Modulhandbuch

High Integrity Systems

Master of Science (M.Sc.)
Fb 2: Informatik und Ingenieurwissenschaften

Fachhochschule Frankfurt am Main
- University of Applied Sciences
Nibelungenplatz 1
60318 Frankfurt am Main

Modulbeschreibung zum Modul 1

Module title	Safety Critical Computer Systems
Module number	1
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Usable in the M.Sc. program Basys – Intelligente Systeme
Duration	1
Status	Mandatory module
Recommended semester	1st /2nd semester Offered each semester for 1st semester students
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Bachelor courses : Programming Languages Software Engineering Mathematics
Requirements for module examination	none
Examination type	Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester
Education goals/ capabilities	• Upon completion of this course, the student is able to: • distinguish between reliability and safety, • critically read accident reports, • perform a hazard analysis on a computer-based system, • write requirements for a safety-critical system and trace safety constraints to design, • work with human factors experts in the design of safe human-computer interaction, • apply the principles of safe design to both systems and software,

	• criticize and evaluate a system design for safety, and design a process for building a safety-critical system, • distinguish between the role of practitioners and managers. Non specialist compentencies (25% of total workload): • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Scientific literature research and handling • Time and project management skills
Module contents	Safety Critical Computer Systems – Lectures Safety Critical Computer Systems - Exercises
Unit teaching modes	Lectures: Interactive teaching Exercises: Teamwork in small development groups
Total workload (h)	150
Language	English
Module Frequency	Winter term
Module coordination	Prof. Dr. Matthias Wagner

Unitbeschreibung zum Modul 1:

Unit name	Safety Critical Computer Systems - Lectures
Code	
Module name	Safety Critical Computer Systems
Lecturers	Prof. Dr. Wagner
Contents	• Introduction into principles of system safety • Safety Critical Computer Systems (SCS) • Terminology • Safety criteria • Hazards analysis • Risk analysis • Risk classification scheme • Safety integrity levels (SIL) • Ethical considerations, risk tolerance levels • Development of safety critical systems • System and Software Engineering Best Practices • SCS requirements analysis • SCS design goals • Fault tolerance • System reliability
Teaching mode	Interactive lectures
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	Nancy Leveson: Safeware Addison Wesley 1995 Nancy Leveson: Engineering a Safer World MIT Press 2012 Neil Storey: Safety Critical Computer Systems Prentice Hall 1996 Hollnagel et al.: Resilience Engineering Ashgate 2010 Current literature will be announced at the beginning of each

	semester.
Module examination	Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 1:

Unit name	Safety Critical Computer Systems - Exercises
Code	
Module name	Safety Critical Computer Systems
Lecturers	Prof. Dr. Wagner
Contents	• Lab exercises with software tools pertaining to the contents described in the unit Safety Critical Computer Systems – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small development groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	Nancy Leveson: Safeware Addison Wesley 1995 Nancy Leveson: Engineering a Safer World MIT Press 2012 Neil Storey: Safety Critical Computer Systems Prentice Hall 1996 Hollnagel et al.: Resilience Engineering Ashgate 2010 Current literature will be announced at the beginning of each semester.
Module examination	None
Module examination assessment	None

Modulbeschreibung zum Modul M2 Advanced Formal Modeling

Module title	Advanced Formal Modeling (M. Sc.)
Module number	2
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	2 weekly hrs: Lectures Advanced Formal Modeling 2 weekly hrs: Exercises Advanced Formal Modeling
Level	M.Sc.
Applicability	Usable in other Computer Science Master programs
Duration	1 Semester
Status	Mandatory module
Recommended semester	1st /2nd semester
Credits	5
General recommended prerequisites	None
Recommended prerequisites	• Basic knowledge of propositional and predicate logic • Basic knowledge of software specification techniques
Requirements for module examination	Successful participation in the unit exercises
Examination type	Written examination of 90 minutes duration
Education goals/ capabilities	• Understanding the mathematical background and theoretical foundations of formal methods in the software engineering processes, which are relevant for safety critical systems. • Assessing the need for zero-defect software in safety critical systems. • Ability to distinguish formal specification methods. • Ability to carry out correctness proofs for simple code fragments. • Studying advanced formal methods. • Understanding the limitation of advanced formal methods. Non specialist compentencies (15% of total workload): • Scientific working style
Module units	Lectures Advanced Formal Modeling Exercises Advanced Formal Modeling
Unit teaching modes	Lectures and Exercises
Total workload(h)	150 h

