

MODULE HANDBOOK

Bachelor of Science

Bachelor Computer Science (FS-FI-BACS)

180 ECTS

Distance Learning

Classification: Undergraduate

Contents

1. Semester

Module DLBCSICS: Introduction to Computer Science

Module Description	11
Course DLBCSICS01: Introduction to Computer Science	13

Module DLBCSIAW: Introduction to Academic Work

Module Description	17
Course DLBCSIAW01: Introduction to Academic Work	19

Module DLBCSM1: Mathematics I

Module Description	23
Course DLBCSM101: Mathematics I	25

Module DLBCSOOPJ: Object-oriented Programming with Java

Module Description	29
Course DLBCSOOPJ01: Object-oriented Programming with Java	31

Module DLBCSDSJCL: Data structures and Java class library

Module Description	35
Course DLBCSDSJCL01: Data structures and Java class library	37

Module DLBCSIDM: Intercultural and Ethical Decision-Making

Module Description	41
Course DLBCSIDM01: Intercultural and Ethical Decision-Making	43

2. Semester

Module DLBCSM2: Mathematics II

Module Description	51
Course DLBCSM201: Mathematics II	53

Module DLBCSWAD: Web Application Development

Module Description	57
Course DLBCSWAD01: Web Application Development	59

Module DLBCSCW: Collaborative Work

Module Description	63
Course DLBCSCW01: Collaborative Work	65

Module DLBDSSPDS: Statistics: Probability and Descriptive Statistics

Module Description	69
Course DLBDSSPDS01: Statistics: Probability and Descriptive Statistics	71

Module DLBCSCAOS: Computer Architecture and Operating Systems

Module Description	75
Course DLBCSCAOS01: Computer Architecture and Operating Systems	77

Module DLBCSPJWD: Project: Java and Web Development

Module Description	81
Course DLBCSPJWD01: Project: Java and Web Development	83

3. Semester**Module DLBCSDMDS: Database Modeling and Database Systems**

Module Description	89
Course DLBCSDMDS01: Database Modeling and Database Systems	91

Module DLBDSPBDM: Project: Build a Data Mart in SQL

Module Description	95
Course DLBDSPBDM01: Project: Build a Data Mart in SQL	97

Module DLBCSRE: Requirements Engineering

Module Description	101
Course DLBCSRE01: Requirements Engineering	103

Module DLBCSCNDS: Computer Networks and Distributed Systems

Module Description	107
Course DLBCSCNDS01: Computer Networks and Distributed Systems	109

Module DLBCSL: Algorithms, Data Structures, and Programming Languages

Module Description	113
Course DLBCSL01: Algorithms, Data Structures, and Programming Languages	115

Module DLBCSITSM: IT-Service Management

Module Description	119
Course DLBCSITSM01: IT-Service Management	121

4. Semester**Module DLBCSPITSM: Project: IT Service Management**

Module Description	129
Course DLBCSPITSM01: Project: IT Service Management	131

Module DLBCSTCSML: Theoretical Computer Science and Mathematical Logic

Module Description	133
Course DLBCSTCSML01: Theoretical Computer Science and Mathematical Logic	135

Module DLBDSIPWP: Introduction to Programming with Python

Module Description	139
Course DLBDSIPWP01: Introduction to Programming with Python	141

Module DLBCSSQA: Software Quality Assurance

Module Description	145
Course DLBCSSQA01: Software Quality Assurance	147

Module DLBCSS: Specification

Module Description	151
Course DLBCSS01: Specification	153

Module DLBCSPSE: Project: Software Engineering

Module Description	157
Course DLBCSPSE01: Project: Software Engineering	159

5. Semester**Module DLBCSSCTCS: Seminar: Current Topics in Computer Science**

Module Description	165
Course DLBCSSCTCS01: Seminar: Current Topics in Computer Science	167

Module DLBCSIDPITS: Introduction to Data Protection and Cyber Security

Module Description	171
Course DLBCSIDPITS01: Introduction to Data Protection and Cyber Security	173

Module DLBCSCT: Cryptography

Module Description	177
Course DLBCSCT01: Cryptography	179

Module DLSFPD: Salesforce Platform Development

Module Description	183
Course DLSFPD01: Salesforce Platform App Builder	185
Course DLSFPD02: Salesforce Platform Developer	188

Module DLBCSEMSE: Mobile Software Engineering

Module Description	191
Course DLBCSEMSE01: Mobile Software Engineering I	193
Course DLBCSEMSE02: Mobile Software Engineering II	196

Module DLBCSEBDCT: Big Data and Cloud Technologies

Module Description	199
Course DLBDSBDT01: Big Data Technologies	201
Course DLBDSCC01: Cloud Computing	204

Module DLBCSEBI: Business Intelligence

Module Description	207
Course DLBCSEBI01: Business Intelligence	209
Course DLBCSEBI02: Project: Business Intelligence	213

Module DLBCSESEWP: Software Engineering with Python

Module Description	215
Course DLBDSOOFPP01: Object oriented and functional programming in Python	217
Course DLBDSDSSE01: Data Science Software Engineering	220

Module DLBCSEITPAM: IT project and architecture management

Module Description	223
Course DLBCSEITPAM01: IT Project Management	226
Course DLBCSEITPAM02: IT Architecture Management	230

6. Semester**Module DLSFPD: Salesforce Platform Development**

Module Description	237
Course DLSFPD01: Salesforce Platform App Builder	239
Course DLSFPD02: Salesforce Platform Developer	242

Module DLBCSEMSE: Mobile Software Engineering

Module Description	245
Course DLBCSEMSE01: Mobile Software Engineering I	247
Course DLBCSEMSE02: Mobile Software Engineering II	250

Module DLBCSEBDCT: Big Data and Cloud Technologies

Module Description	253
Course DLBDSBDT01: Big Data Technologies	255
Course DLBDSCC01: Cloud Computing	258

Module DLBCSEBI: Business Intelligence

Module Description	261
Course DLBCSEBI01: Business Intelligence	263
Course DLBCSEBI02: Project: Business Intelligence	267

Module DLBCSESEWP: Software Engineering with Python

Module Description	269

Course DLBDSOOFPP01: Object oriented and functional programming in Python	271
Course DLBDSDSSE01: Data Science Software Engineering	274

Module DLBCSEITPAM: IT project and architecture management

Module Description	277
Course DLBCSEITPAM01: IT Project Management	280
Course DLBCSEITPAM02: IT Architecture Management	284

Module DLBCSAPM: Agile Project Management

Module Description	287
Course DLBCSAPM01: Agile Project Management	289

Module DLBCSIITL: IT Law

Module Description	293
Course DLBCSIITL01: International IT Law	295

Module DLBCSCSAS: Computer Science and Society

Module Description	299
Course DLBCSCSAS01: Computer Science and Society	301

Module DLBBT: Bachelor Thesis

Module Description	305
Course DLBBT01: Bachelor Thesis	307
Course DLBBT02: Colloquium	310

2020-12-01

1. Semester

Introduction to Computer Science

Module Code: DLBCSICS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Kamal Bhattacharya (Introduction to Computer Science)

Contributing Courses to Module

- Introduction to Computer Science (DLBCSICS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Information representation
- Algorithms and data structures
- Propositional logic / Boolean algebra
- Hardware
- Networks and the internet
- Software
- Computer science as a discipline

Learning Outcomes**Introduction to Computer Science**

On successful completion, students will be able to

- understand basic algorithms and data structures.
- apply basic constructs of propositional logic in programming.
- describe the structure of computer hardware systems.
- specify the structure and the main services of the internet.
- discuss professional conduct in computer science.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Introduction to Computer Science

Course Code: DLBCSICS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The goal of this course is to provide an introduction to computer science and its main concepts. It covers basic topics such as information representation and an introduction to algorithms and data structures. Propositional logic and Boolean algebra are also introduced, both of which form an important basis in computer science, e.g., for expressing conditions in programming. Furthermore, the course introduces the three main components of computing infrastructures: hardware, networks, and software. Finally, the course covers the meta level by looking at the role of computer science as a discipline as well as ethics and professional conduct.

Course Outcomes

On successful completion, students will be able to

- understand basic algorithms and data structures.
- apply basic constructs of propositional logic in programming.
- describe the structure of computer hardware systems.
- specify the structure and the main services of the internet.
- discuss professional conduct in computer science.

Contents

1. Basic concepts of data processing
 - 1.1 Data, information and messages
 - 1.2 Software, firmware and hardware
 - 1.3 Languages, syntax and semantics
 - 1.4 Historical overview
2. Information representation
 - 2.1 Number representation formats
 - 2.2 Representation of non-numerical information
 - 2.3 Data types
 - 2.4 Redundancy and error tolerance

3. Algorithms and data structures
 - 3.1 Algorithms and flow diagrams
 - 3.2 Simple data structures
 - 3.3 Searching and sorting
 - 3.4 Quality of algorithms (correctness, termination, efficiency/complexity)
4. Propositional logic, Boolean algebra and circuit design
 - 4.1 Propositions and logical conclusions
 - 4.2 Conjunctive and disjunctive normal form
 - 4.3 Digital circuit design
5. Hardware and computer architectures
 - 5.1 Computer types and their architecture
 - 5.2 Processors and memory
 - 5.3 Input and output
 - 5.4 Interfaces and drivers
 - 5.5 High-performance computing
6. Networks and the internet
 - 6.1 Wired and wireless networks and their topologies
 - 6.2 The TCP/IP and the ISO/OSI model
 - 6.3 Internet structure and services
 - 6.4 The internet of things
7. Software
 - 7.1 BIOS and operating systems
 - 7.2 Application software and information systems
 - 7.3 Apps
 - 7.4 Embedded systems
 - 7.5 Software development
8. Computer Science as a discipline
 - 8.1 The role and sub-disciplines of computer science
 - 8.2 Artificial intelligence, data science and computer science
 - 8.3 Ethical aspects of computer science
 - 8.4 The ACM Code of Ethics and Professional Conduct

Literature**Compulsory Reading****Further Reading**

- Association for Computing Machinery (ACM). (2018). ACM code of ethics and professional conduct. Retrieved from <https://www.acm.org/code-of-ethics>
- Brookshear, G., & Bylow, D. (2014). Computer science: An overview (12th ed.). Boston, MA: Pearson.
- Dewdney, A. K. (2001). The new turing omnibus. London: Macmillan Education.
- Gruhn, V., & Striemer, R. (Eds.). (2018). The essence of software engineering. Cham: Springer.
- Sedgewick, R., & Wayne, K. (2016). Computer science: An interdisciplinary approach. Boston, MA: Addison-Wesley.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Introduction to Academic Work

Module Code: DLBCSIAW

Module Type	Admission Requirements	Study Level	CP	Student Workload
s. Curriculum/see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	
s. Curriculum/see curriculum	Minimum 1 semester	WiSe/SoSe	

Module Coordinator

Prof. Dr. Maya Stagge (Introduction to Academic Work)

Contributing Courses to Module

- Introduction to Academic Work (DLBCSIAW01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Workbook

Split Exam

Weight of Module

s. Curriculum/see curriculum

Module Contents

- Scientific Theoretical Foundations and Research Paradigms
- Application of Good Scientific Practice
- Methodology
- Librarianship: Structure, Use, and Literature Management
- Forms of Scientific Work at IUBH

Learning Outcomes

Introduction to Academic Work

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,/On successful completion, students will be able to

- understand and apply formal criteria of a scientific work.
- distinguish basic research methods and identify criteria of good scientific practice.
- describe central scientific theoretical basics and research paradigms and their effects on scientific research results.
- use literature databases, literature administration programs, and other library structures properly; avoid plagiarism; and apply citation styles correctly.
- apply the evidence criteria to scientific texts.
- define a research topic and derive a structure for scientific texts.
- compile a list of literature, illustrations, tables, and abbreviations for scientific texts.
- understand and distinguish between the different forms of scientific work at IUBH.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Methods

Links to other Study Programs of IUBH

All Bachelor Programmes in the Business & Management field

Introduction to Academic Work

Course Code: DLBCSIAW01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The application of good scientific practice is one of the basic academic qualifications that should be acquired while studying. This course deals with the distinction between everyday knowledge and science. This requires a deeper understanding of the theory of science, as well as the knowledge of basic research methods and instruments for writing scientific texts. The students therefore gain initial insight into academic research and are introduced to the basic knowledge that will help them in the future to produce scientific papers. In addition, the students receive an overview of the different IUBH examination forms and insight into their requirements and implementation.

Course Outcomes

Nach erfolgreichem Abschluss sind die Studierenden in der Lage,/On successful completion, students will be able to

- understand and apply formal criteria of a scientific work.
- distinguish basic research methods and identify criteria of good scientific practice.
- describe central scientific theoretical basics and research paradigms and their effects on scientific research results.
- use literature databases, literature administration programs, and other library structures properly; avoid plagiarism; and apply citation styles correctly.
- apply the evidence criteria to scientific texts.
- define a research topic and derive a structure for scientific texts.
- compile a list of literature, illustrations, tables, and abbreviations for scientific texts.
- understand and distinguish between the different forms of scientific work at IUBH.

Contents

1. Theory of Science
 - 1.1 Introduction to Science and Research
 - 1.2 Research Paradigms
 - 1.3 Fundamental Research Decisions
 - 1.4 Effects of Scientific Paradigms on Research Design

2. Application of Good Scientific Practice
 - 2.1 Research Ethics
 - 2.2 Evidence Teaching
 - 2.3 Data Protection and Affidavit
 - 2.4 Orthography and Shape
 - 2.5 Identification and Delimitation of Topics
 - 2.6 Research Questions and Structure
3. Research Methods
 - 3.1 Empirical Research
 - 3.2 Literature and Reviews
 - 3.3 Quantitative Data Collection
 - 3.4 Qualitative Data Collection
 - 3.5 Mix of Methods
 - 3.6 Critique of Methods and Self-Reflection
4. Librarianship: Structure, Use, and Literature Management
 - 4.1 Plagiarism Prevention
 - 4.2 Database Research
 - 4.3 Literature Administration
 - 4.4 4.4 Citation and Author Guidelines
 - 4.5 4.5 Bibliography
5. Scientific Work at the IUBH – Research Essay
6. Scientific Work at the IUBH - Project Report
7. Scientific Work at the IUBH - Case Study
8. Scientific Work at the IUBH - Bachelor Thesis
9. Scientific Work at the IUBH – Oral Assignment
10. Scientific Work at the IUBH – Oral Project Report
11. Scientific Work at the IUBH - Colloquium
12. Scientific Work at the IUBH - Portfolio
13. Scientific Work at the IUBH - Exam

Literature**Compulsory Reading****Further Reading**

- Bortz, J./Döring, N. (2012): Forschungsmethoden und Evaluation. Für Human- und Sozialwissenschaftler. 5. Auflage, Springer Medizin Verlag, Heidelberg.
- Braunecker, C. (2016): How to do Empirie, how to do SPSS – eine Gebrauchsanleitung. Facultas Verlags- und Buchhandels AG, Wien.
- Engelen, E.M. et al. (2010): Heureka – Evidenzkriterien in den Wissenschaften, ein Kompendium für den interdisziplinären Gebrauch. Spektrum akademischer Verlag, Heidelberg.
- Flick, U. et al. (2012): Handbuch Qualitative Sozialforschung. Grundlagen, Konzepte, Methoden und Anwendungen. 3. Auflage, Beltz Verlag, Weinheim.
- Hug, T./Poscheschnik, G. (2015): Empirisch Forschen, 2. Auflage, Verlag Huter & Roth KG, Wien.
- Hussy, W. et al. (2013): Forschungsmethoden in Psychologie und Sozialwissenschaften. 2. Auflage, Springer Medizin Verlag, Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Workbook

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

Mathematics I

Module Code: DLBCSM1

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Mathematics I)

Contributing Courses to Module

- Mathematics I (DLBCSM101)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Basic definitions and terms of discrete mathematics
- Sets and propositional logic
- Number systems such as decimal and binary systems
- Graphs and mappings
- Selected topics of elementary number theory
- Cryptography

Learning Outcomes**Mathematics I**

On successful completion, students will be able to

- understand basic terms of discrete mathematics as well as describe them and distinguish them from each other.
- understand concepts of number theory and their application in IT and technology and be able to solve tasks independently by applying these concepts.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Methods.

Links to other Study Programs of IUBH

All Bachelor Programmes in the Business & Management field(s).

Mathematics I

Course Code: DLBCSM101

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Many practical concepts in IT and technology are based on the findings of discrete mathematics. For an in-depth understanding of, for example, data structures, the construction of communication networks, or of solutions to algorithmic problems, a basic understanding of their mathematical background is necessary. This course therefore introduces discrete mathematical terms and concepts, with specific areas of number theory also taught.

Course Outcomes

On successful completion, students will be able to

- understand basic terms of discrete mathematics as well as describe them and distinguish them from each other.
- understand concepts of number theory and their application in IT and technology and be able to solve tasks independently by applying these concepts.

Contents

1. Mathematical Basics
 - 1.1 Basic Concepts
 - 1.2 Proof Techniques
 - 1.3 Finite Sums
2. Sets
 - 2.1 Properties and Calculation Rules for Sets
 - 2.2 Equivalence Relations
3. Propositional Logic
 - 3.1 Statements and Logical Connections
 - 3.2 Truth Tables
 - 3.3 Computational Rules of Propositional Logic
 - 3.4 Simplification of Expressions in Propositional Logic

4. Number Systems
 - 4.1 Decimal System
 - 4.2 Binary System
 - 4.3 Hexadecimal System
5. Mappings
 - 5.1 Mappings and Graphs
 - 5.2 Special Properties of Mappings
6. Basic Algebraic Structures
 - 6.1 Groups
 - 6.2 Rings
 - 6.3 Residual Class Rings
7. Prime Numbers
 - 7.1 Definition and Properties of Prime Numbers
 - 7.2 Prime Number Test
8. Modular Arithmetic
 - 8.1 The Euclidean Algorithm
 - 8.2 Fundamental Theorem of Arithmetic
9. Applications in Cryptography
 - 9.1 The Shift Cryptosystem
 - 9.2 Symmetric vs Asymmetric Cryptosystems
 - 9.3 The RSA Cryptosystem

Literature**Compulsory Reading****Further Reading**

- Beutelspacher, A. (2003): Lineare Algebra. Eine Einführung in die Wissenschaft der Vektoren, Abbildungen und Matrizen. 6. Auflage, Vieweg.
- Beutelspacher, A. (2009): Kryptologie. Eine Einführung in die Wissenschaft vom Verschlüsseln, Verbergen und Verheimlichen. 9. Auflage, Vieweg+Teubner.
- Drews, R./Scholl, W. (2001): Handbuch der Mathematik. Orbis.
- Forster, O. (2001): Analysis 1. 6. Auflage, Vieweg.
- Hoffmann, U. (2005): Mathematik für Wirtschaftsinformatiker. Übungen mit Lösungen. (URL: http://opus.uni-lueneburg.de/opus/volltexte/2006/383/pdf/Uebungen_zur_Mathematik_fuer_Wirtschaftsinformatiker.pdf [letzter Zugriff: 27.02.2017]).
- Teschl, G./Teschl, S. (2013): Diskrete Mathematik und lineare Algebra. 4. Auflage, Springer Vieweg, Berlin, Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Object-oriented Programming with Java

Module Code: DLBCSOOPJ

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Damir Ismailovic (Object-oriented Programming with Java)

Contributing Courses to Module

- Object-oriented Programming with Java (DLBCSOOPJ01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Introduction to the Java language
- Java language constructs
- Introduction to object-oriented system development
- Inheritance
- Object-oriented concepts
- Exception handling
- Interfaces

Learning Outcomes**Object-oriented Programming with Java**

On successful completion, students will be able to

- describe the basic concepts of object-oriented modeling and programming, distinguishing them from one another.
- describe the basic concepts and elements of the Java programming language and have some experience in their use.
- independently create Java programs to solve concrete problems.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Object-oriented Programming with Java

Course Code: DLBCSOOPJ01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Operational information systems are usually planned and programmed to be object-oriented. Therefore, this course teaches the basic skills of object-oriented programming. Theoretical concepts are presented and practiced directly with the programming language Java.

Course Outcomes

On successful completion, students will be able to

- describe the basic concepts of object-oriented modeling and programming, distinguishing them from one another.
- describe the basic concepts and elements of the Java programming language and have some experience in their use.
- independently create Java programs to solve concrete problems.

Contents

1. Introduction to Object-Oriented System Development
 - 1.1 Object Orientation as a Way of Looking at Complex Systems
 - 1.2 The Object as a Basic Concept of Object Orientation
 - 1.3 Phases in the Object-Oriented Development Process
 - 1.4 Basic Principle of Object-Oriented System Development
2. Introduction to Object-Oriented Modeling
 - 2.1 Structuring Problems With Classes
 - 2.2 Identifying Classes
 - 2.3 Attributes as Properties of Classes
 - 2.4 Methods as Functions of Classes
 - 2.5 Associations between Classes
 - 2.6 Unified Modeling Language (UML)

3. Programming Classes in Java
 - 3.1 Introduction to the Java Programming Language
 - 3.2 Basic Elements of a Class in Java
 - 3.3 Attributes in Java
 - 3.4 Methods in Java
 - 3.5 Main Method: Starting Point of a Java Program
4. Java Language Constructs
 - 4.1 Primitive Data Types
 - 4.2 Variables
 - 4.3 Operators and Expressions
 - 4.4 Control Structures
 - 4.5 Packages and Visibility Modifiers .
5. Inheritance
 - 5.1 Modeling and Inheritance in the Class Diagram
 - 5.2 Programming Inheritance in Java
6. Important Object-Oriented Concepts
 - 6.1 Abstract Classes
 - 6.2 Polymorphism
 - 6.3 Static Attributes and Methods
7. Constructors for Generating Objects
 - 7.1 The Standard Constructor
 - 7.2 Overloading Constructors
 - 7.3 Constructors and Inheritance
8. Handling Exceptions with Exceptions
 - 8.1 Typical Scenarios of Exception Handling
 - 8.2 Standard Exceptions in Java
 - 8.3 Defining Your Own Exceptions
9. Programming Interfaces with Interfaces
 - 9.1 Typical Scenarios of Programming Interfaces
 - 9.2 Interfaces as Programming Interfaces in Java

Literature**Compulsory Reading****Further Reading**

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

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Data structures and Java class library

Module Code: DLBCSDSJCL

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Damir Ismailovic (Data structures and Java class library)

Contributing Courses to Module

- Data structures and Java class library (DLBCSDSJCL01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Programming style
- Working with objects
- External packages and libraries
- Data structures
- Strings and calendar
- File system and data streams

Learning Outcomes**Data structures and Java class library**

On successful completion, students will be able to

- understand typical data structures and distinguish them from each other.
- independently create solutions in the Java programming language using the data structures.
- understand scenarios and strategies for comparing objects and implement them in Java.
- describe the possible uses and functions of character strings and calendar objects in Java and have experience using them.
- describe the possible uses and functions of streams in Java and have experience using them.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology fields

Data structures and Java class library

Course Code: DLBCSDSJCL01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Based on the contents of the course "Basics of object-oriented programming with Java", this course deepens the knowledge of object-oriented programming. In particular, data structures, their use cases, and their implementation in the Java language are considered. In addition, strategies and scenarios of object comparisons, the use of functions of the "String" data type, the use of calendar objects, and the use of streams are taught.

Course Outcomes

On successful completion, students will be able to

- understand typical data structures and distinguish them from each other.
- independently create solutions in the Java programming language using the data structures.
- understand scenarios and strategies for comparing objects and implement them in Java.
- describe the possible uses and functions of character strings and calendar objects in Java and have experience using them.
- describe the possible uses and functions of streams in Java and have experience using them.

