledge and skills acquired so far in the course of study. The practical module serves to deepen the professional competence in at least one subfield of computer science.

In addition, key competencies such as cooperation and teamwork skills, presentation and moderation skills, and knowledge acquisition strategies are practiced and deepened. Likewise, through the organization of the project in a team, general transfer and social skills (rhetoric, conflict management) are trained in a practical manner, preparing the students for later industrial professional practice.

Department: Computer Science

Module responsibility: Dean of Studies

Advanced Systems Programming

Module numbers: 30.2650 [PVL 30.2651; Module 30.26500]

Language: english

Study programme:
Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): graded[Successful participation in the practical: The preliminary examination has been completed if the graded practical assignments - exercises and a project in small groups - have been passed with a grade of 4.0 or better]).

PVL percentage: 50%

Required knowledge:
Experience with modern C++ development (C++17)
- Experience writing native software under Linux

Learningobjectives: Thestudents are able to understand, design and implement hardware-efficient systems software. Students will learn the fundamentals of a modern systems programming language (Rust) and how it compares to the widely used systems programming language C++. Students will understand how to balance performance, safety and maintainability while writing systems software. By focusing on two different systems programming languages, good programming skills and a deep understanding of common systems programming concepts are encouraged.

Content:
What is Systems Programming and how does it compare to Application Programming?
- Zero-overhead abstractions in C++ and Rust and how they help to write fast, readable and maintainable code
- The fundamentals of memory management and memory safety
- Error handling concepts in C++ and Rust for writing robust systems software
- System level I/O and Network Programming
- Fearless concurrency
- Profiling and tracing: How to measure, evaluate and tweak performance
- Tools for developing, debugging and maintaining systems software

Students will gain extensive hands-on experience in systems programming by analyzing open-source code and developing their own systems in the lab.

Literature:
Computer systems: a programmer's perspective. Vol.3 - Bryant, R. E., David Richard, O. H.
- The Rust Programming Language - Steve Klabnik, Carol Nichols
- A Tour of C++ (2nd Edition) - Bjarne Stroustrup

Faculty: Computer Science

Expert group: Programming

Responsibility:	Stefan Rapp

Compiler Construction

Module numbers:	30.2548 [PVL 30.2549]
Language:	english
Study programme:	Bachelor 2021 - Elective catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization AE: Application Engineering Bachelor 2007 - Specialization TI: Computer Engineering Bachelor 2007/2004/2002/99 - Elective courses from the computer science area KoSI 2007 - Specialization AE: Application Engineering KoSI 2007 - Specialization TI: Computer Engineering KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Type of course:	V+P = Lecture+Practical
Weekly hours:	3+1
Credit Points:	5
Exam:	written exam
PVL (e.g. Practical):	not graded (Successful participation in the laboratory.)
Requiredknowledge:	Basic,bachelor-level programming skills and fundamental, bachelor-level knowledge of theoretical computer science.
Learningobjectives:	Aftercompleting the course, students should be able to understand and apply all the phases of compilation in order to translate a program in source code into an executable form. Further, they should be able to apply the same techniques to solve commonly occurring cross-compilation (format conversion) tasks.
Content:	The course covers both the theory and practice of compiler construction. Compiler theory is reviewed, and then applied. Topics: • Context Free Languages • Lexical Analysis • Syntax Analysis and Parsing • Error handling • Code Generation • Code Optimization Tools such as Lex and Yacc (Flex and Bison) and LLVM are covered in the lecture and used in the lab.
Literature:	Aho, Lam, Sethi, Ullman: Compiler - , Compilers: Principles, Techniques, and Tools , 2nd Edition, Addison Wesley, 2007.
Lecture style / Teaching aids:	Lecturewith Laboratory. Resources include lecture note, example programs and software tools.
Faculty:	Computer Science
Expert group:	Operating systems / distributed systems

Responsibility: Ronald Moore

Data Warehouse Technologies

English Title: Data Warehouse Technologies

Document numbers: 30.2512 [PVL 30.2513]

Language: GERMAN

Assignment:
CNAM - Preparation cycle
CNAM Master - Preparatory Cycle
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization AE: Application Engineering
Bachelor 2007 - Specialization WI: Business Informatics
Bachelor 2007/2004/2002/99 - Elective courses from the computer science
area KoSI 2007 - Specialization AE: Application Engineering
KoSI 2007 - Specialization WI: Business Informatics
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded
Tests, homework and/or written assignments, or successful participation in the internship; to be announced at the beginning of the
Course announced.

Required Prior Knowledge: Basic bachelor's level knowledge of databases and
Business Informatics

Learning Objectives: Students will

- Know and be able to evaluate the phases of data warehousing and the reference architecture of a data warehouse,
- Be familiar with the multidimensional data model, associated analysis operations, and conceptual modeling notations and be able to apply them using a modeling tool,
- master the relational storage (Star, Snowflake schema) of the multidimensional data model,
- Be familiar with the Extraction - Transformation - Loading (ETL) process in data warehousing,
- know internal data structure concepts of data warehouses,
- Be familiar with and able to apply multidimensional query processing,
- know the extension of the relational database language SQL in the field of data warehousing and be able to apply it practically,
- Know and be able to use a modern business intelligence tool.

Course content: Data Warehouse Architecture

- Database techniques for building and implementing data warehouses
- Multidimensional data modeling

- Extraction, Transformation, Loading (ETL)
- Internal storage structures for data warehouses
- Queries, query processing and query optimization in data warehouses
- Application areas for data warehouses

Literature: Köppen, V; Saake, G.; Sattler, K.-U.: Data Warehouse Technologien, 1st edition, mitp-Verlag, 2012.

- W. Lehner: Datenbanktechnologie für Data-Warehouse-Systeme, 1st edition, dpunkt.verlag, 2003
- A. Bauer, H. Günzel: Data Warehouse Systems - Architecture, Development, Application, 4th edition, dpunkt.verlag, 2013
- W.H. Inmon: Building the Data Warehouse, 4th edition, Wiley, 2005.

Work forms / resources: seminar lecture, electronically available materials,
Lecture hall exercises, practical courses on the computer

Department: Computer Science

Professional group: Databases

Module responsibility: Stephan Karczewski

DevOps Engineering with Kubernetes

Document numbers: 30.2656 [PVL 30.2657]

Language: GERMAN

Assignment: Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. practical course): ungraded (practical work with and on K8s).

Required previous knowledge: Knowledge from the modules "Software Engineering" and "Operating Systems" Knowledge of Linux and its CLI tools
Recommended: Knowledge from the module "Distributed Systems

Learning Objectives:Students will receive a basic and practical introduction to the concepts and techniques of DevOps Engineering and Kubernetes. These include release patterns, feedback, resilience patterns, and knowledge of Kubernetes architecture, objects, helm, etc. Students will understand these and be able to explain and apply them.
In addition, the students know the advantages and disadvantages of using DevOps practices and Kubernetes and the associated risks. The acquired knowledge can be applied by the students in the administration and development of modern containerized IT infrastructures or for the operation of software in such IT infrastructures.

Course Content: General introduction to the topic of DevOps engineering and Kubernetes (K8s):

- Definitions and basic terms and illustrative examples from the

Everyday life especially with regard to the importance in software engineering

- DevOps Principles and Techniques in Software Engineering (Automated Testing, Deployment Pipelines, Release Patterns, CI/CD, Feedback Mechanisms, Telemetry, Organizational Learning, Safety Culture, Resiliance Patterns)
- Introduction to the basics of Kubernetes
- K8s objects (Pods, ReplicaSets, ConfigMaps, etc.)
- Management of application on K8s (e.g. helmet)
- K8s Service Mesh (e.g. Istio)
- Practical exercises with Kubernetes
- Safety aspects in handling K8s
- Automation in modern deployment environments
- Monitoring & Logging in modern deployment environments

Literature:

Kim, Gene, Jez Humble, Patrick Debois, John Willis, and Nicole Forsgren. The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations. IT Revolution, 2021.

- Humble, Jez, and Gene Kim. Accelerate: the science of lean software and DevOps: building and scaling high performing technology organizations. IT Revolution, 2018.
- Burns, Brendan, Joe Beda, and Kelsey Hightower. Kubernetes: up and running: dive into the future of infrastructure. O'Reilly Media, 2019.
- Newman, Sam. Building microservices. "O'Reilly Media, Inc., 2021.
- Richardson, Chris. Microservices patterns: with examples in Java. Simon and Schuster, 2018.

Department: Computer Science

Professional group: Software Engineering

Module responsibility: Stefan T. Ruehl

Digital transformation

English Title: Digital Transformation

Document numbers: 30.2618 [PVL 30.2619; Module 30.26180]

Language: GERMAN

Assignment:

Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching Form: V+S = Lecture+Seminar

SWS: 3+1

CP: 5

Exam: Written exam

PVL (e.g. practical course): graded (graded presentation on a given topic) Proportion of

PVL: 30%.

Learning Objectives: Students will

- can describe the drivers and central characteristics of digitization and use examples to explain the impact of digitization on society, organizations and the individual.

- can describe the key IT innovations of digitization and their characteristics and identify application scenarios for these innovations in an organizational context.
- can explain which strategic issues should be addressed in the course of digitization and how a digital transformation strategy can be anchored and delimited organizationally.
- can explain the procedure for developing a digital transformation strategy and illustrate it with examples.
- can explain how business models are changing in the course of digitization and how this affects the form of value creation as well as products and services.
- can apply the concepts to analyze and design business models.
- can deduce which structural changes digitization requires in companies and what role business process management plays in this.
- can understand the effects of the digital transformation on the Explain IT management and show examples of current developments in this area
- are able to independently answer a specific question from the context of the digital transformation, prepare and present it in a way that is suitable for the recipient.

Course Content:

Basic terms and concepts: Digitization and digital transformation

- Digital transformation: trends, technologies & framework conditions
- Digital transformation strategies
- Business models and business model innovation
- Transformation of services, processes and organization
- Implications for IT management

Literature:

The following list represents a selection of relevant literature on the subject. Additional literature will be announced during the course.

- Gassmann, O., Frankenberger, K. and Csik, M. 2013. Developing business models: 55 innovative concepts with the St. Gallen Business Model Navigator. Munich: Hanser.
- Hess, T., Matt, C., Benlian, A. and Wiesböck, F. 2016. "Options for Formulating a Digital Transformation Strategy," MIS Quarterly Executive (15:2), pp. 123-139.
- Lemke, C. and Brenner, W. 2015. introduction to information systems. Berlin: Springer.
- Osterwalder, A. and Pigneur, Y. 2010. Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. New York: John Wiley & Sons.
- Schallmo, D. and Rusnjak, A. 2017. "Roadmap to Digital Transformation of Business Models," in Digital Transformation of Business Models: Fundamentals, Tools, and Best Practices, D. Schallmo, A. Rusnjak, J. Anzengruber, T. Werani, and M. Jünger (eds.). Wiesbaden: Springer, pp. 1-31.
- Urbach, N. and Ahlemann, F. 2016. IT-Management im Zeitalter der Digitalisierung: Auf dem Weg zur IT-Organisation der Zukunft. Berlin: Springer Gabler.
- Ward, J. and Peppard, J. 2002. strategic planning for information systems, (3rd ed.). New York: John Wiley & Sons, Inc.
- Weill, P. and Ross, J.W. 2009. IT Savvy What Top Executives Must Know to Go from Pain to Gain. Boston, Mass.: Harvard Business Press.

Forms of work / resources: Seminar lecture

Department: Computer Science

Professional group: Information Systems

Module responsibility: Daniel Burda

Effective Agile Software Projects in Industrial Practice

EnglishTitle: EffectiveAgile Software Projects in Industrial

Practice Document Number: 30.2668

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: VP = Lecture with integrated practical course

SWS: 4

CP: 5

Exam: Written exam

Required Prior Knowledge:
Basic bachelor's level knowledge of programming, object-oriented analysis and design and software engineering, and ideally in project management.
Previous successful completion of the modules "Programming1 and 2" and "Object-oriented Analysis and Design" is required.
Previous attendance of the module "Software Engineering" is strongly recommended. Previous or parallel enrollment of the module "Project Management" is recommended.