Language	English
Module frequency	Every Winter term
Module coordination	Prof. Dr. Ruth Schorr
Hints	

Unitbeschreibung zum Modul 2:

Unit name	Lectures Advanced Formal Modeling
Code	
Module name	Advanced Formal Modeling
Lecturers	Prof. Dr. Ruth Schorr, Prof. Dr. Matthias Schubert

Contents	• Basics
	  o Set theory
	  o Relations
	• Propositional logic
	• First order predicate logic
	• Verification techniques for sequential programs like Hoare logic
	• Set-theoretic specification methods
	  o Z-Notation
	  o B-Method
	• Algebraic approach to formal modelling methods with abstract data types

Teaching mode	Lectures
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. Exam preparation	10
Practical part	0
Self study	30
Language	English
Literature	• Monin, Jean-Francois: Understanding Formal Methods, Springer, 2003. • Schneider, Steve: The B-method, Palgrave, 2001. • Spivey, J.M.: The Z Notation: A reference manual, Prentice Hall, 2001. Current literature will be announced at the beginning of each semester.
Module examination	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme
Hints	

Unitbeschreibung zum Modul 2:

Unit name	Exercises Advanced Formal Modeling
Code	
Modul name	Advanced Formal Modeling
Lecturers	Prof. Dr. Ruth Schorr, Prof. Dr. Matthias Schubert
Contents	Exercises and examples to ensure that the students learn to solve problems using the methods from the lecture. To support the learning process continuous feedback is provided.
Teaching mode	Exercises
Weekly hours	2
Total workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	See Unit Lectures Advanced Formal Modeling
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 3 Introductory Data Analysis

Module title	Introductory Data Analysis
Module number	3
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	
Duration	1 semester
Status	Mandatory module
Recommended semester	1st/2nd semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Basic skills in statistics as they are offered in the Bachelor programme Informatik, i.e students should be able to perform the most important methods of inferential statistics in line with some real-world problems, the students should be able to interpret and assess the results of basic statistical methods
Requirements for module examination	50% Regular attendance at exercise groups, (unit Introductory Data Analysis – Exercises) solutions to 40% of weekly exercises in unit Introductory Data Analysis - Exercises short written exposé as stated in unit Introductory Data Analysis - Exercises
Examination type	Written (computer) examination of 90 minutes duration
Education goals/ capabilities	• Confident assessment of the usage of the various methods of univariate and bivariate statistics in the application context. • Knowledge and understanding of different probability concepts (distributions, statistical models, testing procedures and principles) • Capacity to apply methods to selected real world situations • Capacity to use the computer to solve problems in real world situations • Capacity to under stand and judge results of statistical analysis

	• Awareness of dangers of misuse and misinterpretation • Capacity to communicate using statistical language, i.e. explain procedures, results of an analysis and a critique of the results Non specialist compentencies (15% of total workload): • Scientific work style
Module units	Introductory Data Analysis - lectures Introductory Data Analysis - exercises
Unit teaching modes	Lectures using multimedia presentation techniques Exercises on PC using spreadsheets and statistical software tool
Total workload (h)	150
Language	English
Module frequency	Every Winter term
Module coordination	Prof. Dr. Andreas Orth

Unitbeschreibung zum Modul Introductory Data Analysis 3:

Unit name	Introductory Data Analysis - Lectures
Code	
Module name	Introductory Data Analysis
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Descriptive statistics (characterics and plots, univariate and bivariate methods) • Probabilty concepts and theory (Baysian and frequentist approach, formulating of statistical models) • Inferential statistics (concepts and a selection of tests) • Some Test theory (assumptions, hypotheses, OC, alpha/ beta error) • Performing Statistical Tests (Checking Assumptions. preparing the data, understanding results, discussing results) • Performing Statistical Tests (Examples of applications) • non-paramtric tools • Common Errors (how not to proceed)
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley. • • Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them)
Module examination	Prerequisite: Successful participation in Unit: Introductory Data Analysis - Exercises Written (computer) examination of 90 minutes duration

Unitbeschreibung zum Modul 3 Introductory Data Analysis

Unit name	Introductory Data Analysis - Exercises
Code	
Module name	Introductory Data Analysis
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Computer Exercises pertaining to the contents described in the unit Introductory Data Analysis – lectures • short written exposé of one real world problem, inlcuding reasoning on why which methods were selected, including interpretation and critique of results obtained • lesson's learned session after written exposé • exam preparation session for the Module examination
Teaching mode	Using PC in Computer pool to solve problems
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley. • • Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them)
Module examination	50% Regular attendance at exercise groups, (unit Data Mining Methods – Exercises) solutions to 40% of weekly exercises in unit Data Mining Methods - Exercises Compilation of a short written exposé
Module examination assessment	Graded according to published grading schemeNot gradedNot graded