Contents

1. Programming Style
 - 1.1 Code Documentation
 - 1.2 Code Annotations
 - 1.3 Code Conventions
2. Working with Objects
 - 2.1 String Representation of Objects
 - 2.2 Compare with ==
 - 2.3 Compare with Equals()
 - 2.4 Compare by hashCode()
 - 2.5 CompareTo()
 - 2.6 Cloning Objects
3. External Packages and Libraries
 - 3.1 Importing Packages
 - 3.2 The Java Class Library

4. Data Structures
 - 4.1 Arrays
 - 4.2 Collections
 - 4.3 Working with Collections
 - 4.4 Lists
 - 4.5 Quantities (Sets)
 - 4.6 Associative Memory (Maps)
 - 4.7 Stacks (Basement)
 - 4.8 Queues (Snakes)
5. Strings and Calendar
 - 5.1 Strings
 - 5.2 StringBuffer
 - 5.3 Splitting Character Strings
 - 5.4 Date and time
 - 5.5 Calendar
6. File System and Data Streams
 - 6.1 Working with the File System
 - 6.2 Working with Files

Literature

Compulsory Reading

Further Reading

- Bloch, J. (2017): Effective Java. 3. Auflage, Addison-Wesley Professional, Boston.
- Bruegge, B./Dutoit, A.H (2013): Object-Oriented Software Engineering Using UML, Patterns, and Java: Pearson New International Edition. Pearson, London.
- Martin, R. (2009): Clean Code - Refactoring, Patterns, Testen und Techniken für sauberen Code: Deutsche Ausgabe. MITP Verlag, Heidelberg.
- McLaughlin, B./Pollice, G./West, D. (2006): Head First Object-Oriented Analysis and Design. O'Reilly, Heidelberg.
- Oracle (2018): The Java Language Specification. Java SE 11 Edition. (URL: <https://docs.oracle.com/javase/specs/jls/se11/jls11.pdf> [letzter Zugriff: 03.09.2019]).
- Seidl, M. (2015): UML @ Classroom. An Introduction to Object-Oriented Modeling, Springer, Berlin.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☐ Audio ☒ Exam Template	☒ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

DLBCSDSJCL01

Intercultural and Ethical Decision-Making

Module Code: DLBCSIDM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Jürgen Matthias Seeler (Intercultural and Ethical Decision-Making)

Contributing Courses to Module

- Intercultural and Ethical Decision-Making (DLBCSIDM01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Case Study

Split Exam

Weight of Module

see curriculum

Module Contents

- Basics of Intercultural Competence
- Cultural Concepts
- Culture and Ethics
- Implications of Current Ethical Problems in the Area of Interculturality, Ethics, and Diversity
- Intercultural Learning and Working
- Case Studies for Cultural and Ethical Conflicts

Learning Outcomes**Intercultural and Ethical Decision-Making**

On successful completion, students will be able to

- explain the most important terms in the areas of interculturality, diversity, and ethics.
- distinguish different explanatory patterns of culture.
- understand culture at different levels.
- plan processes of intercultural learning and working.
- understand the interdependencies of culture and ethics.
- independently work on a case study on intercultural competence.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Business Administration & Management

Links to other Study Programs of IUBH

All Bachelor Programs in the Business & Management fields

Intercultural and Ethical Decision-Making

Course Code: DLBCSIDM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In this course, students acquire the necessary knowledge to understand intercultural competencies and current developments in the fields of diversity and ethics. Students will understand how to systematically plan and implement learning processes for the development of competences important in these areas. First, important terms are clarified and differentiated from each other, and cultural aspects are explained from different perspectives. In addition, students learn that cultural issues are relevant at different levels, for example, within a state, company, or other group. In this context, students also recognize the connection between ethics and culture with different interdependencies. On the basis of this knowledge, students are then familiarized with the different possibilities and potentials of intercultural and ethical learning and working. Practical cases are used to illustrate the importance of the relationships learned for today's work context in many companies. The students then work on a case study in which the acquired knowledge is systematically applied.

Course Outcomes

On successful completion, students will be able to

- explain the most important terms in the areas of interculturality, diversity, and ethics.
- distinguish different explanatory patterns of culture.
- understand culture at different levels.
- plan processes of intercultural learning and working.
- understand the interdependencies of culture and ethics.
- independently work on a case study on intercultural competence.

Contents

1. Basics of Intercultural and Ethical Competence to Act
 - 1.1 Subject Areas, Terms, and Definitions
 - 1.2 Relevance of Intercultural and Ethical Action
 - 1.3 Intercultural Action - Diversity, Globalization, Ethics
2. Cultural Concepts
 - 2.1 Hofstede's Cultural Dimensions
 - 2.2 Culture Differentiation According to Hall
 - 2.3 Locus of Control Concept to Rotter

3. Culture and Ethics
 - 3.1 Ethics - Basic Terms and Concepts
 - 3.2 Interdependence of Culture and Ethics
 - 3.3 Ethical Concepts in Different Regions of the World
4. Current Topics in the Area of Interculturality, Ethics, and Diversity
 - 4.1 Digital Ethics
 - 4.2 Equality and Equal Opportunities
 - 4.3 Social Diversity
5. Intercultural Learning and Working
 - 5.1 Acculturation
 - 5.2 Learning and Working in Intercultural Groups
 - 5.3 Strategies for Dealing with Cultural Conflicts
6. Case Studies for Cultural and Ethical Conflicts
 - 6.1 Case Study: Interculturality
 - 6.2 Case Study: Diversity
 - 6.3 Case Study: Interculturality and Ethics

Literature

Compulsory Reading

Further Reading

- Emrich, C. (2011): Interkulturelles Management: Erfolgsfaktoren im globalen Business. Kohlhammer-Verlag, Stuttgart/Berlin/Köln.
- Erll, A./Gymnich, M. (2015): Uni-Wissen Interkulturelle Kompetenzen: Erfolgreich kommunizieren zwischen den Kulturen – Kernkompetenzen. 4. Auflage, Klett Lerntraining, Stuttgart.
- Eß, O. (2010): Das Andere lehren: Handbuch zur Lehre Interkultureller Handlungskompetenz. Waxmann Verlag, Münster.
- Hofstede, G./ Hofstede, G. J./Minkov, M. (2017): Lokales Denken, globales Handeln Interkulturelle Zusammenarbeit und globales Management. 6. Auflage, Beck, München.
- Leenen, W.R./Groß, A. (2018): Handbuch Methoden Interkultureller Bildung und Weiterbildung. Verlag Vandenhoeck & Ruprecht, Göttingen.
- Thomas, A. (2011): Interkulturelle Handlungskompetenz. Versiert, angemessen und erfolgreich im internationalen Geschäft. Gabler-Verlag, Wiesbaden.

Study Format Distance Learning

Study Format Distance Learning	Course Type Case Study

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Written Assessment: Case Study

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

DLBCSIDM01

2. Semester

Mathematics II

Module Code: DLBCSM2

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Leonardo Riccardi (Mathematics II)

Contributing Courses to Module

- Mathematics II (DLBCSM201)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Selected topics of linear algebra
- Selected chapters on graphs and algorithms

Learning Outcomes**Mathematics II**

On successful completion, students will be able to

- understand basic concepts of linear algebra, their interrelations, and their application in IT and technology and be able solve tasks independently using these concepts.
- understand and distinguish the basic concepts and important algorithms for graphs and trees from the field of discrete mathematics as well as their application in IT and technology.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Methods.

Links to other Study Programs of IUBH

All Bachelor Programmes in the Business & Management field(s).

Mathematics II

Course Code: DLBCSM201

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course continues the introduction to topics of discrete mathematics which began in the module "Mathematics Fundamentals I". In this course, the concepts of linear algebra are introduced and knowledge about graphs and algorithms for graphs is deepened. Typical questions of applied computer science are selected, and students are shown how they can be solved with graphs.

Course Outcomes

On successful completion, students will be able to

- understand basic concepts of linear algebra, their interrelations, and their application in IT and technology and be able solve tasks independently using these concepts.
- understand and distinguish the basic concepts and important algorithms for graphs and trees from the field of discrete mathematics as well as their application in IT and technology.

Contents

1. Introduction to Matrices
 - 1.1 Basic Concepts of Matrices
 - 1.2 Addition of Matrices
 - 1.3 Scalar Multiplication and Product
2. Inverting Matrices
 - 2.1 Multiplication of Matrices
 - 2.2 Properties of Matrix Multiplication
 - 2.3 Inverse Matrices
3. Linear Systems of Equations
 - 3.1 Gauss Algorithm
 - 3.2 Example Applications of the Gaussian Algorithm
4. Introduction to Graphs
 - 4.1 Undirected Graphs
 - 4.2 Further Properties of Graphs
 - 4.3 Adjacency Matrix

5. The Problem of the Shortest Routes
 - 5.1 Directional Graph or Digraph
 - 5.2 Weighted Graph
 - 5.3 Dijkstra's Algorithm
6. The Königsberg Bridge Problem
 - 6.1 Routing in Graphs
 - 6.2 Eulerian Graph
 - 6.3 Hierholzer's Algorithm
 - 6.4 The Postman Problem
7. A City Tour Where Each City is Visited Exactly Once.
 - 7.1 Special Graphs
 - 7.2 Hamiltonian Graph
 - 7.3 The Ore and Dirac Condition
 - 7.4 The Problem of the Traveling Salesman
8. Trees
 - 8.1 Properties of Trees
 - 8.2 Root Tree
 - 8.3 Spanning Tree
 - 8.4 Minimal Spanning Tree

Literature**Compulsory Reading****Further Reading**

- Hartmann, P. (2014): Mathematik für Informatiker. Ein praxisbezogenes Lehrbuch. 6. Auflage, Springer Vieweg, Wiesbaden.
- Hoffmann, U. (2005): Mathematik für Wirtschaftsinformatiker. Übungen mit Lösungen. (URL: http://opus.uni-lueneburg.de/opus/volltexte/2006/383/pdf/Uebungen_zur_Mathematik_fuer_Wirtschaftsinformatiker.pdf [letzter Zugriff: 27.02.2017]).
- Nitzsche, M. (2009): Graphen für Einsteiger. Rund um das Haus vom Nikolaus. 3. Auflage. Vieweg +Teubner, Wiesbaden.
- Teschl, G./Teschl, S. (2013): Diskrete Mathematik und lineare Algebra. 4. Auflage, Springer Vieweg, Berlin, Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

DLBCSM201

Web Application Development

Module Code: DLBCSWAD

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	DLBCSOOPJ01, DLBCSDSJCL01	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. André Hollstein (Web Application Development)

Contributing Courses to Module

- Web Application Development (DLBCSWAD01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Workbook

Split Exam

Weight of Module

see curriculum

Module Contents

- Architectural Foundations
- Tools of web development
- HTML
- CSS
- Javascript
- Web application testing and security

Learning Outcomes**Web Application Development**

On successful completion, students will be able to

- identify important elements and describe the structure of current web application architectures.
- write simple static web pages using HTML.
- design simple web pages using CSS.
- store and handle structured information using XML.
- program simple dynamic web content using PHP.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field.

Web Application Development

Course Code: DLBCSWAD01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSOOPJ01, DLBCSDSJCL01

Course Description

This course aims to empower students to program simple web applications using established technologies. At first, they will gain important insights into the typical structure of current web application architectures. Based on that knowledge, the students will learn the hypertext markup language (HTML) to develop simple web pages. Next, they will familiarize themselves with the most important and common elements of the cascading stylesheet (CSS) standard to layout the content written in HTML. To implement simple dynamic web pages, students learn how to use Javascript and exemplary supporting frameworks. This is combined with the use of relevant tools for the development and source code management of web pages. Finally, they will learn the very basics of the web page testing and security.

Course Outcomes

On successful completion, students will be able to

- identify important elements and describe the structure of current web application architectures.
- write simple static web pages using HTML.
- design simple web pages using CSS.
- store and handle structured information using XML.
- program simple dynamic web content using PHP.

Contents

1. Architectural Foundations
 - 1.1 Structure and History of the Internet
 - 1.2 Internet protocols and URIs
 - 1.3 Web application architecture
 - 1.4 Current trends
2. Tools of web development
 - 2.1 Development Tools
 - 2.2 Version management
 - 2.3 Package Manager
 - 2.4 Upload/Deployment

3. Static web pages development
 - 3.1 Fundamentals of HTML5
 - 3.2 Fundamentals of CSS
4. Advanced design techniques
 - 4.1 Responsive web design
 - 4.2 Page layout
 - 4.3 Media queries
 - 4.4 CSS Frameworks
5. Web page development with JavaScript
 - 5.1 JavaScript history, ES5/ES6
 - 5.2 JavaScript fundamentals
 - 5.3 Use of JSON
 - 5.4 Common JavaScript Frameworks
6. Web application testing and security
 - 6.1 Testing of web applications
 - 6.2 Basic security concepts and principles

Literature**Compulsory Reading****Further Reading**

- Ali Sunyaev (2020): Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies. Bd. 1st ed., Springer, Cham.
- Bühler, P./ Sinner, D. / Schlaich P. (2017): HTML5 und CSS3: Semantik - Design - Responsive Layouts. Springer, Berlin Heidelberg.
- Ferguson R. (2019): Beginning JavaScript: The Ultimate Guide to Modern JavaScript Development. Bd.3rd edition, Apress, Berkeley, CA.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Workbook

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

DLBCSWAD01

Collaborative Work

Module Code: DLBCSCW

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Karin Halbritter (Collaborative Work)

Contributing Courses to Module

- Collaborative Work (DLBCSCW01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Oral Assignment

Split Exam

Weight of Module

see curriculum

Module Contents

- Self-Directed and Collaborative Learning
- Networking and Cooperation
- Performance in (Virtual) Teams
- Communication, Arguments, and Being Convincing
- Potentials for Conflict and Managing Conflicts
- Self-Management and Personal Skills

Learning Outcomes**Collaborative Work**

On successful completion, students will be able to

- design their own self-directed and collaborative learning processes with analog and digital media.
- initiate local and virtual cooperation and select suitable methods for shaping cooperation.
- assess different forms of communication in relation to the goals and requirements of different situations and reflect one's own communication and argumentation behaviour.
- explain potentials for conflict and the role of emotions in conflicts and describe the use of systemic methods in the target- and solution-oriented handling of conflicts.
- form an idea of one's own resources, present methods of self-management and self-motivation, and derive appropriate strategies.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Business Administration & Management

Links to other Study Programs of IUBH

All Bachelor Programmes in the Business & Management fields

Collaborative Work

Course Code: DLBCSCW01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The course supports the students in building up and expanding important interdisciplinary competences for our networked world, and in doing so, students can take advantage of the opportunities for constructive cooperation with others. It presents essential forms and design possibilities of collaborative learning and working, imparts basic knowledge and tools for self-managed, flexible, and creative thinking, learning and acting and familiarizes students with the topics of empathy and emotional intelligence. Students are also encouraged to use the course contents. In this way, they promote their autonomous competence to act and their competence in the interactive application of tools and in interacting in heterogeneous groups.

Course Outcomes

On successful completion, students will be able to

- design their own self-directed and collaborative learning processes with analog and digital media.
- initiate local and virtual cooperation and select suitable methods for shaping cooperation.
- assess different forms of communication in relation to the goals and requirements of different situations and reflect one's own communication and argumentation behaviour.
- explain potentials for conflict and the role of emotions in conflicts and describe the use of systemic methods in the target- and solution-oriented handling of conflicts.
- form an idea of one's own resources, present methods of self-management and self-motivation, and derive appropriate strategies.

Contents

1. Learning for a Networked World in a Networked World
 - 1.1 Requirements and Opportunities of the VUCA World
 - 1.2 Learning, Information, and Dealing with Knowledge and Ignorance
 - 1.3 C-Model: Collective – Collaborative – Continuous – Connected
 - 1.4 Checking Your Own Learning Behaviour
2. Networking and Cooperation
 - 2.1 Finding and Winning Suitable Cooperation Partners
 - 2.2 Sustainable Relationships: Digital Interaction and Building Trust
 - 2.3 Collaboration: Organizing Locally and Virtually and Using Media
 - 2.4 Social Learning: Agile, Collaborative, and Mobile Planning of Learning Processes

3. Performance in (Virtual) Teams
 - 3.1 Goals, Roles, Organization and Performance Measurement
 - 3.2 Team Building and Team Flow
 - 3.3 Scrum as a Framework for Agile Project Management
 - 3.4 Design Thinking, Kanban, Planning Poker, Working-in-Progress-Limits & Co
4. Communicate and Convince
 - 4.1 Communication as Social Interaction
 - 4.2 Language, Images, Metaphors, and Stories
 - 4.3 It's the Attitude that Counts: Open, Empathetic, and Appreciative Communication
 - 4.4 Listen Actively - Argue - Convince - Motivate
 - 4.5 Analyze Your Own Conversational and Argumentational Skills
5. Recognize Conflict Potentials - Handle Conflicts - Negotiate Effectively
 - 5.1 Respecting Diversity - Seizing Opportunities
 - 5.2 Developing Empathy for Yourself and Others
 - 5.3 Systemic Work Solutions and Reframing
 - 5.4 Negotiate Constructively: Finding Clear Words - Interests Instead of Positions
6. Realize Your Own Projects
 - 6.1 Set Goals Effectively - Focus - Reflect
 - 6.2 The Agile Use of One's Own Time
 - 6.3 (Self-)Coaching and Inner Team
 - 6.4 Strategies and Methods for Self-Management and Self-Motivation
7. Mobilize Your Resources
 - 7.1 Recognizing Resources - Regulating Emotions
 - 7.2 Reflection and Innovation - Lateral Thinking and Creativity
 - 7.3 Transfer Strength and Willpower: Analyzing and Controlling Condition Factors
8. Construction Kit: Overview of Concepts, Tools, and Methods
 - 8.1 Communicate, Cooperate, Negotiate, Argue
 - 8.2 Think, Reflect, Develop Ideas, Decide, Lead Yourself

Literature**Compulsory Reading****Further Reading**

- Baber, A. (2015): Strategic connections. The new face of networking in a collaborative world. Amacom, New York.
- Goleman, D. (2013): Focus. The hidden driver of excellence. Harper Collins USA, New York.
- Kaats, E./Opheij, W. (2014): Creating conditions for promising collaboration. Alliances, networks, chains, strategic partnerships. Springer Management, Berlin.
- Lang, M. D. (2019): The guide to reflective practice in conflict resolution. Rowman & Littlefield, Lanham/Maryland.
- Martin, S. J./Goldstein, N. J./Cialdini, R. B. (2015): The small BIG. Small changes that spark BIG influence. Profile Books, London.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Oral Assignment

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Statistics: Probability and Descriptive Statistics

Module Code: DLBDSSPDS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Dr. Stefan Stöckl (Statistics: Probability and Descriptive Statistics)

Contributing Courses to Module

- Statistics: Probability and Descriptive Statistics (DLBDSSPDS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Probability
- Random variables
- Joint distributions
- Expectation and variance
- Inequalities and limit theorems

Learning Outcomes**Statistics: Probability and Descriptive Statistics**

On successful completion, students will be able to

- define probability, random variable, and probability distribution.
- understand the concept of Bayesian statistics.
- grasp the definition of joint and marginal distributions.
- calculate expectation values and higher moments.
- comprehend important inequality equations and limit theorems.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Methods

Links to other Study Programs of IUBH

All Bachelor Programmes in the Business & Management fields

Statistics: Probability and Descriptive Statistics

Course Code: DLBDSSPDS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Statistical description and analysis are the foundations for data-driven analysis and prediction methods. This course introduces the fundamentals, beginning with a formal definition of probabilities and introduction to the concepts underlying Bayesian statistics. Random variables and probability density distributions are then discussed, as well as the concept of joint and marginal distributions. The importance of various discrete and continuous distributions and their applications is stressed. Characterizing distributions is an important aspect of describing the behavior of probability distributions. Students are familiarized with expectation values, variance, and covariance. The concepts of algebraic and central moments and moment-generating functions complement the characterization of probability distributions. Finally, this course focuses on important inequalities and limit theorems such as the law of large numbers or the central limit theorem.

Course Outcomes

On successful completion, students will be able to

- define probability, random variable, and probability distribution.
- understand the concept of Bayesian statistics.
- grasp the definition of joint and marginal distributions.
- calculate expectation values and higher moments.
- comprehend important inequality equations and limit theorems.

Contents

1. Probability
 - 1.1 Definitions
 - 1.2 Independent events
 - 1.3 Conditional probability
 - 1.4 Bayesian statistics
2. Random Variables
 - 2.1 Random Variables
 - 2.2 Distribution functions and probability mass functions
 - 2.3 Important discrete probability distributions
 - 2.4 Important continuous probability distributions

3. Joint Distributions
 - 3.1 Joint distributions
 - 3.2 Marginal distributions
 - 3.3 Independent random variables
 - 3.4 Conditional distributions
4. Expectation and Variance
 - 4.1 Expectation of a random variable, conditional expectations
 - 4.2 Variance and covariance
 - 4.3 Expectations and variances of important probability distributions
 - 4.4 Algebraic and central moments
 - 4.5 Moment-generating functions
5. Inequalities and Limit Theorems
 - 5.1 Probability inequalities
 - 5.2 Inequalities for expectations
 - 5.3 The law of large numbers
 - 5.4 Central limit theorem

Literature**Compulsory Reading****Further Reading**

- Bruce, P., & Bruce, A. (2017). Practical statistics for data scientists: 50 essential concepts. Sebastopol, CA: O'Reilly.
- Downey, A. B. (2014). Think stats (2nd ed.). Sebastopol, CA: O'Reilly.
- Downey, A. B. (2013). Think Bayes. Sebastopol, CA: O'Reilly.
- Reinhart, A. (2015). Statistics done wrong: The woefully complete guide. San Francisco, CA: No Starch Press.
- Wassermann, L. (2004). All of statistics: A concise course in statistical inference. New York, NY: Springer Science+Business Media.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☒ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

DLBDSSPDS01

Computer Architecture and Operating Systems

Module Code: DLBCSCAOS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Computer Architecture and Operating Systems)

Contributing Courses to Module

- Computer Architecture and Operating Systems (DLBCSCAOS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Basic Concepts for Computer Architecture
- Computer Architecture
- Computer Hardware
- Assembly Languages
- Operating Systems Basics
- Popular Operating Systems

Learning Outcomes**Computer Architecture and Operating Systems**

On successful completion, students will be able to

- to explain the basic concepts of computer architecture.
- compare the different types of computer hardware.
- describe the meaning and functionality of simple assembly programs.
- explain the basic functions of operating systems.
- compare the different types of operating systems.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field.

Computer Architecture and Operating Systems

Course Code: DLBCSCAOS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSICS01

Course Description

This course introduces a fundamental topic of computer science. The architecture of computers and the inner workings of operating systems are fundamental concepts of computer science. This course introduces this topic, including an overview of the various types of computer hardware and an overview of assembly languages, which form a link between computer architecture and operating systems.

Course Outcomes

On successful completion, students will be able to

- to explain the basic concepts of computer architecture.
- compare the different types of computer hardware.
- describe the meaning and functionality of simple assembly programs.
- explain the basic functions of operating systems.
- compare the different types of operating systems.

Contents

1. Basic Concepts for Computer Architecture
 - 1.1 Historical Overview
 - 1.2 Semiconductor Technology
 - 1.3 Digital Logic and Binary Arithmetic
 - 1.4 Hardware Design and Hardware Description Languages
2. Computer Architecture
 - 2.1 Computer Architecture Design Goals
 - 2.2 Instruction Set Architecture
 - 2.3 Microarchitecture
 - 2.4 System Design

3. Computer Hardware
 - 3.1 Personal Computers
 - 3.2 Mainframes
 - 3.3 Servers
 - 3.4 Supercomputers
 - 3.5 Mobile Systems
 - 3.6 Embedded Systems
4. Assembly Languages
 - 4.1 Role and Importance of Assembly Languages
 - 4.2 Introduction to Programming in Assembly Languages
 - 4.3 Compiling and Linking
 - 4.4 Application of Assembly Languages
5. Operating Systems Basics
 - 5.1 Role and Types of Operating Systems
 - 5.2 Operating System Kernel
 - 5.3 File Systems
 - 5.4 Memory Management
 - 5.5 Processes and Threads
 - 5.6 Security
6. Popular Operating Systems
 - 6.1 Basic Concepts of Windows
 - 6.2 Basic Concepts of Unix and Linux
 - 6.3 Basic Concepts of Apple Operating Systems
 - 6.4 Basic Concepts of Mobile Operating Systems

Literature**Compulsory Reading****Further Reading**

- Harris, D.M./Harris, S.L. (2013): Digital Design and Computer Architecture. 2nd edition, Morgan Kaufman, Waltham, MA.
- Patt, Y.N./Patel, S.J. (2019): Introduction to Computing Systems: From Bits & Gates to C/C++ & Beyond. 3rd edition, McGraw-Hill Education, New York.
- Tanenbaum, A.S./Bos, H. (2014): Modern Operating Systems. 4th edition, Pearson Education, Harlow.
- Tutorials Point (2020): Assembly Programming Tutorial. (URL: https://www.tutorialspoint.com/assembly_programming/index.htm [last access 2020-02-04]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Project: Java and Web Development

Module Code: DLBCSPJWD

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	None	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Sandra Rebholz (Project: Java and Web Development)

Contributing Courses to Module

- Project: Java and Web Development (DLBCSPJWD01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Portfolio

Split Exam

Weight of Module

see curriculum

Module Contents

- This course focuses on the development of practical skills. For a given problem and/or context, students independently plan, design, develop, and evaluate small web applications. A set of specific problems and contexts are provided by the tutor and may vary. Each student chooses one and creates an individual solution, the results of which are gathered in a portfolio.