Learning Objectives:
Solid understanding on the application of software engineering methodologies in industrial practice with emphasis in agility, productivity, efficiency and effectiveness.
Competence to select, adapt if necessary, and independently apply the most important industrially proven best practices of the task areas of the following domains in software engineering listed under course content:

- Procedure strategy
- Requirements and analysis
- Architecture and design
- Development
- Test
- Infrastructure and operation
- Project Management

Course Content:
Best practices from the following task areas:

- Process strategy: define basic strategy, analyze project context, define PM method, define form of execution, understand project content, design collaboration
- Requirements & Analysis: Communicate Business, Understand Requirements, Manage Requirements, Specify Business, Manage Business Environment
- Architecture & Design: Understanding architecture as a discipline, providing orientation and establishing guard rails, considering requirements, designing solution architecture, documenting and communicating architecture, reviewing and evaluating architecture, countering architecture risks.
- Development: understand development as a discipline, create fine design, code

write, ensure code quality, document code

- Test: Define test strategy, Plan and prepare test, Manual test, Automated test, Structure, document and evaluate test, Ensure test quality
- Infrastructure & Operations: Integrate development and operations, provide development environment, provide operations infrastructure, automate operations, run software
- Project Management: Understanding Project Management as a Discipline, Planning and Calculating Projects, Monitoring Projects, Controlling Scope, Controlling Risks, Controlling Environment, Communicating in Projects, Leading in Projects

Literature:

The course is "Self Contained", i.e. completely understandable without additional literature. The extensive literature listed below is "Secondary Reading" and can be studied optionally for more depth.

Requirements and Analysis:

- Dean Leffingwell: Agile Software Requirements. Addison-Wesley, 2011
- International Institute of Business Analysis: Agile Extension to the BABOK® Guide. IIBA and Agile Alliance, 2013
- Peter Hruschka: Business Analysis and Requirements Engineering: Improving Products and Processes Sustainably. Hanser Verlag, 2014
- Chris Rupp: Requirements Engineering and Management: From Classic to Agile Practice. Hanser Verlag, 2014
- International Institute of Business Analysis: Guide to the Business Analysis Body of Knowledge® - BABOK® Guide 3.0. IIBA, 2015.
- Jeff Patton: User Story Mapping: the technique for better user understanding in agile product development. O'Reilly, 2015
- Johannes Bergsmann: Requirements Engineering for Agile Software Development: Methods, Techniques and Strategies. dpunkt.Verlag, 2nd edition, 2018.

Architecture and design:

- Martin Fowler: Patterns of Enterprise Application Architecture. Eddison Wesley, 2002
- Michael T. Nygard: Release It!: Design and Deploy Production-Ready Software. O'Reilly, 2018
- Stefan Toth: Process Patterns for Software Architecture: Combinable Practices in Times of Agile and Lean. Hanser Verlag, 3rd edition, 2019
- Gernot Starke: Effective Software Architectures - A Practical Guide. Hanser Verlag, 9th edition, 2020
- Stefan Zörner: Documenting and Communicating Software Architectures: Recording Designs, Decisions and Solutions in a Comprehensible and Effective Way. Hanser Verlag, 3rd edition, 2021

Development:

- Robert C. Martin: Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall, 2008
- Robert C. Martin: Agile Software Development, Principles, Patterns, and Practices. Pearson, 2013
- David Thomas, Andrew Hunt: The Pragmatic Programmer: Journey to Mastery. 20th Anniversary Edition, Addison Wesley, 2019.

Test:

- Andreas Spillner, Tilo Linz: Basiswissen Softwaretest. Training and further education to become a Certified Tester - Foundation Level according to the ISTQB® standard. dpunkt.Verlag, 6th edition, 2019.
- Lisa Crispin, Janet Gregory: Agile Testing: A Practical Guide for Testers and Agile Teams. Addison-Wesley, 2008
- Lisa Crispin, Janet Gregory: More Agile Testing: Learning Journeys for the Whole Team. Addison-Wesley, 2014

Environment and operation:

- Jez Humble, David Farley: Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Addison-Wesley, 2010
- John Gilbert: Cloud Native Development Patterns and Best Practices: Practical architectural patterns for building modern, distributed cloud-native systems. Pact Publishing, 2018
- Nane Kratzke: Cloud-Native Computing. Software engineering of services and applications for the cloud. Hanser Verlag, 2021
- Gene Kim, Jez Humble, Patrick Debois, John Willis: The Devops Handbook: How to Create World-class Agility, Reliability, & Security in Technology Organizations. IT Revolution Press, 2021

Procedure strategy and project management:

- Tom DeMarco: The Deadline. A novel about project management. Hanser Verlag, 1998
- Hans-Jürgen Plewan, Benjamin Poensgen: Productive Software Development: Assessing and Improving Productivity and Quality in Practice. dpunkt.Verlag, 2011
- Jeff Sutherland: Scrum: The Art of Doing Twice the Work in Half the Time. Random House, 2015
- Malte Foegen, Christian Kaczmarek: Organization in a Digital Era: A book for designing responsive and lean organizations using Scaled Agile & Lean patterns. Wibas GmbH, 2016
- Gene Kim, Kevin Behr, George Spafford: The Phoenix Project: A Novel About IT, DevOps, And Helping Your Business Win. IT Revolution Press, 2018
- Ursula Kusay-Merkle: Agile project management in everyday work: for medium and small projects. Springer Gabler, 2018

Forms of work / aids:	Inverted Classroom, i.e. teaching content as videos and corresponding slides. (80-100 separate knowledge building blocks), plenary sessions for consolidation and interaction as well as for the internship components.
Department:	Computer Science
Professional group:	Software Engineering
Module responsibility:	Markus Voß

Introduction to Artificial Intelligence

English Title:	Introduction to Artificial Intelligence
Document numbers:	30.2662 [PVL 30.2663]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective S_5/6-Catalog Bachelor dual KITS 2021 - Semester 6. Bachelor dual KoSI 2021 - Elective S_5/6 Catalog Bachelor KMI 2021 - 4th semester
Teaching form:	V+P = Lecture+Internship
SWS:	3+1
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded(Successful participation in the internship. The preliminary examination is ungraded. Passing the PVL is a prerequisite for admission to the examination).
Prerequisite:	The modules "Mathematics 1", "Mathematics 2", "Programming 2" must be attended. be successfully completed.

Learning Objectives:

Students will

  - know the different subfields of artificial intelligence and their respective basic approaches and strategies
  - Understand how AI applications are constructed in principle
  - know the basic procedures for each of these areas

- The students
  - are able to apply appropriate technologies to given problems in order to solve non-trivial problems
  - Can assess where AI solutions are appropriate

- The students
  - can adapt procedures in order to develop and implement proposed solutions
  - can develop a critical view of developments in AI against the background of philosophical foundations and ethical issues, as well as recognize and assess risks and possible technological consequences of the development of systems with AI technologies

Course Content:

This course provides an overview of the fields of AI with references to more in-depth courses. The following contents are covered:

- Machine learning (ML): Basic ML techniques based on prominent Examples such as artificial neural networks or decision trees; metrics/evaluation techniques to measure the goodness of ML predictions. Relation to symbolic and non-symbolic AI.
- Knowledge representation and processing: basic procedures, e.g. Ontologies and Linked Data; query languages and reasoning. Relation to symbolic and non-symbolic AI.
- Natural language processing (NLP): Application areas of NLP such as e.g. document classification, machine translation or human-machine communication, as well as current technologies for easy implementation of the same; reference to symbolic and non-symbolic AI.
- Computer vision: application areas such as object recognition on images, as well as current technologies for implementing the same; relation to non-symbolic AI.
- Cross-cutting topics: Philosophical foundations and ethical issues of AI; opportunities and risks of autonomous systems; bias in AI applications; impact of AI applications on society and working life.

All contents are practiced in the internship.

Literature:

Bernhard G Humm: Applied Artificial Intelligence - An Engineering Approach. Second Edition. Leanpub, Victoria, British Columbia, Canada, 2016. leanpub.com/AAI

- Russel, S. / Norvig, P. Artificial Intelligence: A Modern Approach (Pearson Series in Artificial Intelligence), 4th ed, 2020.

Further reading:

- Christopher M. Bishop. 2006. pattern recognition and machine learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.
- Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani: An Introduction to Statistical Learning. New York, NY, USA : Springer New York Inc., 2001 (Springer Series in Statistics, vol. 103).
- Ian Goodfellow, Yoshua Bengio and Aaron Courville "Deep Learning", MIT Press 2016.
- Jurafsky, Daniel / Martin, James. 2014. speech and language processing. An Introduction to Natural Language Processing, 2nd ed. Pearson India.

Department:

Computer Science

Division: Artificial Intelligence

Module responsibility: Gunter Grieser

Introduction to mobile communication

English Title: Introduction to mobile communication

Document numbers: 30.2542 [PVL 30.2543]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007/2004/2002/99 - Electives from the computer science area KoSI 2007/2004/2003/2002/99 - Electives from the computer science area

Teaching Form: V+S+P = Lecture+Seminar+Internship.

SWS: 2+1+1

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded
Written elaboration as well as regular and successful participation in the internship and seminar.

Required prior knowledge: Basics of networks, OSI model, Internet protocol

Learning Objectives: After attending this course, students will be able to

- Independently perform analyses on the most common current cellular systems,
- to be able to compare and evaluate competing systems and mobile communications solutions,
- and to assess their capabilities (such as performance and security).

Students receive a comprehensive introduction to the field of mobile communications from a computer science perspective. This includes the structure and functionality of mobile networks, as well as the identification of possible new mobile services and applications. The development of mobile networks from the beginnings of WLAN and GSM networks to GPRS, UMTS and current mobile technologies will be shown and compared.

Knowledge and skills acquired in this module are also fundamental for planning and operating radio networks. This course also looks specifically at local wireless networks (WLAN), personal area wireless networks (WPAN) and campus-wide (regional) wireless networks (WMAN). Furthermore, the knowledge imparted forms important system foundations for the development of embedded systems or mobile applications.

Specifically, students should

- Gain basic knowledge of the transmission characteristics of mobile radio channels and be able to explain them

- Know, understand, and be able to apply basic procedures for planning as well as operating simple personal, local, and campus-wide wireless networks.
- Know, understand and be able to explain the basic features of the protocols on the radio interface (media access, provision of differentiated quality of service classes, mobility support and access security)
- Know, understand, design, and be able to explain basic system architecture for various application scenarios and the associated protocols.
- Know, understand, critically analyze and evaluate the basic features of security procedures for radio networks.
- Know standardized wireless network technologies (such as IEEE 802.11 (WLAN), 802.15 (WPAN), 802.16 (WMAN)) and be able to compare them

Course content:

Fundamentals of radio transmission (e.g., radio spectrum, signals, antennas, signal propagation, multiplexing, multiple access, modulation, spreading techniques, coding).

- Basic media access procedures
- Infrastructure Networks, Adhoc Networks, and Meshed Networks [optional].
- Local Area Networks, IEEE 802.11 (architecture, radio protocols, quality of service, mobility, security, radio and network planning) (WIFI)
- Personal area radio networks, IEEE 802.15, (such as Bluetooth and ZigBee)
- Campus-wide/regional wireless networks, IEEE 802.16 (WIMAX) [optional].
- Mobility support in the network layer (Mobile IP)
- Routing in mobile ad hoc networks [optional]
- Introduction and overview of cellular networks (such as GSM, GPRS, UMTS, HSPA, LTE and other current topics)
- Overview of integration concepts (integration of radio interfaces (Seamless Mobility), integration of multimedia services (IP Multimedia System), service provisioning)
- Overview of other wireless technologies (such as RFID, NFC, and sensor networks) [optional].

Literature:

J. Schiller, "Mobile Communications," Pearson Studies, 2003 (or higher).

- Martin Sauter, "Grundkurs Mobile Kommunikationssysteme - UMTS, HSPA und LTE, GSM, GPRS, Wireless LAN und Bluetooth", Springer Verlag, 5th edition 2013 (or higher).
- Ralf Ackermann and Hans Peter Dittler, "IP-Telefonie mit Asterisk", Edition 2007 (or higher), dpunkt-Verlag Heidelberg
- Jörg Roth, "Mobile Computing", 2nd edition 2005 (or higher), dpunkt-Verlag Heidelberg
- Further current literature will be announced in the course
- Lecturer script

Work forms / resources: Seminar-style lecture, problem-oriented learning (POL), small groups and teamwork, practical, project work, script, supplementary examples, mock exams, exercise sheets, worksheets, case studies and homework assignments

Department: Computer Science

Professional group: Telecommunications

Module responsibility: Michael Massoth

Introduction to the technology and application of RFID

English Title: Introduction to RFID

Document numbers: 30.2334 [PVL 30.2335]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KESS 2014 - Catalog ESS: Embedded Systems
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization TI: Computer Engineering
Bachelor 2007 - Specialization TC: Telecommunications
Bachelor 2007/2004/2002/99 - Elective courses in computer science KoSI
2007 - Specialization TI: Computer Engineering
KoSI 2007 - Specialization TC: Telecommunications
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (certified participation in the exercises of the internship RFID)

Requiredprior knowledge: Basic knowledge in the fields of computer engineering, the Programming and the algorithms and data structures.