Modulbeschreibung zum Modul 4

Module title	Advanced Real-Time Systems
Modul number	4
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	4 SWS, Group Project
Level	M.Sc.
Applicability	Applicable in other computer science master curricula especially master program BaSys – Intelligente Systeme
Duration	1
Status	Mandatory
Recommended semester	1st /2nd semester
Credits	5
General prerequisites	None
Prerequisites	Recommended: Knowledge in operating systems, programming languages, automata theory, formal languages, hardware architectures, modeling, and simulation
Requirements for module examination	The project should be worked out in a team of students (no more than four) with a 2-weekly written report of each participant describing essential aspects of the process from the point of view of each participant.
Module examination	Delivery of a written paper describing the theoretical concept of a real-time project in combination with a working software demonstrating the theory of the paper. The format of the paper has to be in accordance with a paper template of a typical scientific conference
Education goals/ capabilities	• Extending the basic knowledge of real-time systems by reading a typical real-time research-paper • Transfering the knowledge into a theoretical model solving a concrete problem • Transfering the theoretical model into a working software • Validating the software Non specialist compentencies (25% of total workload):

	• Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Scientific literature research and handling • Writing a paper • Time and project management skills • Project documentation
Contents	The main target is the the understanding of the concept of real-time systems (RTS) while using the knowledge within a project to solve a problem theoretically and by software. This includes topics like: - Specification of RTS requirements and expected system behavior with regard to a given problem - Modeling of a RT system function with regard to main scheduling strategies - Implementation of the RTS system - Validation of the implemented system - Theoretical models as basis for the software implementation
Teaching mode	After an introduction the student teams will work in a project setting. They have to use official textbooks and/ or scientific papers to back up their knowledge. The professor can be interviewed on demand.
Total workload (h)	150
Language	English
Module frequency	Winter term
Module coordination	Prof. Dr. Gerd Doeben-Henisch and Prof. Dr. Matthias Deegener
Hints	

Unitbeschreibung:

Unit name	Advanced Real-Time Systems - Project
Code	
Module name	Advanced Real-Time Systems
Lecturers	Prof. Dr. Matthias Deegener, Prof. Dr. Gerd Döben-Henisch
Contents	The main target is the the understanding of the concept of real-time systems (RTS) while using the knowledge within a project to solve a problem theoretically and by software. This includes topics like: • Specification of RTS requirements and expected system behavior with regard to a given problem • Modeling of a RT system function with regard to the main scheduling strategies • Implementation of the RTS system • Validation of the implemented system • Theoretical models as basis for the software implementation
Teaching mode	R&D project with small groups (4 students max.)
Weekly hours	4
Total workload (h)	150
Attendance	60
Exam incl. exam preparation	10
Practical part	60
Self study	80
Language	English
Literature	Current literature, e.g. research papers, will be announced at the begin of the semester
Module examination	See module description
Module examination assessment	Graded according to published grading scheme
Hints	

Modulbeschreibung zum Modul 5 Implementation of DBMS

Module title	Implementation of DBMS
Module number	5
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Applicable in other computer science master curricula
Duration	1
Status	Mandatory module
Recommended semester	1st/2nd semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Good knowledge of the fundamentals of Database Systems (data modeling, DDL, DML), knowledge of computer science algorithms and data structures as well as programming skills in higher programming languages. This corresponds to the following bachelor modules: • Databases • Algorithms and Data Structures • Programming
Requirements for module examination	none
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	Upon completion of this course, the student is able to: • Understand why databases form the backbone of every modern information system, and why a robust database management system (DBMS) is crucial for these systems. • Decide which architectures and implementation issues are relevant for robust DBMS. • Comprehend prerequisites for building and extending a DBMS as well as for building the DBMS part of a larger application in a robust fashion.