Learning Outcomes**Project: Java and Web Development**

On successful completion, students will be able to

- plan the development process of small web applications.
- specify requirements of small web applications to fulfill given needs.
- design and develop web applications that meet specific requirements.
- evaluate if a developed web application meets the specified requirements.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Project: Java and Web Development

Course Code: DLBCSPJWD01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSOOPJ01, DLBCSDSJCL01, DLBCSWAD01, DLBCSDWIS01

Course Description

This course focuses on the development of practical skills. Students apply their skills to create a small web application that fulfills a set of given needs and evaluate the result. The results are gathered and presented in a portfolio which will be assessed.

Course Outcomes

On successful completion, students will be able to

- plan the development process of small web applications.
- specify requirements of small web applications to fulfill given needs.
- design and develop web applications that meet specific requirements.
- evaluate if a developed web application meets the specified requirements.

Contents

- To a given problem and/or a given context, the students plan, design, develop and evaluate small web applications on their own authority. A set of specific problems and contexts are provided by the tutor and may vary. The students choose one and create their own solution.

Literature

Compulsory Reading

Further Reading

- Bloch, J. (2017). Effective Java (3rd ed.). Boston, MA: Addison-Wesley.
- Harrer, S., Lenhard, J., & Dietz, L. (2018). Java by comparison: Become a Java craftsman in 70 examples. Raleigh, NC: Pragmatic Bookshelf.
- Martin, R. M. (2017). Clean architecture: A craftsman's guide to software structure and design. Boston, MA: Prentice Hall.
- Freeman, E., & Robson, E. (2004). Head first design patterns: A brain-friendly guide. Sebastopol, CA: O'Reilly.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Portfolio

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

3. Semester

Database Modeling and Database Systems

Module Code: DLBCSDMDS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Ralf Kneuper (Database Modeling and Database Systems)

Contributing Courses to Module

- Database Modeling and Database Systems (DLBCSDMDS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Fundamentals of relational databases
- Simple database queries
- Entity/Relationship (E/R) Diagrams
- database development
- Complex database queries across multiple tables
- Changing data in databases
- NoSQL database systems

Learning Outcomes**Database Modeling and Database Systems**

On successful completion, students will be able to

- describe the basic concepts of the relational data model and distinguish them from each other.
- visually model data schemas.
- know SQL queries, read data from databases, change the data stock, and have experience in their use.
- design, create, and modify SQL queries and data schemas for SQL databases, and have experience using them.
- independently design database schemas and create database queries to solve concrete problems.
- know the most important NoSQL concepts and distinguish them from each other.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Database Modeling and Database Systems

Course Code: DLBCSDMDS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Stored data form the basis of many value chains of an information and knowledge society. The methodical structuring of data through data schemas therefore forms an important basis for storing information in such a way that it can be retrieved and processed quickly and easily. In addition to the structured storage of data, structured access to large amounts of data must also be possible. This course teaches students how to store data in relational data models and how to access stored data with SQL. In addition to relational database systems, modern DB systems (NoSQL) for storing and accessing data will be presented.

Course Outcomes

On successful completion, students will be able to

- describe the basic concepts of the relational data model and distinguish them from each other.
- visually model data schemas.
- know SQL queries, read data from databases, change the data stock, and have experience in their use.
- design, create, and modify SQL queries and data schemas for SQL databases, and have experience using them.
- independently design database schemas and create database queries to solve concrete problems.
- know the most important NoSQL concepts and distinguish them from each other.

Contents

1. Fundamentals of Relational Databases
 - 1.1 Basic Concepts of the Relational Data Model
 - 1.2 Find and Delete Records in the Database
 - 1.3 SQL and Relational Database Systems
2. Querying Data from a Single Table
 - 2.1 Query Data (SELECT)
 - 2.2 Query Data With Condition (WHERE)
 - 2.3 Sort Query Output (ORDER BY)
 - 2.4 Queries With Group Formation (GROUP BY)
 - 2.5 Subqueries With Nested SELECT Statements

3. Conception and Modeling of Relational Databases
 - 3.1 The Entity Relationship Model
 - 3.2 Relationships and Cardinalities in E/R Models
 - 3.3 Normal Forms of Databases
4. Creation of Relational Databases
 - 4.1 Logical Database Design Activities
 - 4.2 Mapping of the Conceptual Data Model into the Physical Data Model
 - 4.3 Generation of Tables in SQL Databases from E/R Diagrams
5. Complex Database Queries on Multiple Tables
 - 5.1 Composite Quantities (JOIN)
 - 5.2 Set Operations
 - 5.3 Data Views With CREATE VIEW
6. Manipulating Records in Databases
 - 6.1 Insert New Data Records (INSERT)
 - 6.2 Change Existing Records
 - 6.3 Transactions
7. NoSQL Database Systems
 - 7.1 Motivation and Basic Idea
 - 7.2 Selected Groups of NoSQL Systems

Literature**Compulsory Reading****Further Reading**

- Edlich, S./Friedland, A./Hampe, J./Brauer, B./Brückner, M. (2011): NoSQL. Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken. 2. Auflage, Hanser, München.
- Throll, M./Bartosch, O. (2010): Einstieg in SQL. Verstehen, einsetzen, nachschlagen. 4. Auflage, Galileo Computing, Bonn.
- Steiner, R. (2017): Grundkurs Relationale Datenbanken. Einführung in die Praxis der Datenbankentwicklung für Ausbildung, Studium und IT-Beruf. 9. Auflage, Springer Vieweg, Wiesbaden.
- Unterstein, M./Matthiessen, G. (2013): Anwendungsentwicklung mit Datenbanken. 5. Auflage, Springer Vieweg, Wiesbaden.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

DLBCSDMDS01

Project: Build a Data Mart in SQL

Module Code: DLBDSPBDM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Sharam Dadashnia (Project: Build a Data Mart in SQL)

Contributing Courses to Module

- Project: Build a Data Mart in SQL (DLBDSPBDM01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Portfolio

Split Exam

Weight of Module

see curriculum

Module Contents

- This course is about the implementation of a practical database use case employing previously-acquired knowledge on pertaining approaches and methods.

Learning Outcomes**Project: Build a Data Mart in SQL**

On successful completion, students will be able to

- transfer previously-acquired knowledge about database methods and approaches to practical use cases.
- design, architect, and implement a working data-mart solution.
- reason about design choices of and trade-offs between relevant implementation alternatives.
- critically evaluate said choices with respect to the stated design goal.
- describe and explain the resulting solution.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Data Science & Artificial Intelligence.

Links to other Study Programs of IUBH

All Bachelor Programmes in the It & Technology field(s).

Project: Build a Data Mart in SQL

Course Code: DLBDSPBDM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course provides the opportunity to implement a realistic database use case scenario. A list of use case ideas is provided on the online learning platform. In addition, the students can contribute use case ideas of their own in accord with the tutor. The core aim is to apply the hitherto theoretical knowledge of database methods and approaches to solve a real-world application scenario. This entails reasoning about possible design and architectural choices in a rational way, as well as implementing them in a functioning database system.

Course Outcomes

On successful completion, students will be able to

- transfer previously-acquired knowledge about database methods and approaches to practical use cases.
- design, architect, and implement a working data-mart solution.
- reason about design choices of and trade-offs between relevant implementation alternatives.
- critically evaluate said choices with respect to the stated design goal.
- describe and explain the resulting solution.

Contents

- In this course, students apply their knowledge of data modeling and databases to implement a project use case of their choosing. All relevant artefacts, like use case evaluation, chosen implementation method, code, and outcomes, are documented in the form of a written project report.

Literature**Compulsory Reading****Further Reading**

- Date, C. J. (2012). Database design and relational theory. Sebastopol, CA: O'Reilly.
- DeBarros, A. (2018). Practical SQL: A beginner's guide to storytelling with data. San Francisco, CA: No Starch Press.
- Harrington, J. L. (2016). Relational database design and implementation (4th ed.). Burlington, MA: Morgan Kaufmann.
- Hernandez, M. J. (2013). Database design for mere mortals: A hands-on guide to relational database design (3rd ed.). Boston, MA: Addison-Wesley.
- Viescas, J. (2018). SQL queries for mere mortals: A hands-on guide to data manipulation in SQL (4th ed.). Boston, MA: Addison-Wesley.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: yes
Type of Exam	Portfolio

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

DLBDSPBDM01

Requirements Engineering

Module Code: DLBCSRE

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Tobias Brückmann (Requirements Engineering)

Contributing Courses to Module

- Requirements Engineering (DLBCSRE01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Basics of requirements engineering
- Enterprise modeling
- Requirement determination techniques
- Techniques of requirements documentation
- Testing and coordination of requirements
- Managing requirements

Learning Outcomes**Requirements Engineering**

On successful completion, students will be able to

- describe models of enterprise modeling relevant to IT support and have experience in modeling.
- understand techniques and methods for determining requirements of IT systems and be able to distinguish them from each other.
- understand techniques for the documentation of requirements on IT systems and have experience in their use.
- describe techniques for testing, coordinating, and managing the requirements of IT systems and be able to distinguish between them.
- independently select suitable techniques and methods of requirements engineering for given project situations.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Requirements Engineering

Course Code: DLBCSRE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The early phases of software development are largely characterized by the fact that functional and technical requirements for the IT system have to be determined. The determination of these requirements must be carried out extremely carefully because all of the following activities in the SW development process are planned and executed on the basis of documented requirements. In this course, procedures, methods, and models are covered, which make it possible to have a structured and methodical determination and documentation of requirements for operational information systems.

Course Outcomes

On successful completion, students will be able to

- describe models of enterprise modeling relevant to IT support and have experience in modeling.
- understand techniques and methods for determining requirements of IT systems and be able to distinguish them from each other.
- understand techniques for the documentation of requirements on IT systems and have experience in their use.
- describe techniques for testing, coordinating, and managing the requirements of IT systems and be able to distinguish between them.
- independently select suitable techniques and methods of requirements engineering for given project situations.

Contents

1. Fundamentals and Terms of Requirements Engineering
 - 1.1 Requirements Engineering in the Software Process
 - 1.2 Core Activities in Requirements Engineering
 - 1.3 What is a Requirement?
2. Determination of Requirements
 - 2.1 Determination of the System Context
 - 2.2 Determination of the Sources of Requirements
 - 2.3 Selection of the Appropriate Investigative Techniques
 - 2.4 Determine Requirements Using Techniques

3. Selected Investigative Techniques
 - 3.1 Creativity Techniques
 - 3.2 Interview Techniques
 - 3.3 Observation Techniques
 - 3.4 Prototyping
4. Documentation of Requirements
 - 4.1 Activities for Documenting Requirements
 - 4.2 Typical Elements of Requirements Documentation
 - 4.3 Forms of Documentation
5. Modeling of Processes
 - 5.1 Basics and Terms
 - 5.2 Modeling with the Business Process Model and Notation
 - 5.3 Modeling with Event Driven Process Chains
6. Modeling of Systems
 - 6.1 Fundamentals of Unified Modeling Language
 - 6.2 UML Use Case Diagram
 - 6.3 UML Activity Diagram
 - 6.4 UML Class Diagram
 - 6.5 UML State Diagram
7. Checking and Reconciling Requirements
 - 7.1 Activities for Checking and Reconciling Requirements
 - 7.2 Test Criteria
 - 7.3 Test Principles
 - 7.4 Testing Techniques
 - 7.5 Coordination of Requirements
8. Management of Prioritization Requirements and Techniques
 - 8.1 Managing Requirements
 - 8.2 Techniques for Prioritizing Requirements

Literature**Compulsory Reading****Further Reading**

- Allweyer, T. (2009): BPMN 2.0. Business Process Model and Notation. Einführung in den Standard für die Geschäftsprozessmodellierung. 2. Auflage, Books on Demand, Norderstedt.
- Balzert, H. (2010): UML 2 kompakt mit Checklisten. 3. Auflage, Spektrum, Heidelberg.
- Booch, G./Rumbaugh, J./Jacobson, I. (2006): Das UML Benutzerhandbuch. Addison-Wesley, Bonn.
- Cohn, M. (2010): User Stories für die agile Software-Entwicklung mit Scrum, XP u.a. mitp, Frechen.
- Freund, J./Rücker, B. (2012): Praxishandbuch BPMN 2.0. 3. Auflage, Hanser. München.
- Gadatsch, A. (2012): Grundkurs Geschäftsprozess-Management. Methoden und Werkzeuge für die IT-Praxis. Eine Einführung für Studenten und Praktiker. 7. Auflage, Vieweg+Teubner, Wiesbaden.
- Pohl, K. (2008): Requirements Engineering. Grundlagen, Prinzipien, Techniken. 2. Auflage, dpunkt.verlag, Heidelberg.
- Pohl, K./Rupp, C. (2011): Basiswissen Requirements Engineering. Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level. 3. Auflage, dpunkt.verlag, Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Computer Networks and Distributed Systems

Module Code: DLBCSCNDS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Computer Networks and Distributed Systems)

Contributing Courses to Module

- Computer Networks and Distributed Systems (DLBCSCNDS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Computer Networks
- TCP/IP and Internet
- Communication and Coordination
- Distributed Systems Architectures
- Mobile Computing
- Network Security

Learning Outcomes**Computer Networks and Distributed Systems**

On successful completion, students will be able to

- explain and compare the OSI reference model and the TCP/IP protocol stack.
- explain the most important IP-based protocols and services and their application.
- explain and compare different distributed system architectures.
- explain and compare the most important mobile communication networks.
- explain basic challenges and solutions to Internet security.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field.

Computer Networks and Distributed Systems

Course Code: DLBCSCNDS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSICS01

Course Description

To an ever-greater extent, computers are not stand-alone but integrated into networks in which data and other computer system functions can be accessed. This enables distributed systems in which data and certain computer functions can be systematically assigned to different computers within a network in order to jointly accomplish defined tasks. While the various computers within a network or a distributed system were in the past stationary, many computers used within a network are now mobile, leading to completely new application scenarios in both the private and the business context.

Course Outcomes

On successful completion, students will be able to

- explain and compare the OSI reference model and the TCP/IP protocol stack.
- explain the most important IP-based protocols and services and their application.
- explain and compare different distributed system architectures.
- explain and compare the most important mobile communication networks.
- explain basic challenges and solutions to Internet security.

Contents

1. Computer Networks
 - 1.1 Basics of Data Transmission
 - 1.2 OSI Reference Model
 - 1.3 Network Topologies
2. TCP/IP and Internet
 - 2.1 Origin and Structure of the Internet
 - 2.2 TCP/IP Protocol Stack
 - 2.3 Selected IP-based Protocols and Services

3. Communication and Coordination
 - 3.1 Basic Concepts
 - 3.2 Concurrency, Semaphores, and Deadlock
 - 3.3 Remote Procedure Calls
 - 3.4 Message-oriented Communication
 - 3.5 CORBA
 - 3.6 EJBs
4. Distributed Systems Architectures
 - 4.1 Client Server Systems and Distributed Applications
 - 4.2 Service Orientation: SOA, Web Services, and Microservices
 - 4.3 Cloud Applications
 - 4.4 Distributed Database Systems
 - 4.5 High-Performance Computing Cluster
 - 4.6 Distributed Ledger Technologies
5. Mobile Computing
 - 5.1 Fundamentals, Techniques, and Protocols of Mobile Computing
 - 5.2 Mobile Internet and its Applications
 - 5.3 Mobile Communication Networks
6. Network Security
 - 6.1 Introduction to Network Security
 - 6.2 Authentication in Distributed Systems
 - 6.3 Secure Internet Protocols
 - 6.4 Security and Data Protection in Mobile Systems

Literature

Compulsory Reading

Further Reading

- van Steen, M. (2017): Distributed Systems. 3rd edition, CreateSpace Independent Publishing Platform, o.O..
- Sunyaev, A. (2020): Internet Computing. Principles of Distributed Systems and Emerging Internet-Based Technologies. Springer.
- Elmasri, R. / Navathe, S.B. (2017): Fundamentals of Database Systems. 7th edition, Pearson Education Limited, Harlow.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

DLBCSCNDS01

Algorithms, Data Structures, and Programming Languages

Module Code: DLBCSL

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimaldauer: 1 Semester	WiSe/SoSe	English

Module Coordinator

N.N. (Algorithms, Data Structures, and Programming Languages)

Contributing Courses to Module

- Algorithms, Data Structures, and Programming Languages (DLBCSL01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Data structures
- Algorithm design
- Important algorithms
- Programming paradigms and the basic terms of programming languages
- Programme analysis tools
- Overview of common programming languages

Learning Outcomes**Algorithms, Data Structures, and Programming Languages**

On successful completion, students will be able to

- explain basic data structures and compare and apply them in concrete applications.
- explain basic algorithms.
- design, select and apply suitable algorithms and data structures for specific applications
- analyse sketched or programmed algorithms when or before running them
- explain and compare the common programming paradigms and programming languages.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Algorithms, Data Structures, and Programming Languages

Course Code: DLBCSL01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Programming essentially consists of selecting suitable algorithms and data structures for a specific task and converting them into program code. There are many different programming languages, which are based on different procedures and in which algorithms and data structures are implemented differently. In this module, these concepts, which have so far been dealt with using concrete examples, are systematically presented and applied more broadly in order to give students the necessary tools to develop a systematic approach to programming.

Course Outcomes

On successful completion, students will be able to

- explain basic data structures and compare and apply them in concrete applications.
- explain basic algorithms.
- design, select and apply suitable algorithms and data structures for specific applications
- analyse sketched or programmed algorithms when or before running them
- explain and compare the common programming paradigms and programming languages.

Contents

1. Basic Concepts
 - 1.1 Algorithms, Data Structures, and Programming Languages as the Basics of Programming
 - 1.2 Detailing and Abstraction
 - 1.3 Control Structures
 - 1.4 Types of Data
 - 1.5 Basic Data Structures (List, Chain, Tree)
2. Data Structures
 - 2.1 Advanced Data Structures: Queue, Heap, Stack, Graph
 - 2.2 Abstract Data Types, Objects, and Classes
 - 2.3 Polymorphism

3. Algorithm Design
 - 3.1 Induction, Iteration, and Recursion
 - 3.2 Methods of Algorithm Design
 - 3.3 Correctness and Verification of Algorithms
 - 3.4 Efficiency (complexity) of algorithms
4. Basic Algorithms
 - 4.1 Traversing and Linearization of Trees
 - 4.2 Search Algorithms
 - 4.3 Sorting Algorithms
 - 4.4 Search in Strings
 - 4.5 Hash Algorithms
 - 4.6 Pattern Recognition
5. Measuring Programmes
 - 5.1 Type inference and IDE interactive support
 - 5.2 Cyclomatic and referential complexity
 - 5.3 Digesting code documentation
 - 5.4 Compiler optimization
 - 5.5 Code coverage
 - 5.6 Unit and integration testing
 - 5.7 Heap analysis
6. Programming Languages
 - 6.1 Programming Paradigms
 - 6.2 Execution of Programs
 - 6.3 Types of Programming Languages
 - 6.4 Syntax, Semantics, and Pragmatics
 - 6.5 Variables and Type Systems
7. Overview of Important Programming Languages
 - 7.1 Assembler and Webassembly
 - 7.2 C and C++
 - 7.3 Java and C#
 - 7.4 Haskell, Lisp
 - 7.5 JavaScript and its relatives
 - 7.6 Other imperative programming languages

Literature**Compulsory Reading****Further Reading**

- Gumm H. P. /Sommer M. (2013): Einführung in die Informatik. 10. Auflage. Oldenbourg, München.
- Harel, D. (2006): Algorithmik. Die Kunst des Rechnens. Springer, Berlin/Heidelberg/New York.
- Cormen, T.,Leiserson, C., Rivest, R., Stein, C. (2009) Introduction to Algorithms, 3rd edition, MIT Press, Cambridge, Mass., USA

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

IT-Service Management

Module Code: DLBCSITSM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (IT-Service Management)

Contributing Courses to Module

- IT-Service Management (DLBCSITSM01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Fundamentals and terms of IT Service Management
- IT Infrastructure Library (ITIL)
- ITIL - Service Design
- ITIL - Service Transition
- ITIL - Service Operation
- Information Security Management with the IT-Baseline Protection-Framework of the BSI

Learning Outcomes**IT-Service Management**

On successful completion, students will be able to

- identify the basics and challenges of IT service management.
- describe the motivation and structure of the IT Infrastructure Library (ITIL) in order to determine its main elements and distinguish concrete activities in the service life cycle.
- present and compare the activities of ITIL governance and ITIL operational processes and develop concrete solutions using these activities.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

IT-Service Management

Course Code: DLBCSITSM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

IT Service Management is an approach to aligning and understanding a company's IT as a service provider and supporter of operational and business processes. Quality management and the handling of daily operations are at the forefront. This course covers the use of the IT Infrastructure Library (ITIL) in order to teach concepts, procedures, and best practice in IT Service Management (IT Operations). The course therefore considers the management of activities within the SW life cycle, which take place after the development of an IT system, and IT operation as a ongoing process in the day-to-day operation of a company's IT department.

Course Outcomes

On successful completion, students will be able to

- identify the basics and challenges of IT service management.
- describe the motivation and structure of the IT Infrastructure Library (ITIL) in order to determine its main elements and distinguish concrete activities in the service life cycle.
- present and compare the activities of ITIL governance and ITIL operational processes and develop concrete solutions using these activities.

Contents

1. Fundamentals and Terms of IT Service Management
 - 1.1 IT Services
 - 1.2 IT Service Management
2. IT Infrastructure Library (ITIL)
 - 2.1 Service Life Cycle and Process Groups in ITIL
 - 2.2 Service Strategy
 - 2.3 Continual Service Improvement
3. ITIL – Service Design
 - 3.1 Service Level Management
 - 3.2 Service Catalog Management
 - 3.3 Availability Management
 - 3.4 Further Processes in the Service Transition

4. ITIL – Service Transition
 - 4.1 Transition Planning and Support
 - 4.2 Change Management
 - 4.3 Service Asset and Configuration Management (SACM)
 - 4.4 Further Processes in the Service Transition
5. ITIL – Service Operation
 - 5.1 Event Management
 - 5.2 Incident Management
 - 5.3 Problem Management
 - 5.4 Further Processes in the Service Operation
6. Information Security Management with the IT-Baseline Protection Framework of the BSI
 - 6.1 Structure and Elements of BSI Basic Protection
 - 6.2 The Information Security Process

Literature

Compulsory Reading

Further Reading

- Beims, M. (2012): IT-Service Management in der Praxis mit ITIL. Hanser, München.
- Bundesamt für Sicherheit und Informationstechnik (Hrsg.) (2008): BSI-Standard 100-1. Managementsysteme für Informationssicherheit (ISMS). (URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/standard_1001_pdf.pdf?__blob=publicationFile [letzter Zugriff: 27.02.2017]).
- Bundesamt für Sicherheit und Informationstechnik (Hrsg.) (2008): BSI-Standard 100-2. IT-Grundschutz-Vorgehensweise. (URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/standard_1002_pdf.pdf?__blob=publicationFile [letzter Zugriff: 27.02.2017]).
- Bundesamt für Sicherheit und Informationstechnik (Hrsg.) (2014): IT-Grundschutz-Kataloge. 14. Ergänzungslieferung. (URL: https://download.gsb.bund.de/BSI/ITGSK/IT-Grundschutz-Kataloge_2014_EL14_DE.pdf [letzter Zugriff: 27.02.2017]).
- Renner, B./Moser, U./Schmid, D. (2006): IT-Service-Management. Transparente IT-Leistungen & Messbare Qualität. BPX Edition, Rheinfelden.
- Tiemeyer, E. (Hrsg.) (2011): Handbuch IT-Management. Konzepte, Methoden, Lösungen und Arbeitshilfen für die Praxis. Hanser, München.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

DLBCSITSM01

4. Semester

Project: IT Service Management

Module Code: DLBCSPITSM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Project: IT Service Management)

Contributing Courses to Module

- Project: IT Service Management (DLBCSPITSM01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Project Report

Split Exam

Weight of Module

see curriculum

Module Contents

- All phases of an ITIL/IT project are carried out as part of an independent project.