Learning Objectives: The course teaches the underlying techniques for applications in logistics, enterprise resource planning and optimization of business processes:
In addition to one- and two-dimensional barcodes, RFID (Radio Frequency IDentification) technologies will play a prominent role in identification in the future.
It will be introduced to the valid standards of RFID, including the physical conditions such as range and biological effect.
Based on the conditions of real applications, models of business processes are translated into the designs of an IT infrastructure.
Optimization of business processes and consumer and data protection are further focal points.
In the practical course, the fundamentals of some standards are experienced as well as small independent applications are realized with the help of independently developed software. The skills to be achieved by the students are divided into categories such as:

- Analysis competence for the assessment of requirements in the area of Business processes and logistics
- Be able to translate requirements from these areas into an IT structure, technical design and algorithms
- Technological competence RFID

Course content: Introduction to automatic identification systems (barcode, chip cards, biometric procedures), history of RFID

- Technical basics such as frequency, range, coupling and antennas
- basic functionality and designs of RFID tags
- Application and integration in business processes
- RFID infrastructure, IT architecture and services

	• Security, cryptography and data protection • Examples from practice
Literature:	Finkenzeller; RFID Handbook; Hanser; ISBN 3-446-40398-1 • Gillert, Hansen; RFID für die Optimierung von Geschäftsprozessen; Hanser; ISBN 3-446-40507-0; • Script
Work forms / resources:	Seminar lecture and practical course, script, supplementary examples from practice, exam preparation
Department:	Computer Science
Expert group:	Computer Engineering
Module responsibility:	Ralf S. Mayer

Introduction to Software Defined Radio

English Title:	Introduction to Software Defined Radio
Document numbers:	30.2570 [PVL 30.2571]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization TI: Computer Engineering Bachelor 2007 - Specialization TC: Telecommunications Bachelor 2007/2004/2002/99 - Elective courses in computer science KoSI 2007 - Specialization TI: Computer Engineering KoSI 2007 - Specialization TC: Telecommunications KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded (successful participation in the internship)
Required Prior Knowledge:	Basic knowledge in the areas of computer engineering, the Programming and algorithms and data structures, mathematics (analysis and linear algebra)
Learning Objectives:	Students will • Know the basic hardware structure and understand the algorithmic principles for software-based processing of complex signals. • The acquired knowledge should be applicable to simple systems such as amplitude modulated radio services • be able to name and describe the fields of application of software-based FFT methods
Course content:	Basic hardware design of SDR-based systems. • Real signals, aliasing, mixing signals, bandwidth • Complex numbers, complex signals

- Discrete Fourier Transform and FFT
- Digital filters
- Modulation method

Literature: Foundations of Signal Processing, Martin Vetterli, Jelena Kovacevic, Vivek K Goyal, Cambridge University Press, 2014.
The Scientist and Engineer's Guide to Digital Signal Processing, Steven W. Smith, Second Edition, California Technical Publishing , 1999

Forms of work / resources: Seminar lecture and lecture hall exercises, script, examples from the Practice. In the practical course, the understanding of the material of the course is deepened with simple experiments and own developments.

Department: Computer Science

Expert group: Computer Engineering

Module responsibility: Ralf S. Mayer

Enterprise Information Systems

Module numbers: 30.2658 [PVL 30.2659]

Language: english

Study programme: Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded (Successful completion of an ungraded assignment.)

Required knowledge: Basic knowledge of Business Administration and Management Information Systems

- Basic knowledge of Business Process Management
- Ideally domain knowledge from at least one common business area such as sales, accounting, materials management, procurement or HR.

Learning objectives: Students

- can explain the basic concepts of Enterprise Information Systems (EIS)
- can describe the concepts of integrated value chains, integrated Information Systems (IS) and business process automation
- can provide an overview of the current EIS market, product categories (ERP, CRM, SCM etc.), architecture, delivery models, current trends and challenges
- can explain underlying activities of general core business processes such as sales, HR, procurement, financial accounting, controlling etc.
- are able to contribute in projects related to the development, implementation and maintenance of EIS
- are able to execute core business processes using EIS

Content: This module provides an overview of pivotal concepts of EIS and demonstrates how common core business processes can be automated by using such systems. The module advances the theoretical understanding of EIS and provides students with

with the opportunity to apply those concepts on a real-life EIS (e.g., SAP) based on practical exercises and case studies.

The following topics are covered:

- Fundamentals of Enterprise Information Systems, Supply Chain Management Systems, Customer Relationship Management Systems
- Concepts of information integration
- Introduction to core business processes such as sales, HR, procurement, financial accounting, controlling etc. and avenues for automating such processes
- Development, implementation and maintenance of EIS

Literature:

Bradford, M. 2015. Modern ERP: Select, implement, & use today's advanced business systems, (3rd ed.). Middletown: Lulu.

- Gronwald, K.-D. 2020. Integrated Business Information Systems: A Holistic View of the Linked Business Process Chain - ERP-SCM-CRM-BI-Big Data, (2nd ed.). Berlin, Heidelberg: Springer.
- Laudon, K.C., and Laudon, J.P. 2022. Management Information Systems: Managing the Digital Firm. Harlow, Essex: Pearson Education Limited.

Lecture style / Teaching aids:

System demonstrations

- Lectures
- Practical case studies and exercises to be conducted on a real-life EIS (e.g., SAP ERP)
- Moodle activities such as quizzes or tests

Faculty: Computer Science

Expert group: Business Informatics

Responsibility: Daniel Burda

Decision Theory

English title: Decision theory

Document numbers: 30.2612 [PVL 30.2613]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+Ü = Lecture+Exercise

SWS: 2+2

CP: 5

Exam:Written exam (Written exam)

PVL (e.g. practical course): ungraded (processing of exercise, development or design tasks).

Learning objectives:

When individuals, groups or organizations engage in economic activity, decisions are practically always involved. Decision theory deals with the goal-oriented actions of decision-makers who have the freedom to choose from a number of alternative decision options. On the one hand, the course Decision Theory wants to offer well-studied tools for making economic decisions "rationally".

and "better" (normative), and, on the other hand, also wants to explain how real decisions are made in operational practice (descriptive).
Students should:

- recognize, understand and critically discuss how economic decision-making processes are rational (or intentional rational),
- be able to classify and formally describe different decision scenarios occurring in operational practice,
- be able to practically apply methods proposed by decision theory to typical example scenarios from everyday business life,
- Using numerous examples, develop the ability to recognize and discuss the possibilities and limitations of decision theory,
- be able to critically discuss the problem of how "rational decision-making" and "optimal decisions" can be defined and adequate solutions identified.

Course Content:

Normative and descriptive decision theory.

- Basic model of decision theory (axiomatics, preference functions, outcome matrix, utility measurement, decision rule).
- Formal representation of single- and multi-stage decision scenarios
- Decisions in the face of certainty, risk and uncertainty
- Information gathering in uncertain decision scenarios
- Decisions by decision-making bodies

(Interactive decision theory or "game theory" is not covered).

Literature:

Bamberg G, Coenenberg A, Krapp M: Betriebswirtschaftliche Entscheidungslehre, 15th edition. Vahlen, Munich (2012).

- Kahnemann D, Tversky A: "Prospect Theory: An analysis of decision under risk". Econometrica, 47 (2), 263-292 (1979).
- Klein R, Scholl A: Planung und Entscheidung, 2nd edition. Vahlen, Munich (2011).
- Sen A: "Rationality and Social Choice." American Economic Review, 85 (1), 1-24 (1995).

(Further literature will be announced during the course if necessary)Forms

of work / resources: Seminar-like lecture with practical exercises

Department: Computer Science

Professional group: Information Systems

Module responsibility: Oliver Skroch

Genetic Algorithms

Module numbers: 30.2536 [PVL 30.2537]

Language: english

Study programme:

Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization AE: Application Engineering
Bachelor 2007 - Specialization TI: Computer Engineering
Bachelor 2007/2004/2002/99 - Elective courses from the computer science area KoSI 2007 - Specialization AE: Application Engineering

KoSI 2007 - Specialization TI: Computer Engineering
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): notgraded
not graded.
Successful participation in the laboratory.The successful participation in the laboratory consists of implementing a genetic algorithm. The genetic operators for mutation and recombination, as well as the fitness proportionate and rank based selection must be implemented, and the suitability of the algorithm must be shown with the help of test instances.

Required knowledge: basic bachelor-level programming skill (C++ or Java)

Learning objectives:

Knowledge
The students understand the structure of algorithms which rely on the concept of evolution.

- Skills
  - In the laboratory the students have learned to implement a genetic algorithm to solve an underlying search or optimization problem.
- Competencies
  - The students have learned how to solve optimization, search, and other problems with genetic algorithms and know how to deal with problem specific challenges.

Content:

Required key concepts from biology, such as evolution, chromosome, genotype, phenotype, etc....

- The structure of a genetic algorithm and genetic operators.
- Differences between genetic algorithms and other heuristics, such as hill climbing, simulated annealing, etc..
- The theory behind genetic algorithms (schema theorem, implicit parallelism, etc.).
- Practical applications for genetic algorithms and specialized genetic operators.
- Genetic Programming as an advanced branch of genetic algorithms.

Literature:

M. Mitchell: An Introduction to Genetic Algorithms, MIT Press, 1996.

- Z. Michalewicz: Genetic Algorithms + Data Structures = Evolution Programs, Springer Verlag, 3rd edition, 1999.
- D. E. Goldberg : Genetic Algorithms in Search, Optimization and Machine Learning , Addison-Wesley 1989
- W. Banzhaf et al.: Genetic Programming, Morgan Kaufmann Publishers , 1998
- K. O. Stanley, J. Lehman: Why Greatness Cannot Be Planned, Springer Verlag, 2015
- Various publications from scientific journals

Faculty: Computer Science

Expert group: Artificial intelligence

Responsibility: Alexander del Pino

Genetic algorithms

English Title:	Genetic Algorithms
Document numbers:	30.2280 [PVL 30.2281]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization AE: Application Engineering Bachelor 2007 - Specialization TI: Computer Engineering Bachelor 2007/2004/2002/99 - Electives from the Computer Science Department KoSI 2007 - Specialization AE: Application Engineering KoSI 2007 - Specialization TI: Computer Engineering KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. practical course):	ungraded(The successful completion of the preliminary examination consists of the implementation of a genetic algorithm. The genetic operators for mutation and recombination, as well as fitness-proportional and rank-based selection must have been implemented, and the executability must be proven with the help of test instances).
Prerequisite knowledge:	Basic bachelor's level knowledge of programming (C++ or Java).
Learning Objectives:	Knowledge Students will understand the principles and operation of programming techniques based on the concept of evolution from biology. • Skills – In the accompanying practical course, the participants have acquired the ability to put this knowledge into practice in order to solve concrete problems with the help of such methods. • Competencies – Using case studies, the participants learned how concrete optimization, search and other problems can be solved with such methods, and which difficulties have to be solved in individual cases.
Course content:	Required biological principles (evolution, chromosome, genotype, phenotype, etc.) • The structure of a genetic algorithm and the basic genetic operators. • Differentiation of genetic algorithms from other methods such as hill climbing, simulated annealing, etc. • The theory behind genetic algorithms (schema theorem, implicit parallelism, etc.) • Practical applications for genetic algorithms and specialized genetic operators. • Genetic programming as a further development of genetic

Algorithms.

Literature: M. Mitchell: An Introduction to Genetic Algorithms, MIT Press, 1996.

- Z. Michalewicz: Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 3rd edition, 1999.
- D. E. Goldberg: Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley 1989
- W. Banzhaf et al.: Genetic Programming, Morgan Kaufmann Publishers, 1998
- Various publications from professional journals.

Forms of work / resources: Seminar lecture
Internship: Teamwork in small working groups and presentation of internship results.
Lecture slides

Department: Computer Science

Division: Artificial Intelligence

Module responsibility: Alexander del Pino

Graph Data Science

Module numbers: 30.2674 [PVL 30.2675]

Language: english

Study programme: Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded
The practical part (compulsory attendance if not agreed otherwise) consists of two parts:

- Smaller assignments, where correct solutions resp. successful technological implementations have to be shown.
- Individual accompanied project over the whole semester with datasets+technologies of own choice. Results have to be presented in front of the whole group and/or short documentation has to be provided.