	• Assess the role of available parameters of commercial DBMS and thus, be able to tune these parameters in a way that results in a robust and best performing system. Non specialist compentencies (15% of total workload): • Working in teams • Communication in international teams
Module units	• Implementation of DBMS - lectures • Implementation of DBMS - exercises
Unit teaching modes	Interactive lectures Teamwork in lab exercises
Total workload (h)	150
Language	English
Module frequency	Annual
Module coordination	Prof. Dr. Christian Rich
Hints	

Unitbeschreibung zum Modul 5

Unit name	Implementation of DBMS - lectures
Code	
Module name	Implementation of DBMS
Lecturers	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Contents	• DBMS architectures • DBMS memory management • Buffer management • Indexing • Query processing and optimization • Implementation techniques for database operators • Backup and Recovery • Tuning and self-tuning of DBMS
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• Elmasri, R. and S. Navathe. *Fundamentals of Database Systems*, Addison Wesley Publishing Company • Garcia-Molina, H., J. D. Ullman and J. D. Widom, *Database Systems: The Complete Book*, Pearson/Prentice Hall. • • Kifer, M., A. Bernstein and P.M. Lewis, *Database Systems: An Application-Oriented Approach*. Addison Wesley / Pearson. • Ramakrishnan, R. and J. Gehrke, *Database Management Systems*, McGraw-Hill, hardcover as well as eBook • • Sasha, D., P. Bonnet, *Database Tuning: Principles, Experiments, and Troubleshooting Techniques* (The Morgan Kaufmann Series in Data Management Systems) • Silberschatz, A., H. Korth (Autor), A. Silberschatz, *Database Systems Concepts*. McGraw-Hill.
Module examination	Prerequisite: Successful participation in Unit: Implementation of DBMS - Exercises

	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 5

Unit name	Implementation of DBMS - Exercises
Code	
Module name	Implementation of DBMS
Lecturers	Prof. Dr. Justsus Klingemann, Prof. Dr. Christian Rich
Contents	• Exercises tasks based on module contents
Teaching mode	Exercises, team work
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• See unit Implementation of DBMS - lectures
Module examination	none
Module examination assessment	

Modulbeschreibung zum Modul 6

Module title	Pattern Oriented Software Architecture
Module number	6
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Applicable in other computer science master curricula
Duration	1
Status	Mandatory module
Recommended semester	$1^{st}/2^{nd}$ semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Good knowledge in principles and procedures of software engineering, programming skills in object-oriented programming languages
Requirements for module examination	none
Examination type	Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester
Education goals/ capabilities	• Upon completion of this course, the student is able to: • understand the motives of the pattern community. • distinguish between different types of patterns. • apply patterns in the design of SCS. • assess new developments of pattern catalogs and languages. Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	Pattern Oriented Software Architecture - lectures Pattern Oriented Software Architecture - exercises
Unit teaching modes	Interactive lectures Teamwork in lab exercises
Total workload (h)	150

Language	English
Module frequency	Annual
Module coordination	Prof. Dr. Matthias Schubert

Unitbeschreibung zum Modul 6

Unit name	Pattern Oriented Software Architecture - lectures
Code	
Module name	Pattern Oriented Software Architecture
Lecturers	Prof. Dr. Matthias Schubert, Prof. Dr. Jörg Schäfer
Contents	• Software architecture • Origins of the pattern movement • Principles and Practices of Modern Software Development and the prominent Role of Patterns • Pattern-oriented software architecture: Architectural patterns, Design patterns, • Idioms • Application-specific pattern systems • Patterns for software testing • Pattern languages • Alternatives, e.g. Frameworks
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• E. Gamma et. Al: Design Patterns Addison-Wesley, 1998 • Buschmann et al: Pattern Oriented Software Architecture. Addison Wesley 1996
Module examination	Prerequisite: Successful participation in Unit: Pattern Oriented Software Architecture - Exercises Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester

Unitbeschreibung zum Modul 6

Unit name	Pattern Oriented Software Architecture - Exercises
Code	
Module name	Pattern Oriented Software Architecture
Lecturers	Prof. Dr. Matthias Schubert, Prof. Dr. Jörg Schäfer
Contents	• Lab exercises pertaining to the contents described in the unit Pattern Oriented Software Architecture – lectures • lesson's learned session after solved problems
Teaching mode	Teamwork in small R&D groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	See unit Pattern Oriented Software Architecture
Module examination	none
Module examination assessment	

Modulbeschreibung zum Modul 7 Mathematics Update

Module title	Mathematics Update
Module number	7
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	In other Computer Science master curricula
Duration	1
Status	Mandatory module
Recommended semester	2 Offered for 2nd semester students
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Undergraduate level in mathematics
Requirements for module examination	none
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	• Upon completion of this module the student is able to • analyze mathematical problems in a software project's list of requirements • to familiarize with new mathematical fields • assess the suitability and usability of mathematical software tools Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	Mathematics Update - lectures Mathematics Update - exercises
Unit teaching modes	Interactive lectures Exercises with teamwork in small groups