Learning Outcomes**Project: IT Service Management**

On successful completion, students will be able to

- analyze typical problems and company situations from the area of IT service management in different project variations.
- develop, plan, and implement proposed solutions.
- convert theory into a pragmatic approach to a solution with the help of methodical tools from IT service management and project management.
- draw and apply the right conclusions in relation to their specific project environment.
- conceptually apply their theoretical knowledge to company-specific environmental factors.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Project: IT Service Management

Course Code: DLBCSPITSM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Based on the contents of the course "IT Service Management", selected aspects of the core processes of ITIL are deepened, discussed, selected, and applied within the framework of a project in a concept-related manner. All theoretical methods are considered and evaluated.

Course Outcomes

On successful completion, students will be able to

- analyze typical problems and company situations from the area of IT service management in different project variations.
- develop, plan, and implement proposed solutions.
- convert theory into a pragmatic approach to a solution with the help of methodical tools from IT service management and project management.
- draw and apply the right conclusions in relation to their specific project environment.
- conceptually apply their theoretical knowledge to company-specific environmental factors.

Contents

- Analysis, evaluation, and development of recommendations for taking action within the scope of concrete questions concerning aspects of IT Service Management. This is aided by the creation and planning of a project in the theoretical-theme context through all phases of project management.
- The quality assurance of the artefacts created is carried out both by the tutor and by students from the project groups.

Literature

Compulsory Reading

Further Reading

- Beims, M. (2012): IT-Service Management in der Praxis mit ITIL. 3. Auflage, Carl Hanser Verlag, München.
- Kittel, M./Koerting, T./Schött, D. (2006): Kompendium für ITIL-Projekte. Menschen, Methoden, Meilenstein – Von der Analyse zum selbstoptimierenden Prozess. Books on demand.
- ITIL (o. J.): Official ITIL Website. (URL: <http://www.itil-officialsite.com>)

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Theoretical Computer Science and Mathematical Logic

Module Code: DLBCSTCSML

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N. N. (Theoretical Computer Science and Mathematical Logic)

Contributing Courses to Module

- Theoretical Computer Science and Mathematical Logic (DLBCSTCSML01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Proposition and predicate logic
- Finite automata
- Formal languages
- Computability and Turing machines
- Complexity theory
- Petri nets

Learning Outcomes**Theoretical Computer Science and Mathematical Logic**

On successful completion, students will be able to

- formulate and translate predicate logical relationships into programming languages.
- use finite automata and regular expressions to describe technical facts.
- explain the Chomsky hierarchy.
- identify the limits of provability and predictability.
- explain the meaning and relevance of the P=NP problem.
- apply Petri nets for the description of technical facts.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Theoretical Computer Science and Mathematical Logic

Course Code: DLBCSTCSML01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Theoretical computer science and mathematical logic form the theoretical basics of computer science. However, this is not "pure theory", as these fundamentals are applied in many areas of computer science. These include, for example, the formulation of conditions in SQL queries or other programs based on statement and predicate logic, the use of finite state machines to specify systems with state transition diagrams, and the modeling of business and other processes with Petri nets. In addition, theoretical computer science and mathematical logic analyze the limits of computer science and computability, which cannot be exceeded irrespective of the technologies and algorithms used.

Course Outcomes

On successful completion, students will be able to

- formulate and translate predicate logical relationships into programming languages.
- use finite automata and regular expressions to describe technical facts.
- explain the Chomsky hierarchy.
- identify the limits of provability and predictability.
- explain the meaning and relevance of the P=NP problem.
- apply Petri nets for the description of technical facts.

Contents

1. Propositional Logic
 - 1.1 Basic Concepts
 - 1.2 Interpretation and Satisfiability
 - 1.3 Normal Forms
 - 1.4 Proof by Contradiction and Resolution
 - 1.5 Completeness
2. Predicate Logic
 - 2.1 Basic Concepts
 - 2.2 Completeness and Incompleteness
 - 2.3 Logic Programming with Prolog

3. Finite Automata and Regular Expressions
 - 3.1 Basic Concepts of Finite Automata
 - 3.2 Regular Expressions
 - 3.3 Practical Applications
4. Formal Languages and Grammars
 - 4.1 Basic Concepts
 - 4.2 The Chomsky Hierarchy
 - 4.3 Regular Languages
 - 4.4 Context Free Languages
 - 4.5 Context Sensitive Languages
5. Computability and Turing Machines
 - 5.1 Models of Computability
 - 5.2 Turing Machines
 - 5.3 Recursive Functions
 - 5.4 Computability and Decidability
 - 5.5 The Halting Problem
6. Complexity Theory
 - 6.1 Basic Concepts
 - 6.2 Complexity Classes
 - 6.3 $P=NP?$
7. Petri Nets
 - 7.1 Basic Concepts of Graphs and Petri Nets
 - 7.2 Invariants, Liveness, and Safety
 - 7.3 Process Modeling and Analysis with Petri Nets
8. Applications of Mathematical Logic and Theoretical Computer Science
 - 8.1 Parser and Compiler
 - 8.2 Program Verification
 - 8.3 Artificial Intelligence

Literature**Compulsory Reading****Further Reading**

- Dewdney, A.K. (1995): Der Turing Omnibus. Eine Reise durch die Informatik mit 66 Stationen. Springer, Berlin/Heidelberg/New York.
- Erk, K./Priese, L. (2008): Theoretische Informatik. 3. Auflage. Springer eXamen.press, Berlin/Heidelberg.
- Priese, L./Wimmerl, H. (2008): Petri-Netze. 2. Auflage. Springer eXamen.press, Berlin/Heidelberg.
- Schöning, U. (2000): Logik für Informatiker. 5. Auflage. Spektrum Verlag, Heidelberg/ Berlin.
- Schöning, U. (2008): Ideen der Informatik. Grundlegende Modelle und Konzepte der Theoretischen Informatik, 3. Auflage. Oldenbourg, München.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Introduction to Programming with Python

Module Code: DLBDSIPWP

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Dr. Reza Shahbazfar (Introduction to Programming with Python)

Contributing Courses to Module

- Introduction to Programming with Python (DLBDSIPWP01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Python as a programming language for data science
- Variables and built-in datatypes
- Statements and functions
- Error and exception handling
- Important Python data science modules

Learning Outcomes**Introduction to Programming with Python**

On successful completion, students will be able to

- use fundamental Python syntax.
- recollect common elementary data types.
- recognize foundational programming concepts and their realization in Python.
- understand error handling and logging.
- create working programs.
- list the most important libraries and packages for data science.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Data Science & Artificial Intelligence.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Introduction to Programming with Python

Course Code: DLBDSIPWP01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course provides students with a foundational understanding of the Python programming language. Following an introductory exposition to the importance of Python for data science-related programming tasks, students will be acquainted with fundamental programming concepts like variables, data types, and statements. Building on this basis, the important notion of a function is explained and errors, exception handling, and logging are explicated. The course concludes with an overview of the most widely-used library packages for data science.

Course Outcomes

On successful completion, students will be able to

- use fundamental Python syntax.
- recollect common elementary data types.
- recognize foundational programming concepts and their realization in Python.
- understand error handling and logging.
- create working programs.
- list the most important libraries and packages for data science.

Contents

1. Introduction
 - 1.1 Why Python?
 - 1.2 Obtaining and installing Python
 - 1.3 The Python interpreter , IPython, and Jupyter
2. Variables and Data Types
 - 2.1 Variables and value assignment
 - 2.2 Numbers
 - 2.3 Strings
 - 2.4 Collections
 - 2.5 Files

3. Statements
 - 3.1 Assignment, expressions, and print
 - 3.2 Conditional statements
 - 3.3 Loops
 - 3.4 Iterators and comprehensions
4. Functions
 - 4.1 Function declaration
 - 4.2 Scope
 - 4.3 Arguments
5. Errors and Exceptions
 - 5.1 Errors
 - 5.2 Exception handling
 - 5.3 Logs
6. Modules and Packages
 - 6.1 Usage
 - 6.2 Namespaces
 - 6.3 Documentation
 - 6.4 Popular data science packages

Literature

Compulsory Reading

Further Reading

- Barry, P. (2016): Head first Python: A brain-friendly guide. 2nd ed., O'Reilly, Sebastopol, CA.
- Lubanovic, B. (2019): Introducing Python. 2nd ed., O'Reilly, Sebastopol, CA.
- Lutz, M. (2013): Learning Python. 5th ed., O'Reilly, Sebastopol, CA.
- Matthes, E. (2019): Python crash course: A hands-on, project-based introduction to programming. 2nd ed., No Starch Press, San Francisco, CA.
- Ramalho, L. (2015): Fluent Python: Clear, concise, and effective programming. O'Reilly, Sebastopol, CA.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

DLBDSIPWP01

Software Quality Assurance

Module Code: DLBCSSQA

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Software Quality Assurance)

Contributing Courses to Module

- Software Quality Assurance (DLBCSSQA01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Systematic quality assurance of requirements, architectures, and processes
- Systematic testing of software
- Dynamic quality assurance: Testing
- Static quality assurance: Surveying and measuring
- Constructive quality management
- Organization and planning of software quality
- Introduction to software quality assurance

Learning Outcomes**Software Quality Assurance**

On successful completion, students will be able to

- understand motivation, use cases, and scenarios for aspects of quality management in the software process.
- understand important terms and the basis for the conception and execution of software tests.
- understand techniques and methods for constructive quality management and be able to distinguish them from each other.
- understand techniques and methods for analytical quality management and be able to distinguish them from one another.
- understand the general course of test activities and be able to select suitable methods and techniques for quality assurance for various artefacts and activities in the software process.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Software Quality Assurance

Course Code: DLBCSSQA01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Quality assurance is one of the accompanying activities of a software process. From the beginning, all created artefacts (documents, models, and program codes) must be quality-assured because the later an error in a system is detected, the more expensive it is to fix it. The course teaches techniques and procedures for accompanying quality assurance, starting with requirements analysis through to specification, architecture and design, and implementation. Even the quality assurance activities must be quality assured so that the software systems can be delivered at a good standard.

Course Outcomes

On successful completion, students will be able to

- understand motivation, use cases, and scenarios for aspects of quality management in the software process.
- understand important terms and the basis for the conception and execution of software tests.
- understand techniques and methods for constructive quality management and be able to distinguish them from each other.
- understand techniques and methods for analytical quality management and be able to distinguish them from one another.
- understand the general course of test activities and be able to select suitable methods and techniques for quality assurance for various artefacts and activities in the software process.

Contents

1. Introduction to Software Quality Assurance
 - 1.1 Motivation and Terms
 - 1.2 Principles of SW Quality Assurance
 - 1.3 Principles in Software Testing
 - 1.4 Cost of Quality
2. Organization and Planning of Software Quality
 - 2.1 Overview of the Quality Management Process
 - 2.2 Quality Planning and Quality Objectives
 - 2.3 Quality Assurance and Quality Improvement
 - 2.4 Quality Control

3. Constructive Quality Management
 - 3.1 Overview of Constructive Quality Assurance
 - 3.2 Selected Techniques
4. Static Quality Assurance: Surveying and Measuring
 - 4.1 Application and Overview of Static Processes
 - 4.2 Reviewing with Review Techniques
 - 4.3 Trade Fairs and Metrics
 - 4.4 Static Code Analysis
5. Dynamic Quality Assurance: Testing
 - 5.1 Deployment and an Overview of Dynamic Processes
 - 5.2 Use Case Based Test Case Creation
 - 5.3 Equivalence Class Formation and Limit Value Analysis
 - 5.4 State Based Test Case Creation
 - 5.5 Creation of Random Test Data
6. Systematic Testing of Software
 - 6.1 Methodological Testing Activities
 - 6.2 Component Test (Also: Module Test, Unit Test)
 - 6.3 Integration Tests
 - 6.4 System Tests
 - 6.5 Acceptance Tests
7. Systematic Quality Assurance of Requirements, Architectures, and Processes
 - 7.1 Quality Assurance of Requirements
 - 7.2 Quality Assurance of Architectures
 - 7.3 Quality Assurance of Software Processes

Literature**Compulsory Reading****Further Reading**

- Balzert, H. (1997): Lehrbuch der Software-Technik. Software-Management, Software-Qualitätssicherung und Unternehmensmodellierung. Spektrum, Heidelberg.
- Liggesmeyer, P. (2009): Software-Qualität. Testen, Analysieren und Verifizieren von Software. 2. Auflage, Spektrum, Heidelberg.
- Pol, M./Koomen, T./Spillner, A. (2002): Management und Optimierung des Testprozesses. Ein praktischer Leitfaden für erfolgreiches Testen von Software mit TPI und TMap. 2. Auflage, dpunkt.verlag, Heidelberg.
- Schneider, K. (2012): Abenteuer Softwarequalität. Grundlagen und Verfahren für Qualitätssicherung und Qualitätsmanagement. 2. Auflage, dpunkt.verlag, Heidelberg.
- Seidl, R./Sneed, H. S./Baumgartner, M. (2006): Der Systemtest. Anforderungsbasiertes Testen von Software-Systemen. Hanser, München.
- Spillner, T. et al. (2011): Praxiswissen Softwaretest. Testmanagement. Aus- und Weiterbildung zum Certified Tester. 3. Auflage, dpunkt.verlag, Heidelberg.
- Spillner, E./Linz, T. (2012): Basiswissen Softwaretest. Aus- und Weiterbildung zum Certified Tester. 5. Auflage, dpunkt.verlag, Heidelberg.
- Wallmüller, E. (1990): Software-Qualitätssicherung in der Praxis. Hanser, München.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Specification

Module Code: DLBCSS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Specification)

Contributing Courses to Module

- Specification (DLBCSS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Basics of specification
- Specification of user interfaces (GUIs)
- Specification of components and their interfaces
- Technical specification of detailed technical data models
- Specification of business rules
- Specification of data interfaces
- Specification of web services
- Specification of quality and limiting constraints

Learning Outcomes**Specification**

On successful completion, students will be able to

- know the motivation, use cases, and scenarios for the practical use of technical specifications.
- distinguish techniques from each other for the detailed specification of application interfaces and business logic of information systems and have experience with their use.
- identify and independently specify business objects and components.
- describe techniques for the detailed specification of technical interfaces between IT systems and be able to specify interfaces independently.
- explain techniques and procedures for the technical specification of quality properties and limiting conditions.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Specification

Course Code: DLBCSS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Based on the results of the technical requirements analysis, requirements for IT systems must be described as precisely as is appropriate for the respective situation. Based on specifications, project costs are estimated, and decisions are made about the internal design of the system. In this course different models and techniques that are used for the detailed specification of requirements for a system, subsystem, or system component are taught. The structure of the course is based on the structure of typical business information systems. In addition to the application layers, user interface, business logic, and data layer, the specification of web services is also covered.

Course Outcomes

On successful completion, students will be able to

- know the motivation, use cases, and scenarios for the practical use of technical specifications.
- distinguish techniques from each other for the detailed specification of application interfaces and business logic of information systems and have experience with their use.
- identify and independently specify business objects and components.
- describe techniques for the detailed specification of technical interfaces between IT systems and be able to specify interfaces independently.
- explain techniques and procedures for the technical specification of quality properties and limiting conditions.

Contents

1. Introduction to the Specification of Software Systems
 - 1.1 Basics and Terms for Specification
 - 1.2 Elements and Outlines of Specifications
 - 1.3 Structures and Forms of Documentation for Specification
2. Specification of User Interfaces (GUIs)
 - 2.1 Elements of a GUI Specification
 - 2.2 GUI Elements in Individual Dialog Masks
 - 2.3 Validations
 - 2.4 Navigation Between Dialogue Masks

3. Specification of System Components
 - 3.1 Introduction and Motivation for Components
 - 3.2 Specifying the Structure of Systems and Components
 - 3.3 Specification of Component Behavior
4. Specification of Technical System Interfaces
 - 4.1 Specifying Behavior at Interfaces of Components
 - 4.2 Data Structures at Interfaces of Components
5. Specification of Detailed Business Data Models
 - 5.1 Areas of Application of Technical Data Models
 - 5.2 Detailing the UML Class Diagram
 - 5.3 Checking Class Diagrams with UML Object Diagrams
 - 5.4 Typical Elements in Domain-Oriented Data Models
6. Specification of Data Interfaces with Structured Text
 - 6.1 Structured Text as Exchange Format
 - 6.2 Structure and Structure of XML Documents
 - 6.3 Definition of XML Languages
 - 6.4 Derivation of Class Diagrams From XML formats
 - 6.5 Specifying Web Services with WSDL
7. Specification of Quality Properties
 - 7.1 Quality and Quality Models
 - 7.2 Goal/Question/Metric Method (GQM)

Literature**Compulsory Reading****Further Reading**

- Balzert, H. (1997): Lehrbuch der Software-Technik. Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung. Spektrum, Heidelberg/Berlin.
- Balzert, H. (2004): Lehrbuch der Objektmodellierung. Analyse und Entwurf mit der UML 2. 2. Auflage, Spektrum, Heidelberg/Berlin.
- Ebert, C. (2010): Systematisches Requirements Engineering. Anforderungen ermitteln, spezifizieren, analysieren und verwalten. 3. Auflage, dpunkt.verlag, Heidelberg.
- Eckstein, R./Eckstein, S. (2004): XML und Datenmodellierung. XML-Schema und RDF zur Modellierung von Daten und Metadaten einsetzen. dpunkt.verlag, Heidelberg.
- Evans, E. (2003): Domain-Driven Design. Tackling Complexity in the Heart of Software. Addison-Wesley, Boston.
- Grady, R. B. (1992): Practical Software Metrics for Project Management and Process Improvement. Prentice Hall, Upper Saddle River (NJ).
- Overhage, S./Thomas, P. (2005): WS-Specification. Ein Spezifikationsrahmen zur Beschreibung von Web-Services auf Basis des UDDI-Standards. In: Ferstl, O. K. et al. (Hrsg.): Wirtschaftsinformatik 2005. eEconomy, eGovernment, eSociety. Physica-Verlag, Bamberg, S.1539–1558.
- Sommerville, I. (2007): Software Engineering. 8. Auflage, Pearson, München.
- Turowski, K. (Hrsg.) (2002): Vereinheitlichte Spezifikation von Fachkomponenten. Memorandum des Arbeitskreises 5.10.3 Komponentenorientierte betriebliche Anwendungssysteme. (URL: https://www.researchgate.net/publication/242736875_Vereinheitlichte_Spezifikation_von_Fachkomponenten [letzter Zugriff: 27.02.2017]).
- Wallmüller, E. (2001): Software-Qualitätsmanagement in der Praxis. Software-Qualität durch Führung und Verbesserung von Software-Prozessen. 2. Auflage, Hanser, München.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Project: Software Engineering

Module Code: DLBCSPSE

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Project: Software Engineering)

Contributing Courses to Module

- Project: Software Engineering (DLBCSPSE01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Project Report

Split Exam

Weight of Module

see curriculum

Module Contents

- The knowledge acquired in the computer science modules from study semesters 1-3 is applied in small to medium-sized projects. Implementation is carried out in groups of approximately 3-7 students. Important stages of the software life cycle are covered, and the corresponding artifacts (e.g., specification of requirements, design, implementation, tests, and documentation) are created by the students. The quality assurance of these artefacts is carried out by the tutor and by students from other project groups. The students should learn about both the creation and the quality assurance of artefacts in the SW process.

Learning Outcomes**Project: Software Engineering**

On successful completion, students will be able to

- have experience working on a complex project on a practical scenario of industrial SW development.
- understand the typical risks and pitfalls of large software projects within the framework of project realization and be able to use targeted strategies to minimize risks.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Project: Software Engineering

Course Code: DLBCSPSE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The knowledge acquired in the computer science modules from study semesters 1-3 is applied in small to medium-sized projects. Implementation is carried out in groups of approximately 3-7 students. Important stages of the software life cycle are covered, and the corresponding artifacts (e.g., specification of requirements, design, implementation, tests, and documentation) are created by the students. The quality assurance of these artefacts is carried out by the tutor and by students from other project groups. Students will learn about the creation and the quality assurance of artefacts in the software process.

Course Outcomes

On successful completion, students will be able to

- have experience working on a complex project on a practical scenario of industrial SW development.
- understand the typical risks and pitfalls of large software projects within the framework of project realization and be able to use targeted strategies to minimize risks.

Contents

Literature

Compulsory Reading

Further Reading

- none

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

5. Semester

Seminar: Current Topics in Computer Science

Module Code: DLBCSSCTCS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Seminar: Current Topics in Computer Science)

Contributing Courses to Module

- Seminar: Current Topics in Computer Science (DLBCSSCTCS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Research Essay

Split Exam

Weight of Module

see curriculum

Module Contents

- This seminar deals with current topics of computer science. Students make a dive deep into a specific topic within a sub-discipline of their choice.

Learning Outcomes**Seminar: Current Topics in Computer Science**

On successful completion, students will be able to

- discuss in-depth and insightfully a given topic in the field of computer science.
- write about a certain computer science topic in terms of important characteristics, connections, and insights in the form of a research essay.
- execute the basics of scientific work and implement them in the context of a research essay.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Seminar: Current Topics in Computer Science

Course Code: DLBCSSCTCS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This seminar is an opportunity for students to deepen the broad knowledge they will have gained over the previous four semesters of the study program. Students will choose a topic of specific individual interest that is connected to a sub-discipline of computer science. If a student, for example, is interested in the application of artificial intelligence in a specific context, elaborating context-specific use cases from a literature review can be the theme of the essay. Feedback provided by the tutor will help students strengthen any weaknesses they may have in scientific writing and academic work and prepare students for writing their bachelor thesis.

Course Outcomes

On successful completion, students will be able to

- discuss in-depth and insightfully a given topic in the field of computer science.
- write about a certain computer science topic in terms of important characteristics, connections, and insights in the form of a research essay.
- execute the basics of scientific work and implement them in the context of a research essay.

Contents

- Computer science is a broad subject area with many very different facets, depending on the specific sub-discipline. This seminar will address this diversity by taking up current trends in the context of individually-prepared texts. Each participant must create an essay for this purpose. Possible topics include Java and web development, data modeling and database systems, requirements engineering, and core computer science disciplines like operating systems, computer networks, distributed systems, algorithms, data structures, and programming languages.

Literature**Compulsory Reading****Further Reading**

- Brookshear, G. / Bylow, D. (2014): Computer science: An overview. 12th edition, Pearson, Boston, MA.
- Gruhn, V. / Striemer, R. (Eds.) (2018): The essence of software engineering. Springer International Publishing, Cham.
- Springer. (n.d.) Lecture Notes in Computer Science. Springer, Heidelberg.
- Tardos, E. (Ed.). (n.d.) Journal of the ACM.

Study Format Distance Learning

Study Format Distance Learning	Course Type Seminar

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Research Essay

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

DLBCSSCTCS01

Introduction to Data Protection and Cyber Security

Module Code: DLBCSIDPITS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Ralf Kneuper (Introduction to Data Protection and Cyber Security)

Contributing Courses to Module

- Introduction to Data Protection and Cyber Security (DLBCSIDPITS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Fundamentals of IT Security
- Data Protection
- IT Security Management
- Network and Communication Security

Learning Outcomes**Introduction to Data Protection and Cyber Security**

On successful completion, students will be able to

- explain the terms and concepts of IT security and know the typical procedures and techniques which exist in each area.
- cite the legal regulations on data protection and explain their implementation.
- discuss in-depth IT security management and suitable measures for implementation.
- use their overview knowledge of activities and strategies for IT security in software and system development.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Introduction to Data Protection and Cyber Security

Course Code: DLBCSIDPITS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In this course, the students are familiarized with important concepts from the field of IT security. Basic terms are introduced and discussed, and typical application fields, areas of IT security application, and typical procedures and techniques are introduced and described.

Course Outcomes

On successful completion, students will be able to

- explain the terms and concepts of IT security and know the typical procedures and techniques which exist in each area.
- cite the legal regulations on data protection and explain their implementation.
- discuss in-depth IT security management and suitable measures for implementation.
- use their overview knowledge of activities and strategies for IT security in software and system development.