ATTENTION: Taking part in the practical may require a user registration for different cloud services from Google, Microsoft, Amazon or similar. Do not participate in this module if this prerequisite cannot be met.

Required knowledge: Mathematics
1
Mathematics
2
Algorithms and Data Structures
Programming 1
Programming 2
Databases
Distributed Systems
Recommended: Introduction to artificial intelligence

Learningobjectives:

Aftersuccessfully completing this lecture, students should be able to efficiently represent real-world problems and datasets as graph structures in a database. Based thereupon, they should be able to apply automated methods for quantitatively analyzing graph structures and for data-based statistical model inference. Finally, students should be able to interpret and communicate the generated results.

Content:

The lecture consists of the following components:

- Graph Data Science Applications, Complex Networks
- Graph Definitions and Terminology, Toolset
- Property Graph Modeling, Storage and Querying
- Graph Theory Basics
- Quantifying Graph Structures - Vectorization and Importance metrics for nodes and edges (closeness, betweenness, hubs, authorities, neighborhoods)
- Basic Graph Clustering: k-cores, communities
- Random Networks (Generators), Scale Free Property, Preferential Attachment
- Advanced Clustering and Machine Learning on Graphs: Random-Walk based link prediction, Embeddings, Graph (Convolutional) Neural Networks
- Large Graphs & Distributed Graph Processing

Literature:

D. Chakrabarti, C. Faloutsos. Graph Mining. Laws, Tools, and Case Studies. Morgan & Claypool Publishers, 2012

- Albert-László Barabási. Network Science. Cambridge University Press, 2016
- M. Junghanns, A. Petermann, M. Neumann, E. Rahm. Management and Analysis of Big Graph Data: Current Systems and Open Challenges. Big Data Handbook. Springer, 2017
- Filippo Menczer, Santo Fortunato and Clayton A. Davis. A First Course in Network Science. Cambridge University Press, 2020
- William L. Hamilton. Graph Representation Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Vol. 14, No. 3 , Pages 1-159. McGill University, 2020.
- Tanmoy Chakraborty. Social network analysis. Wiley India, 2021

Faculty: Computer Science

Expert group: Data Science

Responsibility: Markus Döhring

Graph Data Science

Document numbers:	30.2676 [PVL 30.2677]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded The practical part (compulsory attendance unless otherwise agreed in advance)

consists of two parts:

- Smaller tasks where the correct solution or correct implementations must be submitted.
- Accompanying individually designed project over the entire lecture period with data set and technology of the students' choice. Interim results have to be presented occasionally in front of the whole group and at the end a short project documentation (approx. 5-6 pages) has to be prepared.

ATTENTION: Participation in the internship may require personal registration with cloud services from Google, Microsoft, Amazon or similar. In the absence of agreement with these preconditions, it is necessary to take an alternative elective module.

Required prior knowledge:

Mathematics 1
Mathematics 2
Algorithms and Data Structures
Programming 1
Programming 2
Databases
Distributed Systems
Recommended: Introduction to Artificial Intelligence

Learning Outcomes:

Upon successful completion of the module, students will be able to efficiently represent real-world problems in the form of data sets as graph-like structures in databases. Based on this, students will be able to apply automatic methods for quantitative analysis of graph structures and for data-based statistical model inference. Finally, students will be able to interpret and communicate the generated results (e.g. in graphical form).

Course content:

The lecture consists of the following components:

- Graph Data Science Applications, Complex Networks
- Graph Definitions and Terminology, Toolset
- Property Graph Modeling, Storage and Querying
- Graph Theory Basics
- Quantifying Graph Structures - Vectorization and Importance metrics for nodes and edges (closeness, betweenness, hubs, authorities, neighborhoods)
- Basic Graph Clustering: k-cores, communities
- Random Networks (Generators), Scale Free Property, Preferential Attachment
- Advanced Clustering and Machine Learning on Graphs: Random-Walk based link prediction, Embeddings, Graph (Convolutional) Neural Networks
- Large Graphs & Distributed Graph Processing

Literature:

D. Chakrabarti, C. Faloutsos. Graph Mining. Laws, Tools, and Case Studies. Morgan & Claypool Publishers, 2012

- Albert-László Barabási. Network Science. Cambridge University Press, 2016
- M. Junghanns, A. Petermann, M. Neumann, E. Rahm. Management and Analysis of Big Graph Data: Current Systems and Open Challenges. Big Data Handbook. Springer, 2017
- Filippo Menczer, Santo Fortunato and Clayton A. Davis. A First Course in Network Science. Cambridge University Press, 2020
- William L. Hamilton. Graph Representation Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Vol. 14, No. 3 , Pages 1-159. McGill University, 2020.
- Tanmoy Chakraborty. Social network analysis. Wiley India, 2021

Department: Computer Science

Expert Group: Data Science

Module responsibility: Markus Döhring

Basics of quality management

English Title: Principles of Quality Management

Document number: 30.2318

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization AE: Application Engineering
Bachelor 2007 - Specialization WI: Business Informatics
Bachelor 2007/2004/2002/99 - Electives from the Computer Science
Department KoSI 2007 - Specialization AE: Application Engineering
KoSI 2007 - Specialization WI: Business Informatics
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Teaching format: V = Lecture

SWS: 2

CP: 2.5

Exam: Written exam

Learning objectives:
Knowledge: The students know theoretical basics and practice-related methods and procedures of quality management. They know the tasks of quality management in the implementation of projects, in line tasks and in the provision of services in the DP and
IT environment. Students will be familiar with quality assurance measures in ongoing production operations. They can appropriately classify quality management in the IT environment.

Skills: Students practice applying their knowledge to simple real-world examples, integrating selected methods and procedures. They can then classify these independently and apply them in simple situations.

Course content:
History of quality and quality management
- Importance of quality in the company
- Basics of QM
- 7 Quality tools
- 7 Management tools
- Normative quality management systems, e.g.
  - DIN EN ISO 9000
  - TQM Systems/Strategic Quality Programs
  - EFQM
  - SPICE/CMMI
  - ITIL
- Operational Excellence
- Integrated Management Systems
- Compliance management
- Product and producer liability
- Project Management

Literature:
G. Benes, P. Groh: Fundamentals of Quality Management; Carl Hanser Verlag; 2012.
- J. Ensthaler: Product and Producer Liability; Pocket Power, Carl Hanser

Publisher, 2006
- Th. Hummel, Ch. Malorny: Total Quality Management; Pocket Power, Carl Hanser Verlag, 2011
- G. Kamiske: Handbuch QM-Methoden: Selecting the right method and implementing it successfully, Carl Hanser Verlag, 2013.
- W. Masing: Handbuch Qualitätsmanagement, Carl Hanser Verlag, 2007
- E. Wallmüller: Software Quality Engineering: A Guide to Better Software Quality; Carl Hanser Verlag, 2011.

Forms of work / resources:	seminar lecture Script and further documents on the websites of the lecturers
Department:	Computer Science
Professional group:	Software Engineering
Module responsibility:	Urs Andelfinger

High Performance I/O

Module numbers:	30.2664 [PVL 30.2665]
Language:	english
Study programme:	Bachelor 2021 - Elective catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Type of course:	V+P = Lecture+Practical
Weekly hours:	2+2
Credit Points:	5
Exam:	written exam
PVL (e.g. Practical):	not graded
Requiredknowledge:	Bachelor-levelknowledge in operating systems and distributed

systems Learning objectives:

Students
- are able to apply I/O performance analysis methods
- know operating systems, concepts, architectures and APIs used to obtain high

performance I/O and are able to evaluate the impact on application performance
- are able to identify suitable applications and assess the required effort to apply

the methods learned
- are able to implement applications in the area of high performance I/O as well as

determine and evaluate their performance
- are able to asses the impact of virtualization mechanisms on the I/O performance of

applications

Content:

I/O performance problems related to network and file systems
- I/O performance analysis
- influence of OS architectures on I/O performance
- programming interfaces
- High performance network I/O in Linux
- zero-copy mechanisms

	- high performance I/O and virtualization
Literature:	Rai Jain: The Art of Computer Systems Performance Analysis, Wiley Professional Computing, 1991. - Brendan Gregg: Systems Performance: Enterprise and the Cloud, Prentice Hall, 2013
Faculty:	Computer Science
Expert group:	Operating systems / distributed systems
Responsibility:	Lars-Olof Burchard

High Performance I/O

Document numbers:	30.2620 [PVL 30.2621]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded (ungraded internship).
Prerequisite knowledge:	Bachelor's level knowledge of operating systems and distributed systems
Learning objectives:	Students - can apply methods for analyzing I/O performance - Know operating systems, concepts and architectures, and APIs that enable high performance I/O and evaluate their impact on application performance. - can identify suitable application possibilities and assess the effort required for implementation - can carry out own implementations in the area of High Performance I/O, as well as measure and evaluate their quality - Can assess impact of virtualization on application I/O
Course content:	problem input/output for network and files. - I/O performance analysis - Influence of BS architectures on I/O performance - Programming interfaces - High Performance Network I/O in Linux - Zero-copy mechanisms - High Performance I/O and Virtualization
Literature:	Rai Jain: The Art of Computer Systems Performance Analysis, Wiley Professional Computing, 1991. - Brendan Gregg: Systems Performance: Enterprise and the Cloud, Prentice Hall, 2013
Department:	Computer Science

Division:	Operating Systems / Distributed Systems
Module responsibility:	Lars-Olof Burchard

Intensive day GDV

English title:	Intensive Training GDV
Document number:	30.7402T
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	T = Tutorial
SWS:	2
CP:	0
Learning objectives:	This course is aimed at students who find it very difficult to get started in graphical data processing. Studying and thus learning is usually divided into 90min blocks. A flow -- a state of high motivation as well as high concentration -- is difficult to achieve. The goal of the intensive days is an intensive examination of a topic, at one's own pace without frustration and with a lot of fun and high motivation, to really grasp and penetrate the topics, to develop confidence in solving corresponding tasks, to collect aha-experiences. The intensive day is not a revision course, a substitute lecture or pure exam preparation. There is no additional knowledge transfer, no frontal repetition of the contents. You apply the contents of GDV to suitable tasks that correspond to your current competencies.
Teaching content:	cf. GDV
Literature:	cf. GDV
Forms of work / resources:	Organized learning group with tutorial support. Information and dates: https://www.fbi.h-da.de/studium/lernen/intensivtage.html
Department:	Computer Science
Professional group:	Multimedia and graphics
Module responsibility:	Dean of Studies

Intensive day PAD1

English title:	Intensive Training PAD1
Document number:	30.7104T
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer

	Bachelor KMI - additional offer KoSI - additional offer
Teaching Form:	T = Tutorial
SWS:	2
CP:	0
Learning objectives:	This offer is aimed at students with no previous knowledge of programming or students who find it very difficult to get started in C++. The study and thus the learning is usually divided into 90min blocks. A flow -- i.e. a state of high motivation as well as high concentration -- is therefore difficult to achieve. The goal of the intensive days is an intensive examination of a topic, in own speed without frustration and with much fun and high motivation to really understand and penetrate the topics, to develop security in programming, to collect aha experiences. The intensive day is no repetitorium, no substitute lecture and also no pure exam preparation. There is no additional knowledge transfer, no frontal repetition of the contents. You apply the contents of PAD1 to suitable tasks that correspond to your current competencies.
Course Content:	Selected individualized topics from PAD1.
Literature:	cf. PAD1
Forms of work / aids:	Self-assessment test,independent work with prepared or own tasks with minimal help from tutors, i.e. tutors give feedback and as much help as you need for the next own step to solve a task, to understand. Group work. Reflection. Plenary discussion. Information and dates: https://www.fbi.h-da.de/studium/lernen/intensivtage.html
Department:	Computer Science
Expert group:	Programming
Module responsibility:	Dean of Studies

Intensive day PAD2

English title:	Intensive Training PAD2
Document number:	30.7208T
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	T = Tutorial
SWS:	2
CP:	0
Required prior knowledge:	Programming, Algorithms and Data Structures 1
Learning Objectives:	This course is designed for students who find it very difficult to get started in C++. The study and thus the learning is usually in

	90min blocks. A flow -- i.e. a state of high motivation as well as high concentration -- is thus difficult to achieve. The goal of the intensive days is an intensive examination of a topic, at one's own pace without frustration and with a lot of fun and high motivation to really grasp and penetrate the topics, to develop security in programming, to gather aha experiences. The intensive day is not a revision course, a substitute lecture or pure exam preparation. There is no additional knowledge transfer, no frontal repetition of the contents. You apply the contents of PAD2 to suitable tasks that correspond to your current competencies.
Course Content:	Selected individualized topics from PAD2.
Literature:	cf. PAD2
Forms of work / aids:	Independent work with prepared or own tasks with minimalhelp from tutors, i.e. tutors give feedback and as much help as you need for the next own step to solve a task, to understand. Group work. Reflection. Plenary discussion. Information and dates: https://www.fbi.h-da.de/studium/lernen/intensivtage.html
Department:	Computer Science
Expert group:	Programming
Module responsibility:	Dean of Studies

Intensive day computer architecture

English Title:	Intensive Training RA
Document number:	30.7106T
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	T = Tutorial
SWS:	2
CP:	0
Learning objectives:	This course is aimed at students who find it very difficult to get started in computer architecture. The study and thus the learning is usually divided into 90min blocks. A flow -- i.e. a state of high motivation as well as high concentration -- is therefore difficult to achieve. The goal of the intensive days is an intensive examination of a topic, in own speed without frustration and with much fun and high motivation to really understand and penetrate the topics, to develop security in programming, to gather aha experiences. The intensive day is not a revision course, a substitute lecture or pure exam preparation. There is no additional knowledge transfer, no frontal repetition of the contents. You apply the contents of RA to suitable tasks that correspond to your current competencies.
Course Content:	Selected individualized topics from RA.

Literature: cf. RA

Forms of work / aids: Independent work with prepared or own tasks with minimalhelp from tutors, i.e. tutors give feedback and as much help as you need for the next own step to solve a task, to understand. Group work. Reflection.
Information and dates:
https://www.fbi.h-da.de/studium/lernen/intensivtage.html

Department: Computer Science

Expert group: Computer Engineering

Module responsibility: Dean of Studies

Internetworking

Module numbers: 30.2602 [PVL 30.2603]

Language: english

Study programme:
Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+P = Lecture+Practical

Weekly hours: 3+1

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded (not grades (successful participation on practical))

Requiredknowledge: Networking and Distributed Systems on bachelor level

Learning objectives: The students:

- are able to analyze the structure and extended functionality of Internet Protocol (IP) based networks
- are able to self-reliantly design, configure, and operate IP-based networks and services based on the knowledge obtained with this course
- know about further reading in the literature
- and understand the methods to keep their knowledge up to date in the fast changing subject of data networks

Content: This module builds on top of the modules "Networks" and "Distributed Systems". The goal is to enhance the theoretical knowledge of computer networks and telecommunication networks.

The content is:

- advanced IPv4 and IPv6,
- transition technologies from IPv4 to IPv6
- transport protocols (e.g., TCP and extensions, Multipath-TCP, SCTP)
- structure and functions of network elements (e.g., router and middleboxes)
- design and operation of IP-based networks
- specialized IP protocols (e.g., IP multicast and IP mobility)
- alternative network protocols (e.g., Peer-to-Peer, Information Centric Networking (ICN))
- further network protocols as necessary (e.g., Instant Messaging and Email)

- current topics out of the packet based data networks

Literature: Computer Networks, Kurose & Ross, Pearson Publisher,
- (Product) Documentation,
- lecture notes,
- Request For Comments (RFCs),
- Internet Drafts,
- Scientific publications

Faculty: Computer Science

Expert group: Telecommunications

Responsibility: Martin Stiemerling

Introduction to artificial intelligence

Module numbers: 30.2666 [PVL 30.2667]

Language: english

Study programme: Bachelor 2021 - Elective S_5/6 Catalog
Bachelor dual KITS 2021 - 6th semester
Bachelor dual KoSI 2021 - Elective S_5/6 Catalog
Bachelor KMI 2021 - 4th semester

Type of course: V+P = Lecture+Practical

Weekly hours: 3+1

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded (Successful participation in the practical.)

Prerequisite for booking: Themodules "Mathematics 1", "Mathematics 2", and "Programming 2" must be passed successfully.

Learning objectives: The students
- know the different areas of Artificial Intelligence and their corresponding basic approaches and strategies
- understand how AI applications are structured in principle
- know for each of these areas the basic methods and

algorithms The students
- are able to use the appropriate technologies for given problems in order to solve non-trivial problems
- can estimate where AI solutions are

appropriate The students
- can adapt methods to develop and realize proposals for solutions
- can develop a critical view of progression in AI against the background of philosophical foundations and ethical questions as well as recognize and assess risks and possible technological consequences of the development of systems with AI technologies

Content: The lecture provides an overview of the areas of AI with references to in-depth courses. The following content is covered:
- Machine learning (ML): Basic ML procedures based on prominent examples such as artificial neural networks or decision trees; Metrics / evaluation procedures for measuring the quality of ML predictions. Relation to symbolic and non-symbolic AI
- Representation and processing of knowledge: basic procedures, e.g. Ontologies and linked data; Query languages and reasoning. Relation to symbolic and non-symbolic AI

- Natural language processing (NLP): Application areas of NLP such as document classification, machine translation or human-machine communication, as well as current technologies for their implementation; Relation to symbolic and non-symbolic AI.
- Computer vision: areas of application such as object recognition on images, as well as current technologies for implementing them; Relation to non-symbolic AI.
- Cross-cutting issues: philosophical foundations and ethical questions of AI; Opportunities and risks of autonomous systems; Bias in AI applications; Effects of AI applications on society and working life.

All content is practiced in the practical.

Literature:

Bernhard G Humm: Applied Artificial Intelligence - An Engineering Approach. Second Edition. Leanpub, Victoria, British Columbia, Canada, 2016. leanpub.com/AAI

- Russel, S. / Norvig, P. Artificial Intelligence: A Modern Approach (Pearson Series in Artificial Intelligence), 4th ed, 2020.

Further literature:

- Christopher M. Bishop. 2006. pattern recognition and machine learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.
- Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani: An Introduction to Statistical Learning. New York, NY, USA : Springer New York Inc., 2001 (Springer Series in Statistics, vol. 103).
- Ian Goodfellow, Yoshua Bengio and Aaron Courville "Deep Learning", MIT Press 2016.
- Jurafsky, Daniel / Martin, James. 2014. speech and language processing. An Introduction to Natural Language Processing, 2nd ed. Pearson India.

Faculty:	Computer Science
Expert group:	Artificial intelligence
Responsibility:	Gunter Grieser

Introduction to Machine Learning

Module numbers:	30.2588 [PVL 30.2589]
Language:	english
Study programme:	Bachelor 2021 - Elective catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization AE: Application Engineering KoSI 2007 - Specialization AE: Application Engineering KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Type of course:	V+P = Lecture+Practical
Weekly hours:	3+1
Credit Points:	5
Exam:	Accompanying tests and evaluation of the solution of the problem sets
PVL (e.g. Practical):	not graded

Required knowledge:	linear algebra, statistics, basics of programming
Learningobjectives:	Thestudents will be able to apply learning algorithms to building smart robots (perception, control), text understanding (web search, anti-spam), computer vision, medical informatics, audio, database mining, and other areas.
Content:	1. linear regression with one variable 2. Linear Algebra Review 3. Linear Regression with Multiple Variables 4. Logistic Regression 5. Regularization 6. Neural Networks: Representation 7. Neural Networks: Learning 8. Deep Learning 9. Decision trees 10. Machine Learning System Design 11.Unsupervised Learning (clustering) 12. Dimensionality Reduction 13. Anomaly Detection 14. Recommender Systems 15. Large Scale Machine Learning
Literature:	1. Mitchell, Tom. Machine Learning. New York, NY: McGraw-Hill, 1997. ISBN: 9780070428072. 2. https://www.coursera.org/learn/machine-learning 3. MacKay, David. Information Theory, Inference, and Learning Algorithms. Cambridge, UK: Cambridge University Press, 2003. ISBN: 9780521642989. Available on-line http://www.inference.phy.cam.ac.uk/mackay/itila/book.html.
Lecture style / Teaching aids:	15 Lessons with 4
problemsets Faculty:	Computer Science
Expert group:	Data Science
Responsibility:	Arnim Malcherek
Professional competencies:	formal, algorithmic, mathematical competencies: high • analytical, design and implementation competencies: high • technological competencies: high • capability for scientific work: low
Interdisciplinary competencies:	• project related competencies: medium • interdisciplinary expertise: basic technical and natural scientific competence • social and self-competencies: analytical competence, competence of knowledge acquisition, fluency

IT Risk Management

English Title:	IT Risc Management
Document number:	84.2006
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KITS 2014 - Catalog ITS: IT Security Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+Ü = Lecture+Exercise

SWS: 3+1

CP: 5

Exam: Written exam

Learning Objectives: Students will

- know the essential terms of risk management
- can apply the individual steps of a risk management process (identification, analysis, management / measures, control)
- Can view risks by category
- can evaluate risks
- know common measures (accept, avoid, minimize, shift) to manage risk
- are familiar with common risk controlling methods
- know relevant standards, regulations and laws

Course content: Risk management terms and principles

- Risk management process
- Methods in the risk management process
- Categorization of risks
- Risk assessment
- Risk management measures
- Risk Controlling
- ISO 31000
- Risk communication and corporate culture

Literature: Knoll, M. 2014. Practice-oriented IT risk management: design, implementation and review. Heidelberg: dpunkt.verlag.

- Königs, H.-P. 2017. IT risk management with a system: practice-oriented management of information security, IT and cyber risks, (5th ed.). Wiesbaden: Springer Fachmedien.
- Prokein, O. 2008. IT risk management: identification, quantification and economic control. Wiesbaden: Gabler.
- Romeike, F. 2018. risk management. Wiesbaden: Springer Gabler.
- Fabian Ahrendts, Anita Marton, Living IT Risk Management, Springer, 2008
- ISO 31000 Risk Management
- Walter Ruf, Thomas Fittkau, Holistic IT Project Management, Oldenbourg, 2007
- Pascal Mangold, IT-Projektmanagement kompakt, Spektrum Verlag

2009Work forms / resources: Lecture notes,exercises

Department: Computer Science

Expert group: IT Security

Module responsibility: Andreas Heinemann

Java Enterprise Database Application Development

English Title: Java Enterprise Database-Driven Application

Development Document Numbers: 30.2636 [PVL 30.2637; Module 30.26360]

Language: GERMAN

Assignment: Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I

Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): graded(In the internship, students gain practical experience in implementing the theoretical content taught in the lecture).

Proportion PVL: 40%

Required Prior Knowledge: Basic bachelor's level knowledge of databases (including ORM), Software Engineering as well as the development of user-centric and web-based applications. Ideally, you have participated in the modules Databases 2 (document no. 30.7406) as well as Software Engineering (document no. 30.7318).

Learning Objectives:

Students should be able to master different paradigms in the development of Java-based database applications and, in particular, assess their specific advantages and disadvantages for the respective application scenario.
- Students will gain hands-on experience in developing and testing database applications based on Spring Boot and JPA

Course content:

Introduction to the Java Enterprise Architecture as well as the related Java Web technologies and frameworks
- Imparting best practice in various areas of software development
- Consideration of alternative as well as complementary database concepts

Literature:

B. Müller, H. Wehr: Java Persistence API 2: Hibernate, EclipseLink, OpenJPA and Extensions, Hanser, 2012.
- A. Gupta: Java EE 7 Essentials, O'Reilly Media, 2013
- Spring Boot 2: Modern Software Development with Spring 5

Work forms / resources: seminar lecture and practical course, script, supplementary examples, electronically available materials, exam examples

Department: Computer Science

Professional group: Databases

Module responsibility: Stefan T. Ruehl

Networks (tutorial)

English Title: Networks (Tutorial)

Document number: 30.7102T

Language: GERMAN

Assignment:

Bachelor - additional offer
Bachelor dual KESS - additional offer
Bachelor dual KITS - additional offer
Bachelor dual KoSI - additional offer
Bachelor KMI - additional offer
KoSI - Additional offer
Bachelor 2014 - 2nd semester
Bachelor dual KESS 2014 - 4th semester

	Bachelor dual KITS 2014 - 4th semester Bachelor dual KoSI 2014 - 2nd semester
Teaching Form:	OL = Open lab
SWS:	2
CP:	0
Department:	Computer Science
Professional group:	Telecommunications

Object-oriented and object-relational databases

English Title:	Object-Oriented and Object-Relational Databases
Document Number:	30.2366 [PVL 30.2367]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization AE: Application Engineering Bachelor 2007 - Specialization WI: Business Informatics Bachelor 2007/2004/2002/99 - Elective courses from the computer science area KoSI 2007 - Specialization AE: Application Engineering KoSI 2007 - Specialization WI: Business Informatics KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. practical course):	ungraded (Bi-weekly completion of practice sheets. 100% of the Tasks must be solved appropriately for successful completion of the PVL).
Prerequisite knowledge:	Basic undergraduate level knowledge of programming, databases, and object-oriented analysis and design.
Learning Objectives:	Students will • know the architecture of non-relational database systems (object-oriented, object-relational and i.e. NoSQL database systems) as well as - in comparison - hierarchical and network database systems, • be able to transform semantic data models into schemas of object-oriented, object-relational and NoSQL database systems, • Be able to use APIs of object-oriented, object-relational and NoSQL database systems and • Be able to use object-oriented, object-relational and NoSQL database systems.
Teaching content:	Architecture of object-oriented, object-relational and NoSQL database management systems and - in comparison - the architecture of hierarchical and network database management systems.