Contents

1. Fundamentals of Data Protection and Cyber Security
 - 1.1 Conceptual Bases, Protection Goals
 - 1.2 Attacks and Threats
 - 1.3 Security Strategy
 - 1.4 Legal Regulations
2. Data Protection
 - 2.1 Data Protection as a Personal Right
 - 2.2 Basic Principles of Data Protection
 - 2.3 EU General Data Protection Regulation
 - 2.4 Further International Regulations on Data Protection
 - 2.5 Cross-Border Data Flow
 - 2.6 Data Protection in Everyday Life

3. Basic Functions of Cyber Security and Their Implementation
 - 3.1 Identification and Authentication
 - 3.2 Rights Management
 - 3.3 Rights Check
 - 3.4 Preservation of Evidence
4. Cyber Security Management
 - 4.1 Basic Concepts and Standards in Cyber Security Management
 - 4.2 Series of Standards ISO 2700x
5. Cyber Security Management in Everyday Life
 - 5.1 Password Management
 - 5.2 Data Backup
 - 5.3 Email Security
 - 5.4 Protection Against Viruses and Other Malware
 - 5.5 Protection Against Social Engineering Attacks
6. Network and Communication Security
 - 6.1 Firewall Technology
 - 6.2 Network Separation
 - 6.3 Security in WLAN, Mobile Networks, Bluetooth, and NFC
7. Cyber Security in the Development of Software and Systems
 - 7.1 Protection of the Development Environment
 - 7.2 Secure Development
 - 7.3 Common Criteria

Literature**Compulsory Reading****Further Reading**

- Eckert, C. (2014): IT-Sicherheit. Konzepte – Verfahren – Protokolle. 9. Auflage, De Gruyter, München.
- Poguntke, W. (2013): Basiswissen IT-Sicherheit. Das Wichtigste für den Schutz von Systemen & Daten. 3. Auflage, W3l, Dortmund.
- Witt, B. C. (2010): Datenschutz kompakt und verständlich. 2. Auflage, Vieweg+Teubner, Wiesbaden.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

DLBCSIDPITS01

Cryptography

Module Code: DLBCSCT

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Cryptography)

Contributing Courses to Module

- Cryptography (DLBCSCT01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Exam, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Protection targets, vulnerabilities, and threats
- Foundations of cryptology and its core components
- Basic cryptographic applications
- Authentication
- Single computer security
- Security communication network
- Security E-Commerce
- Secure software development

Learning Outcomes**Cryptography**

On successful completion, students will be able to

- give an overview of different classes of cryptographic systems.
- give a basic description of symmetric cryptographic methods, in particular One-Time Pad, DES, and AES, and describe their operating principles by means of simple, concrete examples.
- describe the basic hash functions.
- describe basic asymmetric cryptographic methods, especially RSA, and their operating principles by means of simple, concrete examples.
- describe the areas of application of cryptographic procedures and their application scenarios.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology fields

Cryptography

Course Code: DLBCSCT01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course covers basic and targeted in-depth knowledge of cryptographic processes and the practical use of cryptographic systems. After an overview of cryptographic methods, hash functions, symmetric methods, and asymmetric methods are presented. The theoretical basics of selected procedures are taught and practically explained using simple examples. In addition, areas of application and application scenarios for cryptographic procedures are presented.

Course Outcomes

On successful completion, students will be able to

- give an overview of different classes of cryptographic systems.
- give a basic description of symmetric cryptographic methods, in particular One-Time Pad, DES, and AES, and describe their operating principles by means of simple, concrete examples.
- describe the basic hash functions.
- describe basic asymmetric cryptographic methods, especially RSA, and their operating principles by means of simple, concrete examples.
- describe the areas of application of cryptographic procedures and their application scenarios.

Contents

1. Protection Goals, Vulnerabilities, and Threats
 - 1.1 Protection Goals
 - 1.2 Vulnerabilities and Threats
2. Foundations of Cryptology and its Core Components
 - 2.1 Encoding
 - 2.2 Symmetrical Encryption
 - 2.3 Asymmetric Encryption
 - 2.4 One-way Functions and Cryptographic Hash Functions

3. Basic Cryptographic Applications
 - 3.1 Key exchange and Hybrid Processes
 - 3.2 Digital Signature
 - 3.3 Message Authentication Code
 - 3.4 Steganographic Methods
4. Authentication
 - 4.1 Passwords and Public-Key-Certificates
 - 4.2 Challenge-Response-Procedure and Zero-Knowledge-Procedure
 - 4.3 Biometric Methods
 - 4.4 Authentication in Distributed Systems
 - 4.5 Identities Through Smartcards
5. Security of Single Computers
 - 5.1 Malware and Cookies
 - 5.2 Some Special Features of Operating Systems
 - 5.3 Web Server Security
6. Security in Communication Networks
 - 6.1 Security Problems and Defense Concepts
 - 6.2 Internet Standards for Communication Security
 - 6.3 Identity and Anonymity
 - 6.4 Security in Mobile and Wireless Communications
7. Security in E-Commerce
 - 7.1 Email Security
 - 7.2 Online Banking and Online Payments
 - 7.3 Electronic Money
8. Secure Software Development
 - 8.1 Threat Modeling
 - 8.2 Secure Software Design
 - 8.3 Techniques for Safe Programming

Literature**Compulsory Reading****Further Reading**

- Baumann, U./Franz, E./Pfitzmann, A. (2014): Kryptographische Systeme. Springer Vieweg, Wiesbaden.
- Beutelspacher, A. (2014): Kryptologie. Eine Einführung in die Wissenschaft vom Verschlüsseln, Verbergen und Verheimlichen. 10. Auflage, Springer Spektrum, Wiesbaden.
- Eckert, C. (2014): IT-Sicherheit. Konzepte – Verfahren – Protokolle. 9. Auflage, De Gruyter Oldenbourg, München.
- Ertel, W. (2010): Angewandte Kryptographie. 4. Auflage, Hanser, München.
- Spitz, S./Pramateftakis, M./Swoboda, J. (2011): Kryptographie und IT-Sicherheit. Grundlagen und Anwendungen. 2. Auflage, Vieweg+Teubner; Wiesbaden.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☐ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Salesforce Platform Development

Module Code: DLSFPD

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Maik Günther (Salesforce Platform App Builder) / Prof. Dr. Maik Günther (Salesforce Platform Developer)

Contributing Courses to Module

- Salesforce Platform App Builder (DLSFPD01)
- Salesforce Platform Developer (DLSFPD02)

Module Exam Type

Module Exam

Split Exam

Salesforce Platform App Builder

- Study Format "Distance Learning": Written Assessment: Project Report

Salesforce Platform Developer

- Study Format "Distance Learning": Oral Project Report

Weight of Module

see curriculum

Module Contents

Salesforce Platform App Builder

Using the learning platform Trailhead students will learn the fundamentals of Salesforce. At the end of the course, the students will be able to design, build and deploy custom applications. This course prepares them for the Salesforce Platform App Builder Certification.

Salesforce Platform Developer

Using the learning platform Trailhead students will learn how to develop own applications, built from various parts of the Salesforce platform. At the end of the course they will be able to use Apex, Visualforce and basic Lightning components. This course prepares the students for the Salesforce Platform Developer I Certification.

Learning Outcomes

Salesforce Platform App Builder

On successful completion, students will be able to

- define what Salesforce and customer relationship management is,
- design the data model, user interface, and business logic for custom applications,
- customize applications for mobile use,
- design reports and dashboards,
- manage application security and deploy custom applications.

Salesforce Platform Developer

On successful completion, students will be able to

- develop own applications using Apex and basic Lightning components,
- write SOSL, SOQL and DML statements,
- use Visualforce to build custom user interfaces for mobile and web apps,
- build reusable, performant components that follow modern web standards,
- use the built-in testing framework to test Apex and Visualforce.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Marketing & Sales

Links to other Study Programs of IUBH

All Bachelor Programs in the Marketing & Communication field

Salesforce Platform App Builder

Course Code: DLSFPD01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Salesforce is the most used software solution for customer relationship management worldwide. This solution can be customized and personalized for the needs of customers, partners and employees. Using the learning platform Trailhead, students will learn independently the fundamentals of Salesforce and the development of customized application. This course prepares students for the Salesforce Platform App Builder Certification.

Course Outcomes

On successful completion, students will be able to

- define what Salesforce and customer relationship management is,
- design the data model, user interface, and business logic for custom applications,
- customize applications for mobile use,
- design reports and dashboards,
- manage application security and deploy custom applications.

Contents

- The content on the learning platform focuses on the features and functionality to design, build and deploy custom applications. The content also provides knowledge to define business logic and process automation declaratively. Furthermore, the design and management of the correct data models and the customization of applications for individual needs is included in this course. Thus, the content of this course enables to automate repetitive tasks and to optimize processes in customer organizations.

Literature**Compulsory Reading****Further Reading**

- Gupta, R. (2019): Salesforce Platform App Builder Certification. A Practical Study Guide. 1st ed., Apress.
- Weinmeister, P. (2019): Practical Salesforce Development Without Code. Building Declarative Solutions on the Salesforce Platform. 2nd ed., Apress, Berkeley.
- Shaalan, S. (2020): Salesforce for Beginners. A step-by-step guide to creating, managing, and automating sales and marketing processes. Packt Publishing, Birmingham.
- Benioff, M./Langley, M. (2019): Trailblazer. The Power of Business as the Greatest Platform for Change. 1st ed.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Salesforce Platform Developer

Course Code: DLSFPD02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The Salesforce platform not only forms the foundation of core Salesforce products like Sales Cloud and Service Cloud, but it is also possible to build own functionalities and own applications. Using the learning platform Trailhead, students will learn how to use the programmatic pillars of the Salesforce platform: Lightning components, Apex and Visualforce. This course prepares students for the Salesforce Platform Developer I Certification.

Course Outcomes

On successful completion, students will be able to

- develop own applications using Apex and basic Lightning components,
- write SOSL, SOQL and DML statements,
- use Visualforce to build custom user interfaces for mobile and web apps,
- build reusable, performant components that follow modern web standards,
- use the built-in testing framework to test Apex and Visualforce.

Contents

- The content on the learning platform focuses on the development of own functionality and own applications, built from various parts of the Salesforce platform. The content enables to use the programmatic elements Lightning components, Apex and Visualforce. Furthermore, knowledge is provided for data modeling, process automation, user interface design, testing and deployment. Thus, the content of this course enables to extend Salesforce by individual applications to cover the needs in customer organizations.

Literature

Compulsory Reading

Further Reading

- Salesforce (2020): Developer Documentation. (URL: <https://developer.salesforce.com/docs/> [accessed: 12.12.2020])

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Oral Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

DLSFPD02

Mobile Software Engineering

Module Code: DLBCSEMSE

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Mobile Software Engineering I) / N.N. (Mobile Software Engineering II)

Contributing Courses to Module

- Mobile Software Engineering I (DLBCSEMSE01)
- Mobile Software Engineering II (DLBCSEMSE02)

Module Exam Type

Module Exam

Split Exam

Mobile Software Engineering I

- Study Format "Distance Learning": Exam, 90 Minutes

Mobile Software Engineering II

- Study Format "Distance Learning": Written Assessment: Project Report

Weight of Module

see curriculum

Module Contents**Mobile Software Engineering I**

- Basics of mobile software development
- Android system architecture
- Development environment
- Core components of an Android app
- Interaction between application components
- Advanced techniques

Mobile Software Engineering II

Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task.

Learning Outcomes**Mobile Software Engineering I**

On successful completion, students will be able to

- recognize the differences and peculiarities of software development for mobile systems and explain them.
- differentiate between different activities, roles, and risks in the creation, operation, and maintenance of mobile software systems.
- explain and differentiate between the architecture and technical features of the Android platform.
- independently create mobile software systems to solve concrete problems for the “Android” platform.

Mobile Software Engineering II

On successful completion, students will be able to

- independently design and create a prototype of a small mobile application to solve a specific problem.
- recognize typical problems and challenges in the practical implementation of small mobile applications.
- document the conception and implementation of small, independently designed and implemented mobile applications.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programs in the IT & Technology fields

Mobile Software Engineering I

Course Code: DLBCSEMSE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using the mobile platform "Android" as an example, it will be demonstrated how the programming of mobile applications (apps) differs from the development of browser-based information systems, which technologies and programming concepts are typically used, and which typical challenges there are in app development for industrial applications.

Course Outcomes

On successful completion, students will be able to

- recognize the differences and peculiarities of software development for mobile systems and explain them.
- differentiate between different activities, roles, and risks in the creation, operation, and maintenance of mobile software systems.
- explain and differentiate between the architecture and technical features of the Android platform.
- independently create mobile software systems to solve concrete problems for the "Android" platform.

Contents

1. Basics of Mobile Software Development
 - 1.1 Special Features of Mobile Devices
 - 1.2 Special Features of Mobile Software Development
 - 1.3 Classification of Mobile Devices
 - 1.4 The Android Platform
2. Android System Architecture
 - 2.1 The Android System
 - 2.2 Safety and Security
 - 2.3 Communication with Networks
3. Development Environment
 - 3.1 Android Studio
 - 3.2 First App and Emulator Test
 - 3.3 Application Deployment

4. Core Components of an Android App
 - 4.1 Overview of the Components of an Android App
 - 4.2 Activities, Layouts, and Views
 - 4.3 Resources
 - 4.4 Summary in an App
 - 4.5 Graphic Design
5. Interaction Between Application Components
 - 5.1 Intents
 - 5.2 Services
 - 5.3 Broadcast Receiver
6. Advanced Techniques
 - 6.1 Threading
 - 6.2 Application Memory

Literature

Compulsory Reading

Further Reading

- Becker, A./Pant, M. (2015): Android 5. Programmieren für Smartphones und Tablets. 4. Auflage, dpunkt.verlag, Heidelberg.
- Eason, J. (2014): Android Studio 1.0. (URL: <https://android-developers.googleblog.com/2014/12/android-studio-10.html>)
- Eason, J. (2014): Android Studio 1.0. (URL: <https://android-developers.googleblog.com/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Franke, F./Ippen, J. (2012): Apps mit HTML5 und CSS3. Galileo Computing, Bonn.
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>) [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Mobile Software Engineering II

Course Code: DLBCSEMSE02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using the knowledge gained in the course "Mobile Software Engineering using the Android platform as an example", students independently create a mobile application and document its conception and implementation.

Course Outcomes

On successful completion, students will be able to

- independently design and create a prototype of a small mobile application to solve a specific problem.
- recognize typical problems and challenges in the practical implementation of small mobile applications.
- document the conception and implementation of small, independently designed and implemented mobile applications.

Contents

- Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task. Possible topics are, for example:
- A radio app to improve the exchange between listeners and stations in general, and listeners and radio presenters in particular.
- An app that allows a group of board game fans to better organize their regular evening game.
- An app that theses supervisors at IUBH can use to improve their supervision processes.

Literature**Compulsory Reading****Further Reading**

- Eason, J. (2014): Android Studio 1.0. (URL: <http://android-developers.blogspot.de/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>)
- Google Inc. (Hrsg.) (2015a): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015b): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015c): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).
- Hipp, Wyrick & Company, Inc. (Hrsg.) (2015): SQLite Webseite. (URL: <http://sqlite.org/index.html> [letzter Zugriff: 12.06.2015]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Big Data and Cloud Technologies

Module Code: DLBCSEBDCT

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	- none - DLBCSDMDS01	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Christian Müller-Kett (Big Data Technologies) / Prof. Dr. Thomas Zöller (Cloud Computing)

Contributing Courses to Module

- Big Data Technologies (DLBDSBDT01)
- Cloud Computing (DLBDSCC01)

Module Exam Type

Module Exam

Split Exam

Big Data Technologies

- Study Format "Distance Learning": Exam, 90 Minutes

Cloud Computing

- Study Format "Distance Learning": Exam, 90 Minutes

Weight of Module

see curriculum

Module Contents**Big Data Technologies**

- Data types and data sources
- Text-based and binary data formats
- Distributed systems
- Streaming frameworks
- NoSQL approach to data storage

Cloud Computing

- Cloud computing fundamentals
- Relevant enabling technologies for cloud computing
- Introduction to serverless computing
- Established cloud platforms
- Cloud offerings for data science and analytics

Learning Outcomes**Big Data Technologies**

On successful completion, students will be able to

- name types and sources of data.
- understand text-based and binary data formats.
- analyze the requirements and constraints of distributed analysis systems.
- evaluate the applications of streaming frameworks.
- describe the motivation for NoSQL data stores and categorize pertaining established concepts.

Cloud Computing

On successful completion, students will be able to

- understand the fundamentals of cloud computing and cloud service models.
- recognize enabling technologies that underlie current cloud offerings.
- cite the principles of serverless computing.
- analyze characteristics of established cloud offerings.
- describe cloud options for data science and machine learning

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Big Data Technologies

Course Code: DLBDSBDT01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSDMDS01

Course Description

Data are often considered the “new oil”, the raw material from which value is created. To harness the power of data, the data need to be stored and processed on a technical level. This course introduces the four “Vs” of data, as well as typical data sources and types. The course discusses the most common data storage formats encountered in modern systems, focusing both on text-based as well as binary data formats. Handling large amounts of data poses significant challenges for the underlying infrastructure. The course discusses the most important distributed and streaming data handling frameworks which are used in leading edge applications.

Course Outcomes

On successful completion, students will be able to

- name types and sources of data.
- understand text-based and binary data formats.
- analyze the requirements and constraints of distributed analysis systems.
- evaluate the applications of streaming frameworks.
- describe the motivation for NoSQL data stores and categorize pertaining established concepts.

Contents

1. Data Types and Data Sources
 - 1.1 The 4Vs of data: volume, velocity, variety, veracity
 - 1.2 Data sources
 - 1.3 Data types
2. Text-Based and Binary Data Formats
 - 2.1 Simple formats: CSV, YAML
 - 2.2 XML
 - 2.3 JSON
 - 2.4 Hierarchical data format 5 (HDF 5)
 - 2.5 Apache Parquet
 - 2.6 Apache Arrow

3. NoSQL data stores
 - 3.1 Introduction and motivation
 - 3.2 Approaches and technical concepts
4. Distributed Systems
 - 4.1 Hadoop & MapReduce
 - 4.2 Hadoop file system (HDFS)
 - 4.3 Spark
 - 4.4 DASK
5. Streaming Frameworks
 - 5.1 Spark streaming
 - 5.2 Kafka

Literature

Compulsory Reading

Further Reading

- Karau, H., Konwinski, A., Wendell, A., & Zaharia, M. (2015). Learning spark: Lightning-fast data analysis. Sebastopol, CA: O'Reilly.
- Kleppmann, M. (2017). Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. Sebastopol, CA: O'Reilly.
- Narkhede, N., Shapira, G., & Palino, T. (2017). Kafka: The definitive guide: Real-time data and stream processing at scale. Sebastopol, CA: O'Reilly.
- Psaltis, A. (2017). Streaming data: Understanding the real-time pipeline. Shelter Island, NY: Manning.
- White, T. (2015). Hadoop: The definitive guide: Storage and analysis at Internet scale. Sebastopol, CA: O'Reilly.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Cloud Computing

Course Code: DLBDSCC01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Many of the recent advances in data science, particularly machine learning and artificial intelligence, rely on comprehensive data storage and computing power. Cloud computing is one way of providing that power in a scalable way, without considerable upfront investment in hardware and software resources. This course introduces the area of cloud computing together with its enabling technologies. Moreover, the most cutting-edge advances like serverless computing and storage are illustrated. Finally, a thorough overview on popular cloud offerings, especially in regard to analytics capabilities, is given.

Course Outcomes

On successful completion, students will be able to

- understand the fundamentals of cloud computing and cloud service models.
- recognize enabling technologies that underlie current cloud offerings.
- cite the principles of serverless computing.
- analyze characteristics of established cloud offerings.
- describe cloud options for data science and machine learning

Contents

1. Introduction to Cloud Computing
 - 1.1 Fundamentals of Cloud computing
 - 1.2 Cloud Service Models
 - 1.3 Benefits and Risks
2. Enabling Technology
 - 2.1 Virtualization and Containerization
 - 2.2 Storage Technology
 - 2.3 Networks and RESTful Services
3. Serverless Computing
 - 3.1 Introduction to Serverless Computing
 - 3.2 Benefits
 - 3.3 Limitations

4. Established Cloud Platforms
 - 4.1 Google Cloud Platform
 - 4.2 Amazon Web Services
 - 4.3 Microsoft Azure
5. Data Science in the Cloud
 - 5.1 Google Data Science and Machine Learning Services
 - 5.2 Amazon Web Services Data Science and Machine Learning Services
 - 5.3 Microsoft Azure Data Science and Machine Learning Services

Literature

Compulsory Reading

Further Reading

- Chapin, J., & Roberts, M. (2017). What is serverless? Sebastopol, CA: O'Reilly Media.
- Goessling, S., & Jackson, K. L. (2018). Architecting cloud computing solutions. Birmingham: Packt Publishing.
- Kavis, M. J. (2014). Architecting the cloud: Design decisions for cloud computing service models (SaaS, PaaS, and IaaS). Hoboken, NJ: Wiley.
- Mahmood, Z., Puttini, R., & Erl, T. (2013). Cloud computing: Concepts, technology & architecture. Boston, MA: Prentice Hall.
- Rafaels, R. (2018). Cloud computing (2nd ed.). Scotts Valley, CA: CreateSpace Independent Publishing Platform.
- Sehgal, N. K., & Bhatt, P. C. P. (2018). Cloud computing: Concepts and practices. Cham: Springer.
- Zonooz, P. Farr, E., Arora, K., & Laszewski, T. (2018). Cloud native architectures. Birmingham: Packt Publishing.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Business Intelligence

Module Code: DLBCSEBI

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Sebastian Werning (Business Intelligence) / Prof. Dr. Sebastian Werning (Project: Business Intelligence)

Contributing Courses to Module

- Business Intelligence (DLBCSEBI01)
- Project: Business Intelligence (DLBCSEBI02)

Module Exam Type

Module Exam

Split Exam

Business Intelligence

- Study Format "Distance Learning": Exam, 90 Minutes

Project: Business Intelligence

- Study Format "Distance Learning": Written Assessment: Project Report

Weight of Module

see curriculum

Module Contents**Business Intelligence**

- Basics of mobile software development
- Android system architecture
- Development environment
- Core components of an Android app
- Interaction between application components
- Advanced techniques

Project: Business Intelligence

Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task.

Learning Outcomes**Business Intelligence**

On successful completion, students will be able to

- explain the motivation, use cases, and basics of Business Intelligence.
- identify and explain techniques and methods for providing and modeling data, as well as types of data relevant to BI, differentiating between them.
- explain techniques and methods for the generation and storage of information and independently select suitable methods on the basis of concrete requirements.

Project: Business Intelligence

On successful completion, students will be able to

- independently design a solution to a practical problem in the field of Business Intelligence in order to then implement a prototype and document the results.
- identify and explain typical problems and challenges in the design and practical implementation of small BI solutions.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology fields

Business Intelligence

Course Code: DLBCSEBI01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Business Intelligence (BI) is used to obtain information from company data that is relevant for targeted corporate management and the optimization of business activities. This course introduces and discusses techniques, procedures, and models for data provision, information generation, and analysis, as well the distribution of the information obtained. You will then be able to explain the various subject areas of data warehousing and independently select methods and techniques to meet specific requirements.

Course Outcomes

On successful completion, students will be able to

- explain the motivation, use cases, and basics of Business Intelligence.
- identify and explain techniques and methods for providing and modeling data, as well as types of data relevant to BI, differentiating between them.
- explain techniques and methods for the generation and storage of information and independently select suitable methods on the basis of concrete requirements.