Literature:	Heuer: Object-Oriented Databases Addison-Wesley 1997 (2nd edition). • Cattell et al. (Eds.): The Object Database Standard: ODMG 3.0 Morgan Kaufmann Publishers 2000 • Can Türker: SQL:1999 & SQL:2003 dpunkt.verlag 2003 • Jim Paterson, Stefan Edlich,Henrik Hörning, and Reidar Hörning: The Definitive Guide to db4o, Apress 2006 • Stefan Edlich et al.: NoSQL - Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken; Hanser 2011 (2nd edition).
Forms of work / resources:	Seminar lecture and practical course Script, supplementary examples
Department:	Computer Science
Professional group:	Databases
Module responsibility:	Stephan Karczewski

Open laboratory EWA

English Title:	Open Lab EWA
Document number:	30.103Z
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	OL = Open lab
SWS:	2
CP:	0
Department:	Computer Science
Professional group:	Software Engineering

Open laboratory GDV

English title:	Open Lab GDV
Document number:	30.102Z
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	OL = Open lab
SWS:	4
CP:	0
Department:	Computer Science

Professional group:	Multimedia and graphics

Open Lab IT Security

English Title:	Open Lab IT Security
Document number:	30.101Z
Language:	GERMAN
Assignment:	Bachelor - additional offer Bachelor dual KESS - additional offer Bachelor dual KITS - additional offer Bachelor dual KoSI - additional offer Bachelor KMI - additional offer KoSI - Additional offer
Teaching Form:	OL = Open lab
SWS:	4
CP:	0
Department:	Computer Science
Expert group:	IT Security

Penetration Testing

Module numbers:	30.2606 [PVL 30.2607; Module 30.26060]
Language:	english
Study programme:	Bachelor 2021 - Elective catalog I Bachelor dual KITS 2021 - Elective Catalog ITS Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KITS 2014 - Catalog ITS: IT Security Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Type of course:	V+P = Lecture+Practical
Weekly hours:	2+2
Credit Points:	5
Exam:	written exam
PVL (e.g. Practical):	graded (graded report)
PVL percentage:	50%
Required knowledge:	Operating systems, Networking, Developing of web applications, Distributed systems
Learning objectives:	Knowledge: • Definition and classification of hacking and penetration testing • Relevant standards regarding stack phases and risk assessment • Best practices for documentation and reporting • Tools and techniques for identifying and exploiting vulnerabilities Skills:

- Identification of vulnerabilities in IT systems and utilizing them to penetrate the system
- Risk-based evaluation of vulnerabilities
- Documenting the approach and results

Competencies:

- Conducting a reproducible technical security analysis of an IT infrastructure
- Generating a structured report on the results of a technical security analysis of an IT infrastructure

Content: Differences between hacking and penetration testing

- Classification of penetration tests (white, gray and black box tests)
- Penetration Testing Standards, e.g. OWASP (Open Web Application Security Project), OSSTMM (Open Source Security Testing Methodology Manual)
- Anatomy of an attack - from information gathering to exploitation of a vulnerability
- Risk assessment of identified vulnerabilities
- Structure of documentation and reporting

Literature: P. Engebretson; The Basics of Hacking and Penetration Testing; Syngress; 2013.

- P. Engebretson; Hacking Handbuch: Penetrationstests planen und durchführen; Franzis Verlag; 2015
- M. Ruef; The Art of Penetration Testing - Handbook for Professional Hackers; C & L; 2007
- BSI https://www.bsi.bund.de/DE/Publikationen/Studien/Pentest/index_htm.htm
- OWASP Testing Guide https://www.owasp.org/index.php/OWASP_Testing_Project
- OSSTMM http://www.isecom.org/research
- Metasploit Unleashed
  Lecture style / Teaching aids: Seminar lecture,

script

Faculty: Computer Science

Expert group: IT Security

Responsibility: Christoph Krauß

Penetration Testing

Document numbers: 30.2608 [PVL 30.2609; Module 30.26080]

Language: GERMAN

Assignment: Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog ITS
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KITS 2014 - Catalog ITS: IT Security
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): graded (graded report)

Proportion PVL: 50%.

Required prior knowledge: Operating systems, networking, web-based application development, distributed systems.

Learning Objectives:

Knowledge:

- Definition and classification of hacking and penetration testing
- Relevant standards for risk assessment
- Best practices for documentation and reporting
- Tools and techniques for identifying and exploiting vulnerabilities

Skills:

- Identification of vulnerabilities in IT systems and their exploitation
- Vulnerability risk assessment
- Documentation of results

Competencies:

- Implementation of a reproducible, technical security analysis of IT infrastructures
- Generation of a structured report on the results of a technical security analysis of IT infrastructures

Course Content:

Differences between hacking and penetration testing.

- Classification of a penetration test (white, gray and black box test)
- Penetration testing standards, e.g. OWASP (Open Web Application Security Project), OSSTMM (Open Source Security Testing Methodology Manual)
- Anatomy of an attack - from information gathering to exploitation of a vulnerability
- Risk assessment of identified vulnerabilities
- Structure documentation and reporting

Literature:

P. Engebretson; The Basics of Hacking and Penetration Testing; Syngress; 2013.

- P. Engebretson; Hacking Handbuch: Penetrationstests planen und durchführen; Franzis Verlag; 2015
- M. Ruef; The Art of Penetration Testing - Handbook for Professional Hackers; C & L; 2007
- BSI https://www.bsi.bund.de/DE/Publikationen/Studien/Pentest/index_htm.htm
- OWASP Testing Guide https://www.owasp.org/index.php/OWASP_Testing_Project
- OSSTMM http://www.isecom.org/research
- Metasploit Unleashed https://www.offensive-security.com/metasploit-unleashed/
- binsec Academy https://binsec.wiki/en/security/howto/pentest-

training/Work forms / resources:Seminar lecture, script

Department: Computer Science

Expert group: IT Security

Module responsibility: Christoph Krauß

Professional testing

English Title: Advanced Testing

Document numbers: 30.2476 [PVL 30.2477]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization AE: Application Engineering
Bachelor 2007/2004/2002/99 - Elective courses from the computer science
area KoSI 2007 - Specialization AE: Application Engineering
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (successful participation in the internship)

RequiredPrior knowledge: SoftwareEngineering, Programming / Algorithms and Data Structures at Bachelor level.

Learning Objectives:

Students will master current techniques for testing in software projects from the perspective of a software developer or software tester in practice.

- Graduates of the module are able to independently define and adapt test cases in a software project and to apply the common test procedures.

Course content:

Fundamentals of testing: types of tests, quality assurance through Testing, Testing in Software Engineering etc.

- Different test methods: e.g. classical test methods, test-driven development, agile testing
- Test case creation and test coverage
- Testing Techniques: Mocks and Stubs, Dependency Injection
- Concurrent code testing
- Test frameworks: e.g. JUnit, GoogleTest, Jest
- Management of the test process and defects
- Regression testing and test automation (Continuous Integration)
- Performance and load tests
- Many advanced practical examples of test procedures and test techniques from operational practice

Literature:

Frank Witte, Testmanagement und Softwaretest - Theoretische Grundlagen und praktische Umsetzung, Springer Vieweg, Wiesbaden, 2016.

- Baumgartner et.al, Agile Testing - Der agile Weg zur Qualität, Hanser Fachbuch, 2nd, revised and expanded edition. 11/2017

Forms of work / resources:

seminar lecture, exercises/practicum&#8232;

- Group work for the specification of test cases for component, integration and system tests
- Group work on the topics of test-driven development and test automation
- Resources: presentations, supplementary examples, MindMaps

- Computer exercises with Java and testing tools

Department: Computer Science

Professional group: Software Engineering

Module responsibility: Kai Renz

Quantum Computing

English title: Quantum computing

Document numbers: 30.2660 [PVL 30.2661]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: VP = Lecture with integrated practical course

SWS: 4

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (Active participation in the internships embedded in the lecture. The internships are ungraded but will be tested).

RequiredPrior knowledge: Theprogramming and mathematics lectures from the first two semesters should be successfully completed.

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

- Understand the physical and mathematical principles of a quantum computer.
- understand the programming paradigms for quantum computers
- code different simple algorithms for a quantum computer
- Assess and evaluate the potential of quantum computers for different groups of use cases.
- Understand the current physical realizations of quantum computers.
- Assess the state of current research

Course content: Mathematical Fundamentals: To a large extent, knowledge from the Lectures 'Mathematics for Computer Scientists' used, which must be supplemented and deepened in some places, e.g., around Hilbert spaces and their properties.

- Fundamentals of physics: quantum mechanics (principles, examples, and a short history overview)
- Qubits
- Quantum gates and circuits
- Existing quantum algorithms
- Exercise examples of quantum programming using simulation software.
- Existing and planned approaches to the realization of quantum computers.

Literature: Matthias Homeister: Understanding Quantum Computing, Springer Vieweg, 2018, 5th ed.

Circulation

- Eleanor Rieffel, Wolfgang Polak: Quantum Computing, MIT Press, 2014

Further literature will be announced at the beginning of the course.

Department: Computer Science

Expert group: Programming

Module responsibility: Arnim Malcherek

Rapid Prototyping

Module numbers: 30.2670 [PVL 30.2671; Module 30.26700]

Language: english

Study programme:
Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+Pro = Lecture+Project

Weekly hours: 2+2

Credit Points: 5

Exam: Formal presentation and demonstration of project outcome

PVL (e.g. Practical): graded (Work products (CAD design, PCB design, assembled prototype))

PVL percentage: 50%

Learning objectives:

Students can estimate the benefits and risks of hardware and software prototypes

- for startup companies within their business creation efforts
- for established companies in research, the user centered development process and in user evaluations

Students are able to produce simple hardware and software prototypes

- Find appropriate mechanical designs to host electronic components such as microcontrollers, displays and controls
- Can sketch 3D models suitable for rapid prototyping technologies such as 3D printing in a computer aided design (CAD) program such as FreeCAD
- Know frequently used rapid prototyping technologies such as fused filament fabrication, laser sintering, milling, laser cutting and can judge their suitability regarding mechanical and aesthetic properties as well as work effort, cost and lead time
- Can sketch a simple single or dual layer printed cirquit board (PCB) in an electronic design automation program such as KiCAD
- Have produced and assembled at least one case or PCB for a given prototype idea
- Know basics about Surface Mount Technologies (SMT) in PCB manufacturing
- Can integrate basic functionalities in microcontroller firmware using software rapid prototyping platforms such as Arduino
- Can setup basic software UI prototypes with mockup software tools

Content:

Topics include

- Rapid prototyping technologies
  - 3D printing (also known as fused filament fabrication (FFF) or fused deposition modeling(FDM))
  - Laser cutting and engraving
  - Laser sintering including metal sintering

  - Milling
  - Resin printing (also known as stereo lithography (STL))
  - Powder bonding
- Principles of CAD design
  - Working princibles of CAD systems
  - Parametric design
- Principles of PCB design
  - Schematics design
  - PCB Layout and routing
  - Electromagnetic compatibility (EMC) considerations
- Principles of prototype firmware design
  - Rapid prototyping frameworks
  - C/C++ development using a manufacturer supplied SDK
- Principles of software UI prototypes
  - Mockup software
  - Rapid prototyping using platform development
- Economic aspects of Rapid Prototyping
  - Minimal viable product (MVP)
  - Business consequences of ElektroG in Germany and in EU (RoHS, recycling of electronics, EMC)

Literature:

Dan Olsen (2015) "The Lean Product Playbook: How to Innovate with Minimum Viable Products and Rapid Customer Feedback", learning.oreilly.com.