Contents

1. Motivation and Conceptualization
 - 1.1 Motivation and Historical Development
 - 1.2 BI as a Framework
2. Data Provision
 - 2.1 Operative and Dispositive Systems
 - 2.2 The Data Warehouse Concept
 - 2.3 Architectural Variations
3. Data Warehouse
 - 3.1 ETL Process
 - 3.2 DWH and Data Mart
 - 3.3 ODS and Metadata

4. Modelling of Multidimensional Data Spaces

- 4.1 Data Modeling
- 4.2 OLAP Cubes
- 4.3 Physical Storage
- 4.4 Star and Snowflake Scheme
- 4.5 Historicization

5. Analysis Systems

- 5.1 Free Data Research and OLAP
- 5.2 Reporting Systems
- 5.3 Model-Based Analysis Systems
- 5.4 Concept-Oriented Systems

6. Distribution and Access

- 6.1 Information Distribution
- 6.2 Information Access

Literature

Compulsory Reading

Further Reading

- Bachmann, R./Kemper, G. (2011): Raus aus der BI-Falle. Wie Business Intelligence zum Erfolg wird. 2. Auflage, mitp, Heidelberg.
- Bauer, A./Günzel, H. (2008): Data Warehouse Systeme. Architektur, Entwicklung, Anwendung. 3. Auflage, dpunkt.verlag, Heidelberg.
- Betz, R. (2015): Werde Jäger des verlorenen Schatzes. In: Immobilienwirtschaft, Heft 5, S. 1614–1164. (URL <https://www.haufe.de/download/immobilienwirtschaft-ausgabe-052015-immobilienwirtschaft-fachmagazin-fuer-management-recht-praxis-303530.pdf> [letzter Zugriff: 27.02.2017]).
- Bodendorf, F. (2006): Daten- und Wissensmanagement. 2. Auflage, Springer, Berlin.
- Chamoni, P./Gluchowski, P. (Hrsg.) (2006): Analytische Informationssysteme Business Intelligence-Technologien und -Anwendungen. Springer, Berlin.
- Engels, C. (2008): Basiswissen Business Intelligence. W3L, Herdecke/Witten.
- Gansor, T./Totok, A./Stock, S. (2010): Von der Strategie zum Business Intelligence Competency Center (BICC). Konzeption – Betrieb – Praxis. Hanser, München.
- Gluchowski, P./Gabriel, R./Dittmar, C. (2008): Management Support Systeme und Business Intelligence. Computergestützte Informationssysteme für Fach- und Führungskräfte. 2. Auflage, Springer, Berlin/Heidelberg.
- Grothe, M. (2000): Business Intelligence. Aus Informationen Wettbewerbsvorteile gewinnen. Addison-Wesley, München.
- Gutenberg, E. (1983): Grundlagen der Betriebswirtschaft, Band 1. Die Produktion. 18. Auflage, Springer, Berlin/Heidelberg/New York.
- Hannig, U. (Hrsg.) (2002): Knowledge Management und Business Intelligence. Springer, Berlin.
- Hansen, H.-R./Neumann, G. (2001): Wirtschaftsinformatik I. Grundlagen betrieblicher Informationsverarbeitung. 8. Auflage, Lucius & Lucius UTB, Stuttgart.
- Humm, B./Wietek, F. (2005): Architektur von Data Warehouses und Business Intelligence Systemen. In: Informatik Spektrum, S. 3–14. (URL: https://www.fbi.h-da.de/fileadmin/personal/b.humm/Publikationen/Humm__Wietek_-_Architektur_DW__Informatik-Spektrum_2005-01_.pdf [letzter Zugriff: 27.02.2017]).
- Kemper, H.-G./Baars, H./Mehanna, W. (2010): Business Intelligence – Grundlagen und praktische Anwendungen. Eine Einführung in die IT-basierte Managementunterstützung. 3. Auflage, Vieweg+Teubner, Stuttgart.
- Turban, E. et al. (2010): Business Intelligence. A Managerial Approach. 2. Auflage, Prentice Hall, Upper Saddle River (NJ).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Project: Business Intelligence

Course Code: DLBCSEBI02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using well-known methods and techniques from the field of Business Intelligence, students will work independently on a practical question in this course. At the end of the course you will be able to independently design and prototype Business Intelligence applications based on concrete requirements.

Course Outcomes

On successful completion, students will be able to

- independently design a solution to a practical problem in the field of Business Intelligence in order to then implement a prototype and document the results.
- identify and explain typical problems and challenges in the design and practical implementation of small BI solutions.

Contents

- Implementation and documentation of practical questions regarding the use of Business Intelligence applications. Typical scenarios are, for example, "Management of BI projects", "Design of multidimensional data models" and "Prototypical implementation of small BI applications".

Literature

Compulsory Reading

Further Reading

- Brenner, W./Uebersnickel, F. (2015): Design Thinking. Das Handbuch. Frankfurter Allgemeine Buch, Frankfurt a. M.
- Brown, T. (2008): Design Thinking. In: Harvard Business Review, Heft Juni, S. 84–95.
- Meinel, C./Weinberg, U./Krohn, T. (Hrsg.) (2015): Design Thinking Live. Wie man Ideen entwickelt und Probleme löst. Murmann, Hamburg.
- Uebersnickel, F./Brenner, W. (2016): Design Thinking. In: Hoffmann, C. P. et al. (Hrsg.): Business Innovation: Das St. Galler Modell. Springer, Wiesbaden, S. 243–265.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Software Engineering with Python

Module Code: DLBCSESEWP

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	- DLBDSIPWP01 or DLBDSIPWP01_D; DLBDSOOFPP01 or IOBP01 - none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Max Pumperla (Object oriented and functional programming in Python) / Prof. Dr. Max Pumperla (Data Science Software Engineering)

Contributing Courses to Module

- Object oriented and functional programming in Python (DLBDSOOFPP01)
- Data Science Software Engineering (DLBDSSE01)

Module Exam Type

Module Exam	Split Exam
	<u>Object oriented and functional programming in Python</u> - Study Format "Distance Learning": Portfolio (50) <u>Data Science Software Engineering</u> - Study Format "Distance Learning": Exam, 90 Minutes (100)

Weight of Module

see curriculum

Module Contents**Object oriented and functional programming in Python**

- This course introduces the students to the advanced programming concepts of object orientation and functional programming and how they are realized in the Python programming language.

Data Science Software Engineering

- Traditional project management
- Agile project management
- Testing
- Software development paradigms
- From model to production

Learning Outcomes**Object oriented and functional programming in Python**

On successful completion, students will be able to

- explain basic notions in object-oriented programming such as functions and classes.
- understand object-oriented programming concepts and their relation to software design and engineering.
- describe advanced function concepts in Python.
- recognize important ideas from functional programming.
- recall important libraries for functional programming in Python.

Data Science Software Engineering

On successful completion, students will be able to

- understand the concept of project management approaches.
- apply agile approaches in software development.
- create automated software tests.
- understand various software development paradigms.
- evaluate the necessary steps to bring models into a production environment.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Data Science & Artificial Intelligence.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Object oriented and functional programming in Python

Course Code: DLBDSOOFPP01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course builds upon basic knowledge of Python programming (Introduction to Programming with Python, DLBDSIPWP) and is concerned with the exposition of advanced Python programming concepts. To this end, important notions of object-oriented programming like classes and objects and pertaining design principles are outlined. Starting from an in-depth discussion of advanced features of Python functions, functional programming concepts and their implementation in Python are conveyed.

Course Outcomes

On successful completion, students will be able to

- explain basic notions in object-oriented programming such as functions and classes.
- understand object-oriented programming concepts and their relation to software design and engineering.
- describe advanced function concepts in Python.
- recognize important ideas from functional programming.
- recall important libraries for functional programming in Python.

Contents

- This course provides students with a thorough introduction to important notions and concepts from the domain of object-oriented programming such as classes, objects, abstraction, encapsulation, inheritance, polymorphism, composition, and delegation. Additionally, the functional programming paradigm and pertaining ideas like functions as first class objects, decorators, pure functions, immutability and higher order functions are conveyed. Pursuant to the portfolio course type, the aforementioned concepts and ideas are explored by hands-on programming projects.

Literature**Compulsory Reading****Further Reading**

- Lott, S. F. (2018): Functional Python programming: Discover the power of functional programming, generator functions, lazy evaluation, the built-in itertools library, and monads. 2nd ed., Packt Publishing, Birmingham.
- Lutz, M. (2013): Learning Python. 5th ed., O'Reilly, Sebastopol, CA.
- Phillips, D. (2018): Python 3 object-oriented programming: Build robust and maintainable software with object-oriented design patterns in Python 3.8. 3rd ed., Packt Publishing, Birmingham.
- Ramalho, L. (2015): Fluent Python: Clear, concise, and effective programming. O'Reilly, Sebastopol, CA.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Portfolio

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

Data Science Software Engineering

Course Code: DLBDSSE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBDSIPWP01 or DLBDSIPWP01_D; DLBDSOOFPP01 or IOBP01

Course Description

A core part of data science is creating value from data. This means not only the creation of sophisticated predictive models but also the development of these models according to modern software development principles. This course gives a detailed overview of the relevant methods and paradigms which data scientists need to know in order to develop enterprise-grade models. This course discusses traditional and agile project management techniques, highlighting both the Kanban and Scrum approaches. It explores relevant software development paradigms such as test-driven development, pair programming, mob programming, and extreme programming. Special focus is given to the topic of testing and the consideration of how to bring a model into a production environment.

Course Outcomes

On successful completion, students will be able to

- understand the concept of project management approaches.
- apply agile approaches in software development.
- create automated software tests.
- understand various software development paradigms.
- evaluate the necessary steps to bring models into a production environment.

Contents

1. Traditional Project Management
 - 1.1 Requirements engineering
 - 1.2 Waterfall model
 - 1.3 Rational unified process
2. Agile Project Management
 - 2.1 Criticism of the waterfall model
 - 2.2 Introduction to SCRUM
 - 2.3 Introduction to Kanban

3. Testing
 - 3.1 Why testing?
 - 3.2 Unit tests
 - 3.3 Integration tests
 - 3.4 Performance monitoring
4. Software Development Paradigms
 - 4.1 Test-driven development (TDD)
 - 4.2 Pair programming
 - 4.3 Mob programming
 - 4.4 Extreme programming
5. From Model to Production
 - 5.1 Continuous delivery
 - 5.2 Continuous integration
 - 5.3 Building a scalable environment

Literature

Compulsory Reading

Further Reading

- Farcic, V. (2016): The DevOps 2.0 toolkit: Automating the continuous deployment pipeline with containerized microservices. CreateSpace Independent Publishing Platform, Scotts Valley, CA.
- Humble, J./Farley, D. (2010): Continuous delivery: Reliable software releases through build, test, and deployment automation. Addison-Wesley Professional, Boston, MA.
- Humble, J./Molesky, J./O'Reilly, B. (2015): Lean enterprise. O'Reilly Publishing, Sebastopol, CA.
- Hunt, A./Thomas, D. (1999): The pragmatic programmer. From journeyman to master. Addison-Wesley, Reading, MA.
- Martin, R. C. (2008): Clean code. Prentice Hall, Boston, MA.
- Morris, K. (2016): Infrastructure as code. O'Reilly Publishing, Sebastopol, CA.
- Richardson, L./Ruby, S. (2007): RESTful web services. O'Reilly Publishing, Sebastopol, CA.
- Senge, P. (1990): The fifth discipline: The art and practice of the learning organization. Broadway Business, New York, NY.

Study Format Distance Learning

Study Format Distance Learning	Course Type Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

IT project and architecture management

Module Code: DLBCSEITPAM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (IT Project Management) / N.N. (IT Architecture Management)

Contributing Courses to Module

- IT Project Management (DLBCSEITPAM01)
- IT Architecture Management (DLBCSEITPAM02)

Module Exam Type

Module Exam

Split Exam

IT Project Management

- Study Format "Distance Learning": Exam, 90 Minutes

IT Architecture Management

- Study Format "Distance Learning": Exam, 90 Minutes

Weight of Module

see curriculum

Module Contents

IT Project Management

- Basic terms and foundations of IT project management
- Large and small planning techniques
- Techniques for prioritization, cost-estimation, and project controlling
- Techniques for stakeholder, communication, and risk management
- Organization and structure in IT project management
- Schools of thought in IT project management

IT Architecture Management

- Basic terms and foundations of IT enterprise architectures management
- IT application portfolio management
- Architecture governance
- Modeling of IT enterprise architectures
- Frameworks using TOGAF as an example
- Reference models and sample catalogues

Learning Outcomes

IT Project Management

On successful completion, students will be able to

- explain and differentiate between the basic principles and tasks of IT project management.
- explain the important practical techniques and methods necessary for the implementation of IT project management.
- describe the basic procedural models and explain their advantages and disadvantages as well as their possible applications.
- identify possible project risks on the basis of given practical scenarios and select suitable measures from IT project management in order to minimize them in a targeted manner.

IT Architecture Management

On successful completion, students will be able to

- describe and explain the basic principles of IT strategy, governance, and architecture management, differentiating between them.
- explain and differentiate the typical activities of IT architecture management, their interrelationships, and their dependencies.
- explain suitable models of IT architecture management, distinguish between them, and explain their intended purpose.
- explain and describe selected IT architectural frameworks as well as reference models and sample catalogues.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

IT Project Management

Course Code: DLBCSEITPAM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In this course, typical problems in the management of Software projects are discussed and the methods and techniques used to address challenges conveyed. In addition, standard procedural models for IT project management are explained and their strengths and weaknesses specifically identified.

Course Outcomes

On successful completion, students will be able to

- explain and differentiate between the basic principles and tasks of IT project management.
- explain the important practical techniques and methods necessary for the implementation of IT project management.
- describe the basic procedural models and explain their advantages and disadvantages as well as their possible applications.
- identify possible project risks on the basis of given practical scenarios and select suitable measures from IT project management in order to minimize them in a targeted manner.

Contents

1. Basics Terms and Foundations of IT Project Management
 - 1.1 Definition of a Project and Types of IT Projects
 - 1.2 IT Project Lifecycle
 - 1.3 Multi-Project Management – The Project in the Context of the Organization
2. Planning Techniques
 - 2.1 Large-Scale Planning: Milestones, Sub-tasks, and Work Packages
 - 2.2 Large-Scale Planning: Gantt Charts
 - 2.3 Planning and Organization of Work Packages: Kanban Board
3. Prioritization, Estimation of Costs, Project Controlling
 - 3.1 Prioritization
 - 3.2 Estimation of Costs
 - 3.3 Project Controlling

4. Stakeholder, Communication and Risk Management
 - 4.1 Stakeholder Management
 - 4.2 Communication Management
 - 4.3 Risk Management
5. Organization and Structure in IT Project Management
 - 5.1 Overview and Levels of Management from PRINCE2
 - 5.2 Management Processes in PRINCE2
 - 5.3 Pragmatic IT Project Management (PITPM)
 - 5.4 Configuration of an IT Project in PITPM
 - 5.5 Management of a project in PITPM
6. Schools of Thought in IT Project Management
 - 6.1 Agile Software Development
 - 6.2 Value-Based Software Engineering

Literature**Compulsory Reading****Further Reading**

- Berkun, S. (2009): Die Kunst des IT-Projektmanagements. 2. Auflage, O'Reilly, Sebastopol, CA.
- DeMarco, T. (2003): Bärenango. Mit Risikomanagement Projekte zum Erfolg führen. Carl Hanser Verlag, München.
- Geirhos, M. (2011): IT-Projektmanagement. Was wirklich funktioniert – und was nicht. Galileo Computing, Bonn.
- Höhn, R./Höppner S. (2008): Das V-Modell XT. Grundlagen, Methodik und Anwendungen. Springer, Berlin/Heidelberg.
- Malik, M. (2006): Führen, Leisten, Leben. Wirksames Management für eine neue Zeit. Campus, Frankfurt a. M.
- Mangold, P. (2009): IT-Projektmanagement kompakt. 3.Auflage, Spektrum.
- Motzel, E./Pannenbäcker, O. (1998): Projektmanagement-Kanon. Der deutsche Zugang zum Project Management Body of Knowledge. TÜV-Verlag, Köln.
- Patzak, G./Rattay, G.: Projektmanagement. Leitfaden zum Management von Projekten, Projektportfolios und projektorientierten Unternehmen. 5. Auflage, Linde Verlag, Wien.
- Phillips, J. (2010): IT Project Management. On Track from Start to Finish. 3. Auflage, McGraw-Hill, New York, NY.
- Pichler, R. (2007): Scrum. Agiles Projektmanagement erfolgreich einsetzen. dpunkt.verlag, Heidelberg.
- Schwalbe, K. (2010): Information Technology Project Management. 6. Auflage, Course Technology, Independence, KY.
- Tiemeyer, E. (2010): Handbuch IT-Projektmanagement. Vorgehensmodelle, Managementinstrumente, Good Practices. Hanser, München.
- Versteegen, G. (2000): Projektmanagement: mit dem Rational Unified Process. Springer, Berlin/Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

IT Architecture Management

Course Code: DLBCSEITPAM02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In addition to concrete IT projects, such as the development of a new IT system or the introduction of standard software, a strategic management system for organizational-wide IT infrastructure – that is, for all IT hardware and software systems – must be used. Strategic management is the responsibility of the IT enterprise architect, who operates IT architecture management. Their task is to strategically align IT infrastructure with an organization's business and IT strategy. This course covers the typical concepts, methods, procedures, and IT models of architecture management.

Course Outcomes

On successful completion, students will be able to

- describe and explain the basic principles of IT strategy, governance, and architecture management, differentiating between them.
- explain and differentiate the typical activities of IT architecture management, their interrelationships, and their dependencies.
- explain suitable models of IT architecture management, distinguish between them, and explain their intended purpose.
- explain and describe selected IT architectural frameworks as well as reference models and sample catalogues.

Contents

1. Basic Terms and Foundation for the Management of IT Enterprise Architectures
 - 1.1 IT Enterprise Architecture
 - 1.2 Goals of Enterprise Architecture Management
 - 1.3 Processes in the Management of IT Enterprise Architectures
2. IT Application Portfolio Management
 - 2.1 IT Application Portfolio Management Overview
 - 2.2 Application Manual
 - 2.3 Portfolio Analysis
 - 2.4 Development Planning

3. Architecture Governance
 - 3.1 Organizational Structure
 - 3.2 Policy Development and Enforcement
 - 3.3 Project Support
4. Modeling of IT Enterprise Architectures
 - 4.1 Models in the Context of IT Architecture Management
 - 4.2 Forms of Documentation for Processes and Applications
 - 4.3 Forms of Documentation for Systems and Technologies
5. Frameworks Using the Example of TOGAF
 - 5.1 Fundamentals and Use of IT Architecture Frameworks
 - 5.2 Overview and Categories of EAM Frameworks
 - 5.3 The Open Group Architecture Framework (TOGAF)
6. Reference Models and Sample Catalogues
 - 6.1 Architecture Reference Models
 - 6.2 EAM Design Sample Catalogue

Literature

Compulsory Reading

Further Reading

- Hanschke, I. (2011): Enterprise Architecture Management. Einfach und effektiv. Hanser, München.
- Keller, W. (2012): IT-Unternehmensarchitektur. Von der Geschäftsstrategie zur optimalen IT-Unterstützung. 2. Auflage, dpunkt.verlag, Heidelberg.
- Keuntje, J. H./Barkow, R. (Hrsg.) (2010): Enterprise Architecture. Management in der Praxis. Wandel, Komplexität und IT-Kosten im Unternehmen beherrschen. Symposion Publishing, Ettlingen.
- Ross, J. W./ Weill, P./Robertson, D. C. (2006): Enterprise Architecture as Strategy. Creating a Foundation for Business Execution. Harvard Business Review Press, Boston, MA.
- Schwarzer, B. (2009): Einführung in das Enterprise Architecture Management. Verstehen – Planen – Umsetzen. Books on Demand, Norderstedt.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

6. Semester

Salesforce Platform Development

Module Code: DLSFPD

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Maik Günther (Salesforce Platform App Builder) / Prof. Dr. Maik Günther (Salesforce Platform Developer)

Contributing Courses to Module

- Salesforce Platform App Builder (DLSFPD01)
- Salesforce Platform Developer (DLSFPD02)

Module Exam Type

Module Exam

Split Exam

Salesforce Platform App Builder

- Study Format "Distance Learning": Written Assessment: Project Report

Salesforce Platform Developer

- Study Format "Distance Learning": Oral Project Report

Weight of Module

see curriculum

Module Contents

Salesforce Platform App Builder

Using the learning platform Trailhead students will learn the fundamentals of Salesforce. At the end of the course, the students will be able to design, build and deploy custom applications. This course prepares them for the Salesforce Platform App Builder Certification.

Salesforce Platform Developer

Using the learning platform Trailhead students will learn how to develop own applications, built from various parts of the Salesforce platform. At the end of the course they will be able to use Apex, Visualforce and basic Lightning components. This course prepares the students for the Salesforce Platform Developer I Certification.

Learning Outcomes

Salesforce Platform App Builder

On successful completion, students will be able to

- define what Salesforce and customer relationship management is,
- design the data model, user interface, and business logic for custom applications,
- customize applications for mobile use,
- design reports and dashboards,
- manage application security and deploy custom applications.

Salesforce Platform Developer

On successful completion, students will be able to

- develop own applications using Apex and basic Lightning components,
- write SOSL, SOQL and DML statements,
- use Visualforce to build custom user interfaces for mobile and web apps,
- build reusable, performant components that follow modern web standards,
- use the built-in testing framework to test Apex and Visualforce.

Links to other Modules within the Study Program

This module is similar to other modules in the field of Marketing & Sales

Links to other Study Programs of IUBH

All Bachelor Programs in the Marketing & Communication field

Salesforce Platform App Builder

Course Code: DLSFPD01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Salesforce is the most used software solution for customer relationship management worldwide. This solution can be customized and personalized for the needs of customers, partners and employees. Using the learning platform Trailhead, students will learn independently the fundamentals of Salesforce and the development of customized application. This course prepares students for the Salesforce Platform App Builder Certification.

Course Outcomes

On successful completion, students will be able to

- define what Salesforce and customer relationship management is,
- design the data model, user interface, and business logic for custom applications,
- customize applications for mobile use,
- design reports and dashboards,
- manage application security and deploy custom applications.

Contents

- The content on the learning platform focuses on the features and functionality to design, build and deploy custom applications. The content also provides knowledge to define business logic and process automation declaratively. Furthermore, the design and management of the correct data models and the customization of applications for individual needs is included in this course. Thus, the content of this course enables to automate repetitive tasks and to optimize processes in customer organizations.

Literature**Compulsory Reading****Further Reading**

- Gupta, R. (2019): Salesforce Platform App Builder Certification. A Practical Study Guide. 1st ed., Apress.
- Weinmeister, P. (2019): Practical Salesforce Development Without Code. Building Declarative Solutions on the Salesforce Platform. 2nd ed., Apress, Berkeley.
- Shaalan, S. (2020): Salesforce for Beginners. A step-by-step guide to creating, managing, and automating sales and marketing processes. Packt Publishing, Birmingham.
- Benioff, M./Langley, M. (2019): Trailblazer. The Power of Business as the Greatest Platform for Change. 1st ed.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Salesforce Platform Developer

Course Code: DLSFPD02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The Salesforce platform not only forms the foundation of core Salesforce products like Sales Cloud and Service Cloud, but it is also possible to build own functionalities and own applications. Using the learning platform Trailhead, students will learn how to use the programmatic pillars of the Salesforce platform: Lightning components, Apex and Visualforce. This course prepares students for the Salesforce Platform Developer I Certification.

Course Outcomes

On successful completion, students will be able to

- develop own applications using Apex and basic Lightning components,
- write SOSL, SOQL and DML statements,
- use Visualforce to build custom user interfaces for mobile and web apps,
- build reusable, performant components that follow modern web standards,
- use the built-in testing framework to test Apex and Visualforce.

Contents

- The content on the learning platform focuses on the development of own functionality and own applications, built from various parts of the Salesforce platform. The content enables to use the programmatic elements Lightning components, Apex and Visualforce. Furthermore, knowledge is provided for data modeling, process automation, user interface design, testing and deployment. Thus, the content of this course enables to extend Salesforce by individual applications to cover the needs in customer organizations.

Literature

Compulsory Reading

Further Reading

- Salesforce (2020): Developer Documentation. (URL: <https://developer.salesforce.com/docs/> [accessed: 12.12.2020])

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Oral Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

DLSFPD02

Mobile Software Engineering

Module Code: DLBCSEMSE

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Mobile Software Engineering I) / N.N. (Mobile Software Engineering II)

Contributing Courses to Module

- Mobile Software Engineering I (DLBCSEMSE01)
- Mobile Software Engineering II (DLBCSEMSE02)

Module Exam Type

Module Exam

Split Exam

Mobile Software Engineering I

- Study Format "Distance Learning": Exam, 90 Minutes

Mobile Software Engineering II

- Study Format "Distance Learning": Written Assessment: Project Report

Weight of Module

see curriculum

Module Contents**Mobile Software Engineering I**

- Basics of mobile software development
- Android system architecture
- Development environment
- Core components of an Android app
- Interaction between application components
- Advanced techniques

Mobile Software Engineering II

Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task.

Learning Outcomes**Mobile Software Engineering I**

On successful completion, students will be able to

- recognize the differences and peculiarities of software development for mobile systems and explain them.
- differentiate between different activities, roles, and risks in the creation, operation, and maintenance of mobile software systems.
- explain and differentiate between the architecture and technical features of the Android platform.
- independently create mobile software systems to solve concrete problems for the “Android” platform.

Mobile Software Engineering II

On successful completion, students will be able to

- independently design and create a prototype of a small mobile application to solve a specific problem.
- recognize typical problems and challenges in the practical implementation of small mobile applications.
- document the conception and implementation of small, independently designed and implemented mobile applications.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programs in the IT & Technology fields

Mobile Software Engineering I

Course Code: DLBCSEMSE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using the mobile platform "Android" as an example, it will be demonstrated how the programming of mobile applications (apps) differs from the development of browser-based information systems, which technologies and programming concepts are typically used, and which typical challenges there are in app development for industrial applications.