- Douglas Bryden (2014) "CAD and Rapid Prototyping for Product Design," learning.oreilly.com.
- Andrew Gregory, editor (2022) "FreeCAD for makers", Hackspace Magazine
- Joan Horvath, Rich Cameron (2020) "Mastering 3D Printing: A Guide to Modeling, Printing, and Prototyping," learning.oreilly.com.
- Michael Margolis, Brian Jepson, Nicholas Robert Weldin (2020) "Arduino Cookbook," 3rd Edition, learning.oreilly.com.

Faculty: Computer Science

Expert group: Computer engineering

Responsibility: Stefan Rapp

Semantic Knowledge Management in Organizations

Module number: 30.2644

Language: english

Study programme:

Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam:

Assessment of individual project work (incl. presentation): 70%
Written assessment about the lecture material: 30%

Learning objectives:

After completing the module, students are capable of:

- modelling relevant entities of a knowledge management problem by means of

semantic knowledge graph technology
- formalizing a knowledge management problem by means of a semantic knowledge representation framework
- developing lightweight ontologies that represent the main entities of a problem domain and the relationships between them
- expressing attributes of and relationships between domain entities in the form of domain ontologies and semantic knowledge graphs
- implementing semantic knowledge graphs using the Semantic MediaWiki software
- Querying semantic knowledge graphs

Content:

The course consists of 6 main parts:
- Introduction to organizational knowledge management
- Foundations of semantic knowledge representation frameworks
- Introduction to Semantic MediaWiki
- Semantic MediaWiki extensions for the collaborative development of knowledge graphs and ontologies
- Individual project work
- Project presentation and written examination at the semester's

end The course teaches:
- Fundamentals of organizational knowledge management
- Foundations of semantic knowledge representation frameworks and knowledge organization systems
- Introduction to the Semantic MediaWiki software
- Building lightweight semantic domain models using Semantic MediaWiki
- Methods, tools and technologies for knowledge graph construction
- Formalization of domain entities and their relationships in the form of lightweight ontologies
- Querying semantic knowledge graph data

Literature:

Hitzler, P., Krötzsch, M.,, Rudolph, S. (2010). Foundations of semantic web technologies. Chapman and Hall/CRC Press. ISBN: 9781420090505
- Krötzsch, M., Vrandecic;, D. & Völkel, M. (2006). Semantic MediaWiki. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold & L. Aroyo (eds.), The Semantic Web - ISWC 2006 , Vol. 4273 (pp. 935--942) . Springer Berlin Heidelberg. ISBN: 978-3-540-49029-6.
- Zander, Stefan; Swertz, Christian; Verdú, Elena; Jesús Verdú Pérez, María; Henning, Peter. (2016). A Semantic MediaWiki-based Approach for the Collaborative Development of Pedagogically Meaningful Learning Content Annotations.
- Markus Krötzsch, Frantisek Simancik, Ian Horrocks (2013). A Description Logic Primer. https://arxiv.org/abs/1201.4089
- M. Krötzsch, F. Simancik and I. Horrocks, "Description Logics," in IEEE Intelligent Systems, vol. 29, no. 1, pp. 12-19, Jan.-Feb. 2014. doi: 10.1109/MIS.2013.123
- https://www.semantic-mediawiki.org/wiki/Help:User_manual

Additional literature recommendations will be announced in the lecture.

Lecture style / Teaching aids:

Seminaristic teaching style with practical exercises
Additional resources will be announced in the lecture

Faculty: Computer Science

Expert group: Artificial intelligence

Responsibility: Stefan Zander

Semantic knowledge management in the enterprise

English Title: Semantic Knowledge Management in

Organizations Document Number: 30.2646

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Examination:
Individual project work in the context of the internship with presentation of the implementation at the end of the semester (weighting: 70%)
Written exam on lecture material (weighting: 30%)

Learning Objectives:
Upon completion of the course, students will be able to:

- Independently identify a knowledge management problem and develop an IT-based approach to solving it based on the language and design concepts of semantic knowledge graph modeling.
- Operationalize an identified knowledge management problem, i.e., create a suitable semantic description model in the form of a lightweight ontology using Semantic MediaWiki.
- Formalize the relevant entities of an object domain and their relationships to each other and transform them into an ontology-based knowledge graph
- implement and refine lightweight ontologies and semantic knowledge graphs using Semantic MediaWiki software
- Apply basic language and design concepts from the field of semantic knowledge graph modeling to an existing knowledge management problem and implement them using Semantic MediaWik software.

Course content:
The LVA is divided into 3 main components:

- Introduction to basic technologies and theoretical foundations
- Individual project work (practical implementation of a knowledge management problem using Semantic MediaWiki)
- Project presentations at the end of the semester

The course broadly addresses the following questions:

- Why do companies need functioning knowledge management in the age of digitalization ?
- Why is knowledge management becoming more and more important for companies as digitization progresses and which technological approaches help here ?
- What new challenges does digitalization pose for companies and how can a functioning knowledge management system help to overcome them?
- How can the transformation of companies towards a learning organization be supported with technical measures ?
- What are the advantages of semantic knowledge graphs over other knowledge representation models in building corporate knowledge bases, the so-called "corporate knowledge spaces" ?

- What is the role of ontologies in knowledge representation and management ?
- How does Semantic MediaWiki support knowledge management in the enterprise ?
- By which technical measures can business users be involved in the construction of semantic knowledge graphs ?

Specifically, content from the following subject areas will be covered:

- Delimitation data - information - knowledge - action - competence
- Knowledge staircase for Industry 4.0
- Knowledge management models and approaches
- Basics of machine-processable semantics
- Languages and technologies for creating semantic
- Knowledge representation models and knowledge graphs
- Role of ontologies in sem. Knowledge Management
- Introduction to Semantic MediaWiki
- Methods and tools for ontology creation with Semantic MediaWiki
- Query languages for semantic knowledge representation models.
- Semantic MediaWiki extensions for collaborative ontology creation

Literature:

Hitzler, P., Krötzsch, M.,, Rudolph, S. (2010). Foundations of semantic web technologies. Chapman and Hall/CRC Press. ISBN: 9781420090505

- Krötzsch, M., Vrandecic, D. & Völkel, M. (2006). Semantic MediaWiki. In I. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold & L. Aroyo (eds.), The Semantic Web - ISWC 2006 , Vol. 4273 (pp. 935--942) . Springer Berlin Heidelberg . ISBN: 978-3-540-49029-6.
- Zander, Stefan; Swertz, Christian; Verdú, Elena; Jesús Verdú Pérez, María; Henning, Peter. (2016). A Semantic MediaWiki-based Approach for the Collaborative Development of Pedagogically Meaningful Learning Content Annotations.
- Markus Krötzsch, Frantisek Simancik, Ian Horrocks (2013). A Description Logic Primer. https://arxiv.org/abs/1201.4089
- M. Krötzsch, F. Simancik and I. Horrocks, "Description Logics," in IEEE Intelligent Systems, vol. 29, no. 1, pp. 12-19, Jan.-Feb. 2014. doi: 10.1109/MIS.2013.123
- https://www.semantic-mediawiki.org/wiki/Help:User_manual

Additional recommended reading will be announced in the LVA.

Work forms / resources:	Seminar style teaching with practical exercisesAids will be announced in the lecture.
Department:	Computer Science
Division:	Artificial Intelligence
Module responsibility:	Stefan Zander

Simulation of robotic systems

Module numbers:	30.2642 [PVL 30.2643]
Language:	english
Study programme:	Bachelor 2021 - Elective catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KESS 2014 - Catalog ESS: Embedded Systems Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules

	Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007/2004/2002/99 - Elective courses from the computer science area KoSI 2007/2004/2003/2002/99 - Elective courses from the computer science area
Type of course:	V+P = Lecture+Practical
Weekly hours:	2+2
Credit Points:	5
Exam:	written exam
PVL (e.g. Practical):	not graded (ungraded practical exercises)
Requiredknowledge:	Basic bachelor-level programming skills
Learning objectives:	The students get to know the structure and functionality of robot simulation systems. They are able to use such systems appropriately, integrate them in working environments, modify existing systems and develop them further to the given needs.
Content:	Processes and concepts, methodical and practical knowledge for the design, implementation and use of robot simulation systems are conveyed: • Structure of robot systems • Modeling the robot work cell • Modeling the control • Programming in robot simulation systems • Calibration • Collision detection • Outlook for collision-free motion planning
Literature:	K.M. Lynch, F.C. Park: Modern Robotics - Mechanics, Planning, and Control Cambridge University Press, 2017 (English).
Lecture style / Teaching aids:	Seminaristiclecture with laboratory. Resources include lecture notes, example programs and software tools.
Faculty:	Computer Science
Expert group:	Computer engineering
Responsibility:	Thomas Horsch

Robot system simulation

English Title:	Simulation of Robotic Systems
Document numbers:	30.2260 [PVL 30.2261]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KESS 2014 - Catalog ESS: Embedded Systems Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007 - Specialization TI: Computer Engineering Bachelor 2007/2004/2002/99 - Elective courses from the computer science area KoSI 2007 - Specialization TI: Computer Engineering KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science
Teaching form:	V+P = Lecture+Internship

SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded (successful participation in the internship (testat))
Requiredprior knowledge:	Basic knowledge on bachelor level in programming Learning
objectives:	The students know structure and function of robot simulation systems. They can use these systems for their intended purpose, integrate them into working environments, modify existing systems and develop them further as required.
Course content:	Procedures and concepts, methodical and practical knowledge for the design, implementation and use of robot simulation systems are taught. • Robot system structure • Modeling of the robot workcell • Control modeling • Programming in robot simulation systems • Calibration • Collision detection • Outlook Collision-free motion planning
Literature:	W. Weber: Industrieroboter- Methoden der Steuerung und Regelung, Hanser Verlag, 2009
Forms of work / resources:	Seminar-like lecture with computer-aided examples as well as Lecture hall exercises, in the practical course the understanding of the material of the course is supported and deepened with the help of experiments.
Department:	Computer Science
Expert group:	Computer Engineering
Module responsibility:	Thomas Horsch

Social engineering

Document numbers:	30.2622 [PVL 30.2623]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog ITS Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KITS 2014 - Catalog ITS: IT Security Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Teaching form:	V+P = Lecture+Internship
SWS:	3+1
CP:	5
Exam:	Written exam
PVL (e.g. practical course):	ungraded (ungraded practical exercise)
RequiredPrevious knowledge:	Passed LV IT-Sicherheit

Learning Objectives:

Participants will

- understand how social engineering works not only on a technical level, but also on a psychological level.
- know the different phases of a social engineering attack.
- can name, analyze and evaluate different attack vectors.
- are able to recognize social engineering by means of technical and organizational measures and to develop and implement countermeasures.
- can conceptualize and implement security awareness training and evaluate its success.
- Can apply methods from IT auditing to perform appropriate testing for the presence and effectiveness of controls or employee awareness regarding social engineering.

Course content:

Theoretical basis for social engineering (definition, mode of action, human behavior, attack vectors)

- Embedding social engineering into the IT security management of companies (e.g. security incident process)
- Presentation of concrete social engineering attack vectors (e.g. Email phishing, attacks via telephone, distribution of USB sticks, physical access attempts).
- Use of "The Social-Engineer Toolkit" for the implementation of Social Engineering penetration testing
- Measures to detect social engineering (e.g. spam filters, anti-malware, security awareness, authentication)
- Measures to prevent social engineering (e.g. security awareness, technical solutions)
- Approaches for testing security awareness in companies
- Imparting practical experience

Literature:

Kevin D. Mitnick and William L. Simon: "The Art of Deception: Controlling the Human Element of Security"; John Wiley & Sons; 2011.

- Christopher Hadnagy: "Social engineering: The art of human hacking"; John Wiley & Sons; 2010.
- Katharina Krombholz et al: "Advanced social engineering attacks."; Journal of Information Security and applications 22; 2015.
- Matthew Tischer et al: "Users really do plug into USB drives they find."; IEEE Symposium on Security and Privacy; 2016.
- The Social Engineering Framework (https://www.social-engineer.org/)

Forms of work / resources: Seminar lecture (group work and internship integrated),
Lecture notes

Department: Computer Science

Expert group: IT Security

Module responsibility: Christoph Krauß

Software development for embedded systems

English Title: Software Development for Embedded Systems

Document numbers: 83.7418 [PVL 83.7419]

Language: GERMAN

Assignment: Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I

Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KESS 2014 - 4th semester
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded

Required Prior Knowledge: Basic knowledge in the areas of computer engineering, the Programming and the algorithms and data structures.