Course Outcomes

On successful completion, students will be able to

- recognize the differences and peculiarities of software development for mobile systems and explain them.
- differentiate between different activities, roles, and risks in the creation, operation, and maintenance of mobile software systems.
- explain and differentiate between the architecture and technical features of the Android platform.
- independently create mobile software systems to solve concrete problems for the "Android" platform.

Contents

1. Basics of Mobile Software Development
 - 1.1 Special Features of Mobile Devices
 - 1.2 Special Features of Mobile Software Development
 - 1.3 Classification of Mobile Devices
 - 1.4 The Android Platform
2. Android System Architecture
 - 2.1 The Android System
 - 2.2 Safety and Security
 - 2.3 Communication with Networks
3. Development Environment
 - 3.1 Android Studio
 - 3.2 First App and Emulator Test
 - 3.3 Application Deployment

4. Core Components of an Android App
 - 4.1 Overview of the Components of an Android App
 - 4.2 Activities, Layouts, and Views
 - 4.3 Resources
 - 4.4 Summary in an App
 - 4.5 Graphic Design
5. Interaction Between Application Components
 - 5.1 Intents
 - 5.2 Services
 - 5.3 Broadcast Receiver
6. Advanced Techniques
 - 6.1 Threading
 - 6.2 Application Memory

Literature

Compulsory Reading

Further Reading

- Becker, A./Pant, M. (2015): Android 5. Programmieren für Smartphones und Tablets. 4. Auflage, dpunkt.verlag, Heidelberg.
- Eason, J. (2014): Android Studio 1.0. (URL: <https://android-developers.googleblog.com/2014/12/android-studio-10.html>)
- Eason, J. (2014): Android Studio 1.0. (URL: <https://android-developers.googleblog.com/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Franke, F./Ippen, J. (2012): Apps mit HTML5 und CSS3. Galileo Computing, Bonn.
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>) [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Mobile Software Engineering II

Course Code: DLBCSEMSE02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using the knowledge gained in the course "Mobile Software Engineering using the Android platform as an example", students independently create a mobile application and document its conception and implementation.

Course Outcomes

On successful completion, students will be able to

- independently design and create a prototype of a small mobile application to solve a specific problem.
- recognize typical problems and challenges in the practical implementation of small mobile applications.
- document the conception and implementation of small, independently designed and implemented mobile applications.

Contents

- Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task. Possible topics are, for example:
- A radio app to improve the exchange between listeners and stations in general, and listeners and radio presenters in particular.
- An app that allows a group of board game fans to better organize their regular evening game.
- An app that theses supervisors at IUBH can use to improve their supervision processes.

Literature**Compulsory Reading****Further Reading**

- Eason, J. (2014): Android Studio 1.0. (URL: <http://android-developers.blogspot.de/2014/12/android-studio-10.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015): Android Developer Guide. (URL: <http://developer.android.com/guide>)
- Google Inc. (Hrsg.) (2015a): App Components. (URL: <http://developer.android.com/guide/components/index.html> [letzter Zugriff: 12.06.2015]).
- Google Inc. (Hrsg.) (2015b): Installing the Android SDK. (URL: <http://developer.android.com/sdk/installing/index.html> [letzter Zugriff: 13.05.2015]).
- Google Inc. (Hrsg.) (2015c): Resources Overview. (URL: <http://developer.android.com/guide/topics/resources/overview.html> [letzter Zugriff: 12.06.2015]).
- Hipp, Wyrick & Company, Inc. (Hrsg.) (2015): SQLite Webseite. (URL: <http://sqlite.org/index.html> [letzter Zugriff: 12.06.2015]).

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Big Data and Cloud Technologies

Module Code: DLBCSEBDCT

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	- none - DLBCSDMDS01	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Christian Müller-Kett (Big Data Technologies) / Prof. Dr. Thomas Zöller (Cloud Computing)

Contributing Courses to Module

- Big Data Technologies (DLBDSBDT01)
- Cloud Computing (DLBDSCC01)

Module Exam Type

Module Exam

Split Exam

Big Data Technologies

- Study Format "Distance Learning": Exam, 90 Minutes

Cloud Computing

- Study Format "Distance Learning": Exam, 90 Minutes

Weight of Module

see curriculum

Module Contents**Big Data Technologies**

- Data types and data sources
- Text-based and binary data formats
- Distributed systems
- Streaming frameworks
- NoSQL approach to data storage

Cloud Computing

- Cloud computing fundamentals
- Relevant enabling technologies for cloud computing
- Introduction to serverless computing
- Established cloud platforms
- Cloud offerings for data science and analytics

Learning Outcomes**Big Data Technologies**

On successful completion, students will be able to

- name types and sources of data.
- understand text-based and binary data formats.
- analyze the requirements and constraints of distributed analysis systems.
- evaluate the applications of streaming frameworks.
- describe the motivation for NoSQL data stores and categorize pertaining established concepts.

Cloud Computing

On successful completion, students will be able to

- understand the fundamentals of cloud computing and cloud service models.
- recognize enabling technologies that underlie current cloud offerings.
- cite the principles of serverless computing.
- analyze characteristics of established cloud offerings.
- describe cloud options for data science and machine learning

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Big Data Technologies

Course Code: DLBDSBDT01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBCSDMDS01

Course Description

Data are often considered the “new oil”, the raw material from which value is created. To harness the power of data, the data need to be stored and processed on a technical level. This course introduces the four “Vs” of data, as well as typical data sources and types. The course discusses the most common data storage formats encountered in modern systems, focusing both on text-based as well as binary data formats. Handling large amounts of data poses significant challenges for the underlying infrastructure. The course discusses the most important distributed and streaming data handling frameworks which are used in leading edge applications.

Course Outcomes

On successful completion, students will be able to

- name types and sources of data.
- understand text-based and binary data formats.
- analyze the requirements and constraints of distributed analysis systems.
- evaluate the applications of streaming frameworks.
- describe the motivation for NoSQL data stores and categorize pertaining established concepts.

Contents

1. Data Types and Data Sources
 - 1.1 The 4Vs of data: volume, velocity, variety, veracity
 - 1.2 Data sources
 - 1.3 Data types
2. Text-Based and Binary Data Formats
 - 2.1 Simple formats: CSV, YAML
 - 2.2 XML
 - 2.3 JSON
 - 2.4 Hierarchical data format 5 (HDF 5)
 - 2.5 Apache Parquet
 - 2.6 Apache Arrow

3. NoSQL data stores
 - 3.1 Introduction and motivation
 - 3.2 Approaches and technical concepts
4. Distributed Systems
 - 4.1 Hadoop & MapReduce
 - 4.2 Hadoop file system (HDFS)
 - 4.3 Spark
 - 4.4 DASK
5. Streaming Frameworks
 - 5.1 Spark streaming
 - 5.2 Kafka

Literature

Compulsory Reading

Further Reading

- Karau, H., Konwinski, A., Wendell, A., & Zaharia, M. (2015). Learning spark: Lightning-fast data analysis. Sebastopol, CA: O'Reilly.
- Kleppmann, M. (2017). Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. Sebastopol, CA: O'Reilly.
- Narkhede, N., Shapira, G., & Palino, T. (2017). Kafka: The definitive guide: Real-time data and stream processing at scale. Sebastopol, CA: O'Reilly.
- Psaltis, A. (2017). Streaming data: Understanding the real-time pipeline. Shelter Island, NY: Manning.
- White, T. (2015). Hadoop: The definitive guide: Storage and analysis at Internet scale. Sebastopol, CA: O'Reilly.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Cloud Computing

Course Code: DLBDSCC01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Many of the recent advances in data science, particularly machine learning and artificial intelligence, rely on comprehensive data storage and computing power. Cloud computing is one way of providing that power in a scalable way, without considerable upfront investment in hardware and software resources. This course introduces the area of cloud computing together with its enabling technologies. Moreover, the most cutting-edge advances like serverless computing and storage are illustrated. Finally, a thorough overview on popular cloud offerings, especially in regard to analytics capabilities, is given.

Course Outcomes

On successful completion, students will be able to

- understand the fundamentals of cloud computing and cloud service models.
- recognize enabling technologies that underlie current cloud offerings.
- cite the principles of serverless computing.
- analyze characteristics of established cloud offerings.
- describe cloud options for data science and machine learning

Contents

1. Introduction to Cloud Computing
 - 1.1 Fundamentals of Cloud computing
 - 1.2 Cloud Service Models
 - 1.3 Benefits and Risks
2. Enabling Technology
 - 2.1 Virtualization and Containerization
 - 2.2 Storage Technology
 - 2.3 Networks and RESTful Services
3. Serverless Computing
 - 3.1 Introduction to Serverless Computing
 - 3.2 Benefits
 - 3.3 Limitations

4. Established Cloud Platforms
 - 4.1 Google Cloud Platform
 - 4.2 Amazon Web Services
 - 4.3 Microsoft Azure
5. Data Science in the Cloud
 - 5.1 Google Data Science and Machine Learning Services
 - 5.2 Amazon Web Services Data Science and Machine Learning Services
 - 5.3 Microsoft Azure Data Science and Machine Learning Services

Literature

Compulsory Reading

Further Reading

- Chapin, J., & Roberts, M. (2017). What is serverless? Sebastopol, CA: O'Reilly Media.
- Goessling, S., & Jackson, K. L. (2018). Architecting cloud computing solutions. Birmingham: Packt Publishing.
- Kavis, M. J. (2014). Architecting the cloud: Design decisions for cloud computing service models (SaaS, PaaS, and IaaS). Hoboken, NJ: Wiley.
- Mahmood, Z., Puttini, R., & Erl, T. (2013). Cloud computing: Concepts, technology & architecture. Boston, MA: Prentice Hall.
- Rafaels, R. (2018). Cloud computing (2nd ed.). Scotts Valley, CA: CreateSpace Independent Publishing Platform.
- Sehgal, N. K., & Bhatt, P. C. P. (2018). Cloud computing: Concepts and practices. Cham: Springer.
- Zonooz, P. Farr, E., Arora, K., & Laszewski, T. (2018). Cloud native architectures. Birmingham: Packt Publishing.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Business Intelligence

Module Code: DLBCSEBI

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Sebastian Werning (Business Intelligence) / Prof. Dr. Sebastian Werning (Project: Business Intelligence)

Contributing Courses to Module

- Business Intelligence (DLBCSEBI01)
- Project: Business Intelligence (DLBCSEBI02)

Module Exam Type

Module Exam

Split Exam

Business Intelligence

- Study Format "Distance Learning": Exam, 90 Minutes

Project: Business Intelligence

- Study Format "Distance Learning": Written Assessment: Project Report

Weight of Module

see curriculum

Module Contents**Business Intelligence**

- Basics of mobile software development
- Android system architecture
- Development environment
- Core components of an Android app
- Interaction between application components
- Advanced techniques

Project: Business Intelligence

Conception, implementation, and documentation of small, mobile applications on the basis of a concrete task.

Learning Outcomes**Business Intelligence**

On successful completion, students will be able to

- explain the motivation, use cases, and basics of Business Intelligence.
- identify and explain techniques and methods for providing and modeling data, as well as types of data relevant to BI, differentiating between them.
- explain techniques and methods for the generation and storage of information and independently select suitable methods on the basis of concrete requirements.

Project: Business Intelligence

On successful completion, students will be able to

- independently design a solution to a practical problem in the field of Business Intelligence in order to then implement a prototype and document the results.
- identify and explain typical problems and challenges in the design and practical implementation of small BI solutions.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology fields

Business Intelligence

Course Code: DLBCSEBI01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Business Intelligence (BI) is used to obtain information from company data that is relevant for targeted corporate management and the optimization of business activities. This course introduces and discusses techniques, procedures, and models for data provision, information generation, and analysis, as well the distribution of the information obtained. You will then be able to explain the various subject areas of data warehousing and independently select methods and techniques to meet specific requirements.

Course Outcomes

On successful completion, students will be able to

- explain the motivation, use cases, and basics of Business Intelligence.
- identify and explain techniques and methods for providing and modeling data, as well as types of data relevant to BI, differentiating between them.
- explain techniques and methods for the generation and storage of information and independently select suitable methods on the basis of concrete requirements.

Contents

1. Motivation and Conceptualization
 - 1.1 Motivation and Historical Development
 - 1.2 BI as a Framework
2. Data Provision
 - 2.1 Operative and Dispositive Systems
 - 2.2 The Data Warehouse Concept
 - 2.3 Architectural Variations
3. Data Warehouse
 - 3.1 ETL Process
 - 3.2 DWH and Data Mart
 - 3.3 ODS and Metadata

4. Modelling of Multidimensional Data Spaces

- 4.1 Data Modeling
- 4.2 OLAP Cubes
- 4.3 Physical Storage
- 4.4 Star and Snowflake Scheme
- 4.5 Historicization

5. Analysis Systems

- 5.1 Free Data Research and OLAP
- 5.2 Reporting Systems
- 5.3 Model-Based Analysis Systems
- 5.4 Concept-Oriented Systems

6. Distribution and Access

- 6.1 Information Distribution
- 6.2 Information Access

Literature

Compulsory Reading

Further Reading

- Bachmann, R./Kemper, G. (2011): Raus aus der BI-Falle. Wie Business Intelligence zum Erfolg wird. 2. Auflage, mitp, Heidelberg.
- Bauer, A./Günzel, H. (2008): Data Warehouse Systeme. Architektur, Entwicklung, Anwendung. 3. Auflage, dpunkt.verlag, Heidelberg.
- Betz, R. (2015): Werde Jäger des verlorenen Schatzes. In: Immobilienwirtschaft, Heft 5, S. 1614–1164. (URL <https://www.haufe.de/download/immobilienwirtschaft-ausgabe-052015-immobilienwirtschaft-fachmagazin-fuer-management-recht-praxis-303530.pdf> [letzter Zugriff: 27.02.2017]).
- Bodendorf, F. (2006): Daten- und Wissensmanagement. 2. Auflage, Springer, Berlin.
- Chamoni, P./Gluchowski, P. (Hrsg.) (2006): Analytische Informationssysteme Business Intelligence-Technologien und -Anwendungen. Springer, Berlin.
- Engels, C. (2008): Basiswissen Business Intelligence. W3L, Herdecke/Witten.
- Gansor, T./Totok, A./Stock, S. (2010): Von der Strategie zum Business Intelligence Competency Center (BICC). Konzeption – Betrieb – Praxis. Hanser, München.
- Gluchowski, P./Gabriel, R./Dittmar, C. (2008): Management Support Systeme und Business Intelligence. Computergestützte Informationssysteme für Fach- und Führungskräfte. 2. Auflage, Springer, Berlin/Heidelberg.
- Grothe, M. (2000): Business Intelligence. Aus Informationen Wettbewerbsvorteile gewinnen. Addison-Wesley, München.
- Gutenberg, E. (1983): Grundlagen der Betriebswirtschaft, Band 1. Die Produktion. 18. Auflage, Springer, Berlin/Heidelberg/New York.
- Hannig, U. (Hrsg.) (2002): Knowledge Management und Business Intelligence. Springer, Berlin.
- Hansen, H.-R./Neumann, G. (2001): Wirtschaftsinformatik I. Grundlagen betrieblicher Informationsverarbeitung. 8. Auflage, Lucius & Lucius UTB, Stuttgart.
- Humm, B./Wietek, F. (2005): Architektur von Data Warehouses und Business Intelligence Systemen. In: Informatik Spektrum, S. 3–14. (URL: https://www.fbi.h-da.de/fileadmin/personal/b.humm/Publikationen/Humm__Wietek_-_Architektur_DW__Informatik-Spektrum_2005-01_.pdf [letzter Zugriff: 27.02.2017]).
- Kemper, H.-G./Baars, H./Mehanna, W. (2010): Business Intelligence – Grundlagen und praktische Anwendungen. Eine Einführung in die IT-basierte Managementunterstützung. 3. Auflage, Vieweg+Teubner, Stuttgart.
- Turban, E. et al. (2010): Business Intelligence. A Managerial Approach. 2. Auflage, Prentice Hall, Upper Saddle River (NJ).

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Project: Business Intelligence

Course Code: DLBCSEBI02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Using well-known methods and techniques from the field of Business Intelligence, students will work independently on a practical question in this course. At the end of the course you will be able to independently design and prototype Business Intelligence applications based on concrete requirements.

Course Outcomes

On successful completion, students will be able to

- independently design a solution to a practical problem in the field of Business Intelligence in order to then implement a prototype and document the results.
- identify and explain typical problems and challenges in the design and practical implementation of small BI solutions.

Contents

- Implementation and documentation of practical questions regarding the use of Business Intelligence applications. Typical scenarios are, for example, "Management of BI projects", "Design of multidimensional data models" and "Prototypical implementation of small BI applications".

Literature

Compulsory Reading

Further Reading

- Brenner, W./Uebersnickel, F. (2015): Design Thinking. Das Handbuch. Frankfurter Allgemeine Buch, Frankfurt a. M.
- Brown, T. (2008): Design Thinking. In: Harvard Business Review, Heft Juni, S. 84–95.
- Meinel, C./Weinberg, U./Krohn, T. (Hrsg.) (2015): Design Thinking Live. Wie man Ideen entwickelt und Probleme löst. Murmann, Hamburg.
- Uebersnickel, F./Brenner, W. (2016): Design Thinking. In: Hoffmann, C. P. et al. (Hrsg.): Business Innovation: Das St. Galler Modell. Springer, Wiesbaden, S. 243–265.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

Software Engineering with Python

Module Code: DLBCSESEWP

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	- DLBDSIPWP01 or DLBDSIPWP01_D; DLBDSOOFPP01 or IOBP01 - none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Prof. Dr. Max Pumperla (Object oriented and functional programming in Python) / Prof. Dr. Max Pumperla (Data Science Software Engineering)

Contributing Courses to Module

- Object oriented and functional programming in Python (DLBDSOOFPP01)
- Data Science Software Engineering (DLBDSSE01)

Module Exam Type

Module Exam

Split Exam

Object oriented and functional programming in Python

- Study Format "Distance Learning": Portfolio (50)

Data Science Software Engineering

- Study Format "Distance Learning": Exam, 90 Minutes (100)

Weight of Module

see curriculum

Module Contents**Object oriented and functional programming in Python**

- This course introduces the students to the advanced programming concepts of object orientation and functional programming and how they are realized in the Python programming language.

Data Science Software Engineering

- Traditional project management
- Agile project management
- Testing
- Software development paradigms
- From model to production

Learning Outcomes**Object oriented and functional programming in Python**

On successful completion, students will be able to

- explain basic notions in object-oriented programming such as functions and classes.
- understand object-oriented programming concepts and their relation to software design and engineering.
- describe advanced function concepts in Python.
- recognize important ideas from functional programming.
- recall important libraries for functional programming in Python.

Data Science Software Engineering

On successful completion, students will be able to

- understand the concept of project management approaches.
- apply agile approaches in software development.
- create automated software tests.
- understand various software development paradigms.
- evaluate the necessary steps to bring models into a production environment.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Data Science & Artificial Intelligence.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Object oriented and functional programming in Python

Course Code: DLBDSOOFPP01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

This course builds upon basic knowledge of Python programming (Introduction to Programming with Python, DLBDSIPWP) and is concerned with the exposition of advanced Python programming concepts. To this end, important notions of object-oriented programming like classes and objects and pertaining design principles are outlined. Starting from an in-depth discussion of advanced features of Python functions, functional programming concepts and their implementation in Python are conveyed.

Course Outcomes

On successful completion, students will be able to

- explain basic notions in object-oriented programming such as functions and classes.
- understand object-oriented programming concepts and their relation to software design and engineering.
- describe advanced function concepts in Python.
- recognize important ideas from functional programming.
- recall important libraries for functional programming in Python.

Contents

- This course provides students with a thorough introduction to important notions and concepts from the domain of object-oriented programming such as classes, objects, abstraction, encapsulation, inheritance, polymorphism, composition, and delegation. Additionally, the functional programming paradigm and pertaining ideas like functions as first class objects, decorators, pure functions, immutability and higher order functions are conveyed. Pursuant to the portfolio course type, the aforementioned concepts and ideas are explored by hands-on programming projects.

Literature**Compulsory Reading****Further Reading**

- Lott, S. F. (2018): Functional Python programming: Discover the power of functional programming, generator functions, lazy evaluation, the built-in itertools library, and monads. 2nd ed., Packt Publishing, Birmingham.
- Lutz, M. (2013): Learning Python. 5th ed., O'Reilly, Sebastopol, CA.
- Phillips, D. (2018): Python 3 object-oriented programming: Build robust and maintainable software with object-oriented design patterns in Python 3.8. 3rd ed., Packt Publishing, Birmingham.
- Ramalho, L. (2015): Fluent Python: Clear, concise, and effective programming. O'Reilly, Sebastopol, CA.

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Portfolio

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☒ Live Tutorium/Course Feed

Data Science Software Engineering

Course Code: DLBDSSE01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	DLBDSIPWP01 or DLBDSIPWP01_D; DLBDSOOFPP01 or IOBP01

Course Description

A core part of data science is creating value from data. This means not only the creation of sophisticated predictive models but also the development of these models according to modern software development principles. This course gives a detailed overview of the relevant methods and paradigms which data scientists need to know in order to develop enterprise-grade models. This course discusses traditional and agile project management techniques, highlighting both the Kanban and Scrum approaches. It explores relevant software development paradigms such as test-driven development, pair programming, mob programming, and extreme programming. Special focus is given to the topic of testing and the consideration of how to bring a model into a production environment.

Course Outcomes

On successful completion, students will be able to

- understand the concept of project management approaches.
- apply agile approaches in software development.
- create automated software tests.
- understand various software development paradigms.
- evaluate the necessary steps to bring models into a production environment.

Contents

1. Traditional Project Management
 - 1.1 Requirements engineering
 - 1.2 Waterfall model
 - 1.3 Rational unified process
2. Agile Project Management
 - 2.1 Criticism of the waterfall model
 - 2.2 Introduction to SCRUM
 - 2.3 Introduction to Kanban

3. Testing
 - 3.1 Why testing?
 - 3.2 Unit tests
 - 3.3 Integration tests
 - 3.4 Performance monitoring
4. Software Development Paradigms
 - 4.1 Test-driven development (TDD)
 - 4.2 Pair programming
 - 4.3 Mob programming
 - 4.4 Extreme programming
5. From Model to Production
 - 5.1 Continuous delivery
 - 5.2 Continuous integration
 - 5.3 Building a scalable environment

Literature

Compulsory Reading

Further Reading

- Farcic, V. (2016): The DevOps 2.0 toolkit: Automating the continuous deployment pipeline with containerized microservices. CreateSpace Independent Publishing Platform, Scotts Valley, CA.
- Humble, J./Farley, D. (2010): Continuous delivery: Reliable software releases through build, test, and deployment automation. Addison-Wesley Professional, Boston, MA.
- Humble, J./Molesky, J./O'Reilly, B. (2015): Lean enterprise. O'Reilly Publishing, Sebastopol, CA.
- Hunt, A./Thomas, D. (1999): The pragmatic programmer. From journeyman to master. Addison-Wesley, Reading, MA.
- Martin, R. C. (2008): Clean code. Prentice Hall, Boston, MA.
- Morris, K. (2016): Infrastructure as code. O'Reilly Publishing, Sebastopol, CA.
- Richardson, L./Ruby, S. (2007): RESTful web services. O'Reilly Publishing, Sebastopol, CA.
- Senge, P. (1990): The fifth discipline: The art and practice of the learning organization. Broadway Business, New York, NY.

Study Format Distance Learning

Study Format Distance Learning	Course Type Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

IT project and architecture management

Module Code: DLBCSEITPAM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (IT Project Management) / N.N. (IT Architecture Management)

Contributing Courses to Module

- IT Project Management (DLBCSEITPAM01)
- IT Architecture Management (DLBCSEITPAM02)

Module Exam Type

Module Exam

Split Exam

IT Project Management

- Study Format "Distance Learning": Exam, 90 Minutes

IT Architecture Management

- Study Format "Distance Learning": Exam, 90 Minutes

Weight of Module

see curriculum

Module Contents**IT Project Management**

- Basic terms and foundations of IT project management
- Large and small planning techniques
- Techniques for prioritization, cost-estimation, and project controlling
- Techniques for stakeholder, communication, and risk management
- Organization and structure in IT project management
- Schools of thought in IT project management

IT Architecture Management

- Basic terms and foundations of IT enterprise architectures management
- IT application portfolio management
- Architecture governance
- Modeling of IT enterprise architectures
- Frameworks using TOGAF as an example
- Reference models and sample catalogues

Learning Outcomes**IT Project Management**

On successful completion, students will be able to

- explain and differentiate between the basic principles and tasks of IT project management.
- explain the important practical techniques and methods necessary for the implementation of IT project management.
- describe the basic procedural models and explain their advantages and disadvantages as well as their possible applications.
- identify possible project risks on the basis of given practical scenarios and select suitable measures from IT project management in order to minimize them in a targeted manner.