Learning Objectives: Students will

- can formulate requirements in the embedded systems (ES) environment
- master object-oriented modeling and implementation of ES with efficient use of memory and CPU
- Are capable of using a common operating system with POSIX interface for ES development.
- use processes and threads for concurrent programming and master methods for interprocess communication and synchronization in ES
- can realize the connection of sensors and actuators by means of event-driven software architectures
- know aspects of embedded applications in communication protocols and bus systems and derive from this influences on the system behavior
- formulate requirements of ES for operating systems and know examples of this
- Understand quality assurance methods and influences on maintainability

Course content: Embedded Systems (ES) terms and basic ideas.

- Requirements especially non-functional requirements for ES
- Procedures for model-based design and object-oriented implementation
- Efficient use of system resources such as memory and CPU
- Operating systems for ES especially with POSIX API
- Scheduling and timing of ES, concurrency, processes and threads
- Interprocess communication and synchronization in ES
- Event-driven architectures, state machines
- Aspects of embedded applications in communication protocols and bus systems
- Selected embedded operating systems
- Quality assurance and maintainability

Literature:
Holt, Huang, Embedded Operating Systems - a practical approach, Springer 2014 Werner Zimmermann, Ralf Schmidgall - Bussysteme in der Fahrzeugtechnik, Springer 2014.
Alt, Modellbasierte Systementwicklung mit SysML, Carl Hanser Verlag, 2012
Berns, Schürmann, Trapp, Eingebettete Systeme, Vieweg+Teubner, 2010
Schröder, Gockel, Dillmann, Embedded Linux, Verlag, 2009
Marwedel, Embedded Systems, Springer, 2008
Automotive Embedded Systems; Wietzke, Tran; Springer Verlag, 2005
Corbet, Rubini, Kroah-Hartman, Linux Device Drivers 3rd Edition, O'Reilly, 2005

Forms of work / resources: Seminar lecture and lecture hall exercises, in the practical course the Understanding of the course material deepened while developing a small ES application in multiple iterations.

Department: Computer Science

Expert group: Computer Engineering

Responsible for the module: Jens-Peter Akelbein

Software development for HMI systems

English Title: Software Development for HMI Systems

Document numbers: 30.2600 [PVL 30.2601]

Language: GERMAN

Assignment:
Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KESS 2014 - Catalog ESS: Embedded Systems
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules

Teaching form: V+P = Lecture+Internship

SWS: 2+2

CP: 5

Exam: Written exam

PVL (e.g. internship): ungraded (successful participation in the internship)

Required Prior Knowledge: Basic knowledge in the fields of computer engineering, the Programming and the algorithms and data structures and software development for embedded systems.

Learning objectives: An HMI system (Human Machine Interface) consists of hardware (often embedded hardware with display and touch screen) and individual software for visualization, operation and control of a machine. HMI systems are used, for example, in industrial automation, in infotainment systems (car, aircraft, TV) or in operating devices for the Internet of Things.

The students

- can formulate requirements for the realization of Human Machine Interfaces (HMI) for embedded systems
- master tools for the development of HMI software
- Understand basic architectural patterns in touch applications and be able to use them
- implement a user interface for an embedded system
- can implement requirements and methods of platform-independent software development
- know multitouch operating concepts
- understand methods of quality assurance regarding stability, maintainability, performance and energy efficiency

Course content: Fields of application for Human Machine Interfaces (HMI) in embedded systems in industry and for consumer devices.

- HMI development requirements, project planning and frameworks
- Methods, techniques and tools for planning and realization of embedded HMI
- Architectural pattern of event-driven programming in touch applications

	- Programming graphical applications with Qt and Qt Quick - Operating concepts with multitouch - Platform-independent development and quality assurance for different hardware, form factors and operating systems
Literature:	Current, suitable literature will be announced at the beginning of the course.
Work forms / resources:	Seminar lecture and lecture hall exercises, in the practical course the Understanding of the course material deepened while developing a small HMI application in multiple iterations.
Department:	Computer Science
Expert group:	Computer Engineering
Module responsibility:	Eva Brucherseifer

Software security

English Title:	Software Security
Document numbers:	84.7220 [PVL 84.7221]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KITS 2021 - 4th semester Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KESS 2014 - Catalog ESS: Embedded Systems Bachelor dual KITS 2014 - 2nd semester Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules Bachelor 2007/2004/2002/99 - Electives from the computer science area KoSI 2007/2004/2003/2002/99 - Electives from the computer science area
Teaching form:	V+P = Lecture+Internship
SWS:	3+1
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded (successful participation in the internship)
Required prior knowledge:	Basic knowledge in the areas of programming, algorithms and data structures and IT security.
Learning Objectives:	Students will - are familiar with process models for the development of secure software - can apply methods and tools for software security assessment - can evaluate software designs with regard to security - Are familiar with best practices in the area of software security - can identify and evaluate security requirements for software
Course content:	procedure models for the development of secure software (SSDLC). - Views of customers and attackers (use case, misuse case) - Software security and software design - Modeling, design and analysis of secure IT systems (security engineering) - Safe programming

- Safety certifications and their limits
- Maturity models (OpenSAMM, BSI-MM) and metrics
- Methods and tools for software security assessment
- Safety tests
- Secure delivery and deployment of software (secure deployment)
- Case studies

Literature:	Ross Anderson: Security Engineering, Wirley, 2e, 2008. • Dorothy Denning: Chryptography and Data Security, Addison-Wesley, 1982. • Claudia Eckert: IT-Sicherheit, Oldenbourg, 8e, 2013. • Bruce Schneier: Applied Cryptography, Wiley, 2e,
1996.Forms of work / resources:	Seminar lecture + practical course
Department:	Computer Science
Expert group:	IT Security
Module responsibility:	Oliver Weissmann

Stochastic modeling and simulation

English Title:	Stochastic models and simulation
Document numbers:	30.2654 [PVL 30.2655]
Language:	GERMAN
Assignment:	Bachelor 2021 - Elective Catalog I Bachelor dual KITS 2021 - Elective Catalog I Bachelor dual KoSI 2021 - Elective Catalog I Bachelor KMI 2021 - Elective Catalog I Bachelor 2014 - Catalog I: Application and system-oriented modules Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Teaching form:	V+P = Lecture+Internship
SWS:	2+2
CP:	5
Exam:	Written exam
PVL (e.g. internship):	ungraded(The internships are conducted with statistical software (R). Ungraded testates will be given at the practicum dates. All testates must be available as a prerequisite for the exam).
Prerequisite:	BBPO 2014: Course "Fundamentals of Analysis" successfully completed. BBPO 2021: Course "Mathematics 2" successfully completed.
Learning Objectives:	Students will achieve the following learning objectives: • Mastery of the mathematical basics for stochastic simulations • Understanding/mastery of the central concepts of stochastics. • Description of applications as a stochastic model • Mastery of a statistical programming language • Understanding of random processes The following competencies are taught in the module: • Modeling applications using stochastic terms • Set up (implement) simple simulation models • Visualization of simulation results
Course contents:	Contents of the course are:

- Mathematical foundations of stochastics
- Random variables
- Discrete and continuous distributions
- Conditional distributions
- Modeling of random processes
- Simulation of random processes (simulation methods, Monte Carlo simulation)
- Applications

Literature:

Fahrmeir et al.: Stats.

- Horgan: Probability with R
- Ross: Probability

Forms of work / tools: Thepracticals are carried out with statistical software (R).

Department: Mathematics and Natural Sciences

Module responsibility: Christoph Becker

Unix for Software Developers

Module numbers: 30.2554 [PVL 30.2555]

Language: english

Study programme:

Bachelor 2021 - Elective catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I
Bachelor 2014 - Catalog I: Application and system-oriented modules
Bachelor dual KoSI 2014 - Catalog I: Application and system-oriented modules
Bachelor KMI 2014 - Catalog I: Application and system-oriented modules
Bachelor 2007 - Specialization AE: Application Engineering
Bachelor 2007 - Specialization TI: Computer Engineering
Bachelor 2007/2004/2002/99 - Electives from the Computer Science
Department KoSI 2007 - Specialization AE: Application Engineering
KoSI 2007 - Specialization TI: Computer Engineering
KoSI 2007/2004/2003/2002/99 - Elective courses in the field of computer science

Type of course: V+P = Lecture+Practical

Weekly hours: 2+2

Credit Points: 5

Exam: written exam

PVL (e.g. Practical): not graded (Successful participation in the lab exams)

Required knowledge: Basic knowledge in programming and operating systems courses (Bachelors)

Learningobjectives:

The students should

- understand concepts, terms and correlations about software development in the Unix environment and related systems (including Linux).
- work with Unix and be able to solve software development tasks.
- learn to administer Unix systems.
- Get to know the capabilities of different Unix tools and be able to use these tools on their own.

Content:

Overview of Unix

- Linux file systems and process concept
- Commands and system administration tools
- Shell and shell programming
- terminal administration

- system programming in Unix
- Security aspects of current Linux distributions
- Selected topics for current Linux distributions

Literature: W.R. Stevens; Advanced Programming in the UNIX Environment; W.R. Stevens; Addison-Wesley; 2005.

- Bolsky/Korn; The KornShell; Hanser; 1991
- J. Christ; TerminalBuch vi; Oldenbourg; 1989
- T. Klein; Buffer Overflows and Format String Vulnerabilities; dpunkt.verlag; 2003

Lecture style / Teaching aids: Lecture notes

Faculty: Computer

Science

Expert group: Operating systems / distributed systems

Responsibility: Benedict Reuschling

Preparatory Seminar Bachelor Thesis

English Title: Preparation Seminar for the Bachelor Thesis

Document number: 30.2672

Language: GERMAN

Assignment: Bachelor 2021 - Elective Catalog I
Bachelor dual KITS 2021 - Elective Catalog I
Bachelor dual KoSI 2021 - Elective Catalog I
Bachelor KMI 2021 - Elective Catalog I

Teaching Form: S = Seminar

SWS: 2

CP: 2.5

Examination: Written elaboration in the form of a term paper (70%-part) and plenary presentation (30%-part).

Required prior knowledge: Module "Scientific work

Learning objectives: Students deepen central concepts of scientific work and write them down in an own scientific elaboration on a self-selected topic as preparation for the bachelor thesis.

Specifically, the course enables students to,

- name the essential characteristics of scientific work and apply them to one's own work
- present the most important key points of a self-selected topic in the form of a synopsis
- operationalize a self-selected topic and elaborate it according to the principles of good scientific practice
- Recognize plagiarism and avoid it in your own work
- to give a science-related / scientific technical lecture
- peer-review other scientific papers
- Identify good and bad aspects in academic papers and transfer them to your own final paper
- Conduct an independent literature search on the state of the art of a topic area and perform a source-critical evaluation of the relevant literature
- develop a suitable methodological approach for the own project

and critically evaluate.

Upon completion of the module, students will be familiar with the basic concepts and methods of scientific work as they apply to the preparation of the bachelor's thesis.

Course content:

The module content is based on the competencies required for the preparation of a scientific thesis.

- Essential characteristics of scientific/scientific work and scientific work
- Description of the own project in the form of an exposé
- How does a good abstract succeed?
- Operationalization of the own project (delimitation of the problem and derivation of research questions)
- Procedure for setting up and carrying out one's own science-related work (topic search, literature research, operationalization, conceptualization, formalization, evaluation, discussion, reflection, etc.).
- The selection of appropriate research methodologies
- Slipping into an expert role and writing an expert report
- Conducting a science-related presentation
- Writing of the own project in the form of a scientific paper under consideration of the principles of scientific honesty
- Discussion of the strengths and weaknesses of selected bachelor theses

Literature:

Wayne Booth et al. The Craft of Research, University of Chicago Press, 3e, 2008.

- Justin Zobel, Writing for Computer Science, Springer; 2e, 2004.
- Matthias Karmasin, Rainer Ribing, Die Gestaltung wissenschaftlicher Arbeiten: A Guide for Seminar Papers, Bachelor, Master and Magister Theses and Dissertations, UTB, 2012.
- Norbert Frank, Joachim Stary, The Technique of Scientific Work, UTB, 2011
- Helmut Balzert et. al, Scientific Work - Science, Sources, Artifacts, Organization, Presentation, W3l, 2008.

Additional literature will be announced in the course Work

forms / resources:

Seminar materials,technical articles, textbooks, scientific Research portals etc.

Department: Computer Science

Division: Social and Cultural Aspects of Computer Science

Module responsibility: Stefan Zander