IT Architecture Management

On successful completion, students will be able to

- describe and explain the basic principles of IT strategy, governance, and architecture management, differentiating between them.
- explain and differentiate the typical activities of IT architecture management, their interrelationships, and their dependencies.
- explain suitable models of IT architecture management, distinguish between them, and explain their intended purpose.
- explain and describe selected IT architectural frameworks as well as reference models and sample catalogues.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

IT Project Management

Course Code: DLBCSEITPAM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In this course, typical problems in the management of Software projects are discussed and the methods and techniques used to address challenges conveyed. In addition, standard procedural models for IT project management are explained and their strengths and weaknesses specifically identified.

Course Outcomes

On successful completion, students will be able to

- explain and differentiate between the basic principles and tasks of IT project management.
- explain the important practical techniques and methods necessary for the implementation of IT project management.
- describe the basic procedural models and explain their advantages and disadvantages as well as their possible applications.
- identify possible project risks on the basis of given practical scenarios and select suitable measures from IT project management in order to minimize them in a targeted manner.

Contents

1. Basics Terms and Foundations of IT Project Management
 - 1.1 Definition of a Project and Types of IT Projects
 - 1.2 IT Project Lifecycle
 - 1.3 Multi-Project Management – The Project in the Context of the Organization
2. Planning Techniques
 - 2.1 Large-Scale Planning: Milestones, Sub-tasks, and Work Packages
 - 2.2 Large-Scale Planning: Gantt Charts
 - 2.3 Planning and Organization of Work Packages: Kanban Board
3. Prioritization, Estimation of Costs, Project Controlling
 - 3.1 Prioritization
 - 3.2 Estimation of Costs
 - 3.3 Project Controlling

4. Stakeholder, Communication and Risk Management
 - 4.1 Stakeholder Management
 - 4.2 Communication Management
 - 4.3 Risk Management
5. Organization and Structure in IT Project Management
 - 5.1 Overview and Levels of Management from PRINCE2
 - 5.2 Management Processes in PRINCE2
 - 5.3 Pragmatic IT Project Management (PITPM)
 - 5.4 Configuration of an IT Project in PITPM
 - 5.5 Management of a project in PITPM
6. Schools of Thought in IT Project Management
 - 6.1 Agile Software Development
 - 6.2 Value-Based Software Engineering

Literature**Compulsory Reading****Further Reading**

- Berkun, S. (2009): Die Kunst des IT-Projektmanagements. 2. Auflage, O'Reilly, Sebastopol, CA.
- DeMarco, T. (2003): Bärenango. Mit Risikomanagement Projekte zum Erfolg führen. Carl Hanser Verlag, München.
- Geirhos, M. (2011): IT-Projektmanagement. Was wirklich funktioniert – und was nicht. Galileo Computing, Bonn.
- Höhn, R./Höppner S. (2008): Das V-Modell XT. Grundlagen, Methodik und Anwendungen. Springer, Berlin/Heidelberg.
- Malik, M. (2006): Führen, Leisten, Leben. Wirksames Management für eine neue Zeit. Campus, Frankfurt a. M.
- Mangold, P. (2009): IT-Projektmanagement kompakt. 3.Auflage, Spektrum.
- Motzel, E./Pannenbäcker, O. (1998): Projektmanagement-Kanon. Der deutsche Zugang zum Project Management Body of Knowledge. TÜV-Verlag, Köln.
- Patzak, G./Rattay, G.: Projektmanagement. Leitfaden zum Management von Projekten, Projektportfolios und projektorientierten Unternehmen. 5. Auflage, Linde Verlag, Wien.
- Phillips, J. (2010): IT Project Management. On Track from Start to Finish. 3. Auflage, McGraw-Hill, New York, NY.
- Pichler, R. (2007): Scrum. Agiles Projektmanagement erfolgreich einsetzen. dpunkt.verlag, Heidelberg.
- Schwalbe, K. (2010): Information Technology Project Management. 6. Auflage, Course Technology, Independence, KY.
- Tiemeyer, E. (2010): Handbuch IT-Projektmanagement. Vorgehensmodelle, Managementinstrumente, Good Practices. Hanser, München.
- Versteegen, G. (2000): Projektmanagement: mit dem Rational Unified Process. Springer, Berlin/Heidelberg.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

IT Architecture Management

Course Code: DLBCSEITPAM02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

In addition to concrete IT projects, such as the development of a new IT system or the introduction of standard software, a strategic management system for organizational-wide IT infrastructure – that is, for all IT hardware and software systems – must be used. Strategic management is the responsibility of the IT enterprise architect, who operates IT architecture management. Their task is to strategically align IT infrastructure with an organization's business and IT strategy. This course covers the typical concepts, methods, procedures, and IT models of architecture management.

Course Outcomes

On successful completion, students will be able to

- describe and explain the basic principles of IT strategy, governance, and architecture management, differentiating between them.
- explain and differentiate the typical activities of IT architecture management, their interrelationships, and their dependencies.
- explain suitable models of IT architecture management, distinguish between them, and explain their intended purpose.
- explain and describe selected IT architectural frameworks as well as reference models and sample catalogues.

Contents

1. Basic Terms and Foundation for the Management of IT Enterprise Architectures
 - 1.1 IT Enterprise Architecture
 - 1.2 Goals of Enterprise Architecture Management
 - 1.3 Processes in the Management of IT Enterprise Architectures
2. IT Application Portfolio Management
 - 2.1 IT Application Portfolio Management Overview
 - 2.2 Application Manual
 - 2.3 Portfolio Analysis
 - 2.4 Development Planning

3. Architecture Governance
 - 3.1 Organizational Structure
 - 3.2 Policy Development and Enforcement
 - 3.3 Project Support
4. Modeling of IT Enterprise Architectures
 - 4.1 Models in the Context of IT Architecture Management
 - 4.2 Forms of Documentation for Processes and Applications
 - 4.3 Forms of Documentation for Systems and Technologies
5. Frameworks Using the Example of TOGAF
 - 5.1 Fundamentals and Use of IT Architecture Frameworks
 - 5.2 Overview and Categories of EAM Frameworks
 - 5.3 The Open Group Architecture Framework (TOGAF)
6. Reference Models and Sample Catalogues
 - 6.1 Architecture Reference Models
 - 6.2 EAM Design Sample Catalogue

Literature

Compulsory Reading

Further Reading

- Hanschke, I. (2011): Enterprise Architecture Management. Einfach und effektiv. Hanser, München.
- Keller, W. (2012): IT-Unternehmensarchitektur. Von der Geschäftsstrategie zur optimalen IT-Unterstützung. 2. Auflage, dpunkt.verlag, Heidelberg.
- Keuntje, J. H./Barkow, R. (Hrsg.) (2010): Enterprise Architecture. Management in der Praxis. Wandel, Komplexität und IT-Kosten im Unternehmen beherrschen. Symposion Publishing, Ettlingen.
- Ross, J. W./ Weill, P./Robertson, D. C. (2006): Enterprise Architecture as Strategy. Creating a Foundation for Business Execution. Harvard Business Review Press, Boston, MA.
- Schwarzer, B. (2009): Einführung in das Enterprise Architecture Management. Verstehen – Planen – Umsetzen. Books on Demand, Norderstedt.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Exam, 90 Minutes

Student Workload					
Self Study 90 h	Presence 0 h	Tutorial 30 h	Self Test 30 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☒ Live Tutorium/Course Feed

Agile Project Management

Module Code: DLBCSAPM

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Dr. Mark Menzel (Agile Project Management)

Contributing Courses to Module

- Agile Project Management (DLBCSAPM01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Project Report

Split Exam

Weight of Module

see curriculum

Module Contents

- In this course, students are taught action competences in the field of agile project management. They will be familiarized with the values, activities, roles, and artifacts of agile procedures using Scrum as an example.

Learning Outcomes**Agile Project Management**

On successful completion, students will be able to

- explain the differences between agile and plan-driven project management.
- explain agile principles.
- work together in an agile manner according to the values defined in Scrum.
- apply the activities defined in Scrum.
- take responsibility for the roles defined in Scrum.
- create and maintain the artefacts defined in Scrum.

Links to other Modules within the Study Program

This module is similar to other modules in the fields of Computer Science & Software Development

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology fields

Agile Project Management

Course Code: DLBCSAPM01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Students will receive a practical introduction to agile project management in this course. In addition to teaching its individual basic principles, the differences between agile project management and plan-driven project management will be examined in detail. In order to understand and experience agile project management, the values, activities, roles, and artefacts of typical agile procedures are presented using Scrum and then practiced on an example project.

Course Outcomes

On successful completion, students will be able to

- explain the differences between agile and plan-driven project management.
- explain agile principles.
- work together in an agile manner according to the values defined in Scrum.
- apply the activities defined in Scrum.
- take responsibility for the roles defined in Scrum.
- create and maintain the artefacts defined in Scrum.

Contents

- This course teaches students various skills in the field of agile project management. In contrast to plan-driven project management, the principles of agility used in modern software development are taught. Using the example of Scrum, students will acquire skills in applying an agile approach, and then apply their knowledge of respective roles and activities in a simple project to gain initial practical experience, documenting it in a project report. The content of the projects results from the individual abilities and requirements of the students.

Literature**Compulsory Reading****Further Reading**

- Röpstorff, S./Wiechmann, R. (2012): Scrum in der Praxis. Erfahrungen, Problemfelder und Erfolgsfaktoren. dpunkt.verlag, Heidelberg.
- Rubin, K. S. (2014): Essential Scrum. Umfassendes Scrum-Wissen aus der Praxis. Mitp Verlag, Frechen.
- Roock, A. (2011): Software-Kanban. Eine Einführung. In: Projektmagazin, Heft 4,
- Leffingwell, D. et al. (o. J.): Scaled Agile Framework. (URL: <http://scaledagileframework.com/> [letzter Zugriff: 17.07.2015]).
- Schwaber, K./Sutherland, J. (o. J.): The Scrum Guide - The definitive Guide to Scrum: The Rules of the Game. www.scrumguides.org

Study Format Distance Learning

Study Format Distance Learning	Course Type Project

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: no
Type of Exam	Written Assessment: Project Report

Student Workload					
Self Study 120 h	Presence 0 h	Tutorial 30 h	Self Test 0 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

DLBCSAPM01

IT Law

Module Code: DLBCSIITL

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (International IT Law)

Contributing Courses to Module

- International IT Law (DLBCSIITL01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Case Study, 90 Minutes

Split Exam

Weight of Module

see curriculum

Module Contents

- Basic Concepts of Legal Systems
- Internet and Domain Law
- Contracts
- Intellectual Property
- Data Protection / Privacy

Learning Outcomes**International IT Law**

On successful completion, students will be able to

- describe basic concepts of IT law.
- provide examples of different approaches to IT law in different countries.
- identify legal questions as they arise in IT.
- apply the core ideas of data protection and privacy in their work.
- distinguish the different types of contracts and intellectual property as they relate to IT.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

International IT Law

Course Code: DLBCSIITL01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

The application of IT is embedded in a legal framework which computer scientists need to know and adhere to in their work. This applies to the way their own work is performed which, for example, may be governed by contracts with suppliers and/or customers. Computer scientists create and use intellectual property, and this leads to questions of copyright, software patents, etc. Beyond this, IT strongly influences the social environment and therefore needs to abide by regulations such as data protection. The goal of this module is to provide students with a basic understanding of these legal aspects so they can take them into account, apply them in simple cases, and recognize when more specialised legal knowledge is required. Since IT is a topic that connects different countries and legal frameworks, the course looks at some of the common legal questions as they are handled in the European Union, the USA, and India.

Course Outcomes

On successful completion, students will be able to

- describe basic concepts of IT law.
- provide examples of different approaches to IT law in different countries.
- identify legal questions as they arise in IT.
- apply the core ideas of data protection and privacy in their work.
- distinguish the different types of contracts and intellectual property as they relate to IT.

Contents

1. Basic Concepts of Legal Systems
 - 1.1 The Role of Law in IT
 - 1.2 Basic Concepts of the Legal System in the European Union
 - 1.3 Basic Concepts of the Legal System in the USA
 - 1.4 Basic Concepts of the Legal System in India
2. Internet and Domain Law
 - 2.1 Web Sites and the Law
 - 2.2 Net Neutrality
 - 2.3 Domain Registration
 - 2.4 Internet Crime

3. Contracts
 - 3.1 Types of IT Contracts
 - 3.2 Electronic Contracts and Electronic Signatures
 - 3.3 Licences
 - 3.4 Free and Open Source Software
 - 3.5 Buying and Selling Off-the-Shelf Software
 - 3.6 Software Development Contracts
4. Intellectual Property
 - 4.1 Brands, Trade Marks and Domain Names
 - 4.2 Copyright
 - 4.3 Software Patents
 - 4.4 Digital and Data Ownership
5. Data Protection/Privacy
 - 5.1 Basic Concepts of Data Protection
 - 5.2 Data Protection in the European Union: the GDPR
 - 5.3 Data Protection in the USA
 - 5.4 Data Protection in India
 - 5.5 Trans-Border Data Flows

Literature

Compulsory Reading

Further Reading

- Hoeren, T., & Pinelli, S. (2018). Agile programming – Introduction and current legal challenges. *Computer Law & Security Review*, 34(5), pp. 1131-1138. Retrieved from www.uni-muenster.de/Jura.itm/hoeren/itm/wp-content/uploads/Hr.-Hoeren-29.10.pdf
- Lloyd, I. (2018). *Information technology law* (8th ed.). Oxford: Oxford University Press.
- Murray, A. (2019). *Information technology law: The law and society* (4th ed.). Oxford: Oxford University Press.
- Soma, J. T. (2014). *Privacy law in a nutshell*. St. Paul, MN: West Academic.
- Wikia.org. (n.d.). The IT law wiki [web encyclopedia]. Retrieved from https://itlaw.wikia.org/wiki/The_IT_Law_Wiki#

Study Format Distance Learning

Study Format Distance Learning	Course Type Case Study

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: no
Type of Exam	Written Assessment: Case Study, 90 Minutes

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☒ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☒ Guideline ☐ Live Tutorium/Course Feed

DLBCSIITL01

Computer Science and Society

Module Code: DLBCSCSAS

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	5	150 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

N.N. (Computer Science and Society)

Contributing Courses to Module

- Computer Science and Society (DLBCSCSAS01)

Module Exam Type

Module Exam

Study Format: Distance Learning
Written Assessment: Written Assignment

Split Exam

Weight of Module

see curriculum

Module Contents

- The role of computer science
- The impact of the information society on the economy and society
- Infrastructure vulnerability
- Informatics and the military
- Responsibility in Information Technology

Learning Outcomes**Computer Science and Society**

On successful completion, students will be able to

- name significant stages of development in computer science and the Internet.
- describe the role of computer science as a science and its relation to other sciences.
- explain and discuss the main economic and social implications of the information society.
- explain and discuss the causes and effects of infrastructure vulnerability.
- discuss and apply relevant ethical principles to issues in computer science.

Links to other Modules within the Study Program

This module is similar to other modules in the field(s) of Computer Science & Software Development.

Links to other Study Programs of IUBH

All Bachelor Programmes in the IT & Technology field(s).

Computer Science and Society

Course Code: DLBCSCSAS01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		5	none

Course Description

Computer science is a science that shapes society in a special way, and thus has a significant impact on society. Depending on its application and prevailing conditions, it can help to improve society and living conditions or cause and exacerbate surveillance, unemployment, and other social problems. In many cases, it is not the results of computer science itself that are "good" or "bad", but the results can be used to serve very different purposes ("dual use"). In this module students acquire the basic tools to understand the effects of computer science in general and its effects on their everyday work as a data scientist.

Course Outcomes

On successful completion, students will be able to

- name significant stages of development in computer science and the Internet.
- describe the role of computer science as a science and its relation to other sciences.
- explain and discuss the main economic and social implications of the information society.
- explain and discuss the causes and effects of infrastructure vulnerability.
- discuss and apply relevant ethical principles to issues in computer science.

Contents

1. Introduction
 - 1.1 Computer Science, Society, and the Information Society
 - 1.2 Historical Overview of Computer Science and the Internet
 - 1.3 What is Computer Science?
 - 1.4 Relevant Organizations
2. The Role of Data in the Information Society
 - 2.1 Data as a Commodity
 - 2.2 Data Protection and Data Spying
 - 2.3 Long-Term Archiving

3. Economic Impacts of the Information Society
 - 3.1 Globalization and the Formation of Monopolies in the Economy
 - 3.2 Open Movement
 - 3.3 Change in the Labor Market
 - 3.4 Intellectual Property
4. Social impacts of the Information Society
 - 4.1 Social Networks
 - 4.2 Surveillance
 - 4.3 Digitization of Education
 - 4.4 Women in Computer Science
5. Infrastructure Vulnerability
 - 5.1 Attacks and Accidents
 - 5.2 Technical Infrastructure
 - 5.3 Political and Social Infrastructure
6. Computer Science and the Military
 - 6.1 Military as the Driver of Computer Science
 - 6.2 Cyber War
 - 6.3 Dual Use
7. Responsibility in Computer Science
 - 7.1 Ethics of Responsibility According to Jonas
 - 7.2 Ethical Guidelines for Computer Science in Society
 - 7.3 Decision-making by Algorithms
 - 7.4 Mechanisms for Implementing Responsibility in Computer Science

Literature**Compulsory Reading****Further Reading**

- Bittner, P. et al. (Hrsg.) (2014): Gesellschaftliche Verantwortung in der digital vernetzten Welt. Reihe: Kritische Informatik. Band 8, LIT-Verlag, Berlin/Münster/ Wien/Zürich/London.
- Clegg, B. (2017): Big Data. How the Information Revolution is Transforming Our Lives. Icon Books, London.
- Drösner, C. (2016): Total berechenbar? Wenn Algorithmen für uns entscheiden. Hanser, München.
- Eibl, M./Loebel, J.-M./Reiterer, H. (2015): Grand Challenge „Erhalt des digitalen Kulturerbes“. In: Informatik-Spektrum, Band 38, Heft 4, S. 269–276.
- Weber-Wulff, D. et al. (2009): Gewissensbisse. Ethische Probleme der Informatik. Biometrie – Datenschutz – geistiges Eigentum. Transcript Verlag, Bielefeld.

Study Format Distance Learning

Study Format Distance Learning	Course Type Online Lecture

Information about the examination	
Examination Admission Requirements	BOLK: yes Course Evaluation: yes
Type of Exam	Written Assessment: Written Assignment

Student Workload					
Self Study 110 h	Presence 0 h	Tutorial 20 h	Self Test 20 h	Practical Experience 0 h	Hours Total 150 h

Instructional Methods	
☐ Learning Sprints® ☒ Course Book ☐ Vodcast ☒ Shortcast ☐ Audio ☒ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Bachelor Thesis

Module Code: DLBBT

Module Type	Admission Requirements	Study Level	CP	Student Workload
see curriculum	none	BA	10	300 h

Semester / Term	Duration	Regularly offered in	Language of Instruction
see curriculum	Minimum 1 semester	WiSe/SoSe	English

Module Coordinator

Degree Program Advisor (SGL) (Bachelor Thesis) / Degree Program Advisor (SGL) (Colloquium)

Contributing Courses to Module

- Bachelor Thesis (DLBBT01)
- Colloquium (DLBBT02)

Module Exam Type

Module Exam

Split Exam

Bachelor Thesis

- Study Format "Distance Learning": Written Assessment: Bachelor Thesis

Colloquium

- Study Format "Distance Learning": Presentation: Colloquium

Weight of Module

see curriculum

Module Contents**Bachelor Thesis**

- Bachelor's thesis
- Colloquium on the bachelor's thesis

Colloquium**Learning Outcomes****Bachelor Thesis**

On successful completion, students will be able to

- work on a problem from their major field of study by applying the specialist and methodological skills they have acquired during their studies.
- independently analyze selected tasks with scientific methods, critically evaluate them, and develop appropriate solutions under the guidance of an academic supervisor.
- record and analyze existing (research) literature appropriate to the topic of their bachelor's thesis.
- prepare a detailed written elaboration in compliance with scientific methods.

Colloquium

On successful completion, students will be able to

- present a problem from their field of study using academic presentation and communication techniques.
- reflect on the scientific and methodological approach chosen in their bachelor's thesis.
- demonstrate that they can actively answer subject-related questions from the subject experts (reviewers of the bachelor's thesis).

Links to other Modules within the Study Program

All modules in the bachelor program

Links to other Study Programs of IUBH

All bachelor programs in distance learning

Bachelor Thesis

Course Code: DLBBT01

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		9	none

Course Description

The aim and purpose of the bachelor's thesis is to successfully apply the subject-specific and methodological competencies acquired during the course of study in the form of an academic dissertation with a thematic reference to the major field of study. The content of the bachelor's thesis can be a practical-empirical or theoretical-scientific problem. Students should prove that they can independently analyze a selected problem with scientific methods, critically evaluate it, and work out proposed solutions under the subject-methodological guidance of an academic supervisor. The topic chosen by the student from their respective field of study should meet the acquired scientific competences, deepening their academic knowledge and skills in order to meet the future needs of the field.

Course Outcomes

On successful completion, students will be able to

- work on a problem from their major field of study by applying the specialist and methodological skills they have acquired during their studies.
- independently analyze selected tasks with scientific methods, critically evaluate them, and develop appropriate solutions under the guidance of an academic supervisor.
- record and analyze existing (research) literature appropriate to the topic of their bachelor's thesis.
- prepare a detailed written elaboration in compliance with scientific methods.

Contents

- The bachelor's thesis must be written on a topic that relates to the content of the respective major field of study. In the context of the bachelor's thesis, the problem, as well as the scientific research goal, must be clearly emphasized. The work must reflect the current state of knowledge of the topic to be examined by means of an appropriate literature analysis. The student must prove their ability to use the acquired knowledge theoretically and/or empirically in the form of an independent and problem-solution-oriented application.

Literature**Compulsory Reading****Further Reading**

- Hunziker, A.W. (2010): Spaß am wissenschaftlichen Arbeiten. So schreiben Sie eine gute Semester-, Bachelor- oder Masterarbeit. 4. Auflage, Verlag SKV, Zürich.
- Wehrlin, U. (2010): Wissenschaftliches Arbeiten und Schreiben. Leitfaden zur Erstellung von Bachelorarbeit, Masterarbeit und Dissertation – von der Recherche bis zur Buchveröffentlichung. AVM, München.
- Selection of literature according to topic

Study Format Distance Learning

Study Format Distance Learning	Course Type Thesis

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: yes
Type of Exam	Written Assessment: Bachelor Thesis

Student Workload					
Self Study 270 h	Presence 0 h	Tutorial 0 h	Self Test 0 h	Practical Experience 0 h	Hours Total 270 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☒ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed

Colloquium

Course Code: DLBBT02

Study Level	Language of Instruction	Contact Hours	CP	Admission Requirements
BA	English		1	none

Course Description

The colloquium will take place after the submission of the bachelor's thesis. This is done at the invitation of the experts. During the colloquium, students must prove that they have independently produced the content and results of the written work. The content of the colloquium is a presentation of the most important work contents and research results by the student as well as the answering of questions by experts.

Course Outcomes

On successful completion, students will be able to

- present a problem from their field of study using academic presentation and communication techniques.
- reflect on the scientific and methodological approach chosen in their bachelor's thesis.
- demonstrate that they can actively answer subject-related questions from the subject experts (reviewers of the bachelor's thesis).

Contents

- The colloquium includes a presentation of the most important results of the bachelor's thesis, followed by the student answering the reviewers' technical questions.

Literature

Compulsory Reading

Further Reading

- Renz, K.-C. (2016): Das 1 x 1 der Präsentation. Für Schule, Studium und Beruf. 2. Auflage, Springer Gabler, Wiesbaden.

Study Format Distance Learning

Study Format Distance Learning	Course Type Thesis Defense

Information about the examination	
Examination Admission Requirements	BOLK: no Course Evaluation: yes
Type of Exam	Presentation: Colloquium

Student Workload					
Self Study 30 h	Presence 0 h	Tutorial 0 h	Self Test 0 h	Practical Experience 0 h	Hours Total 30 h

Instructional Methods	
☐ Learning Sprints® ☐ Course Book ☐ Vodcast ☐ Shortcast ☐ Audio ☐ Exam Template	☐ Review Book ☐ Creative Lab ☐ Guideline ☐ Live Tutorium/Course Feed