Total workload (h)	150
Language	English
Module frequency	Summer semester for students starting in Winter, Winter semester for students starting in Summer
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 7

Unit name	Mathematics Update - Lectures
Code	
Module name	Mathematics Update
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Schubert, Prof. Dr. Matthias Wagner, Prof. Dr. Ruth Schorr
Contents	• Linear Algebra • Geometry • Discrete Mathematics • Calculus • Scientific Computing •
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• H. Anton, Calculus, A new horizon, Sixth Edition, John Wiley and Sons, New York, 1999; • H. Anton, Elementary Linear Algebra, John Wiley and Sons, New York, 1994; • J. Stewart, Calculus, Cengage Learning Emea; Auflage: 7th Revised edition, 2011; • Scilab/Matlab on-line literature • Press et al.: Numerical Recipes. Cambridge University Press, 2007
Module examination	Prerequisite: Successful participation at Unit: Mathematics Update - Exercises Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 7

Unit name	Mathematics Update - Exercises
Code	
Module name	Mathematics Update
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Schubert, Prof. Dr. Matthias Wagner, Prof. Dr. Ruth Schorr
Contents	• Computer Exercises pertaining to the contents described in the unit Mathematics Update – Lectures • lesson's learned session
Teaching mode	Teamwork in small R&D groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	See unit Mathematics Update Lectures
Module examination	none
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 8.1

Module title	Advanced Distributed Systems
Module number	8.1
Study program	M.Sc. Program High-Integrity Systems
Module code	
Units	Lectures Exercises
Level	Master
Applicability	Usable in other Computer Science Master programs
Duration	1
Status	Elective Subject
Recommended semester	1st/2nd semester
Credits	5
General recommended prerequisites	None

Recommended prerequisites	• Knowledge of networking principles, basic knowledge of distributed applications, as well as programming skills in object-oriented programming languages.
	• This corresponds to the following modules of the Bachelor program Computer Science (Informatik):
	  • Rechnernetze
	  • Verteilte Anwendungen
	  • Objektorientierte Programmierung
	  • OOP Vertiefung

Requirements for module examination	None
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	• Understanding the advantages and problems of distributed systems. • Knowledge of different distributed architectures and algorithms. • Ability to analyze distributed systems, in particular with respect to robustness.

	Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	• Advanced Distributed Systems - lectures • Advanced Distributed Systems - exercises
Unit teaching modes	Lectures: Interactive group lecturing Exercises: Teamwork in small groups
Total workload (h)	150
Language	English
Module frequency	Annual, Summer term
Module coordination	Prof. Dr. Justus Klingemann

Unitbeschreibung zum Modul 8.1

Unit name	Advanced Distributed Systems Lectures
Code	
Module name	Advanced Distributed Systems
Lecturers	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Contents	• Properties of distributed systems • Time and synchronization • Distributed algorithms • Middleware for distributed systems • Consistency and replication
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	G. Coulouris, J. Dollimore, T. Kindberg: Distributed Systems: Concepts and Design, Addison-Wesley A. Tanenbaum, M. van Steen: Distributed Systems: Principles and Paradigms, Prentice-Hall
Module examination	Written examination of 90 minutes duration at the end of the semester
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 8.1

Unit name	Advanced Distributed Systems Exercises
Code	
Module name	Advanced Distributed Systems
Lecturers	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Contents	• Lab exercises with software tools pertaining to the contents described in the unit Distributed Systems – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small software development groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	See unit Advanced Distributed Systems lectures
Module examination	none
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 8.2

Module title	Advanced Testing Methods
Module number	8.2
Study program	M.Sc. Program High-Integrity Systems
Module code	
Level	M.Sc.
Applicability	Usable in other Computer Science Master programs
Duration	1
Status	Elective Subject
Recommended semester	1st/2nd semester
Credits	5
General recommended prerequisites	None
Recommended prerequisites	Extended knowledge in software engineering, very good programming skills in procedural and object-oriented programming languages
Requirements for module examination	None
Examination type	Written examination of 90 minutes duration
Education goals/ capabilities	Upon completion of this course, the student is able to: • assess different testing methodologies, • master various powerful testing procedures, • differentiate between the testing of procedural and object oriented software, • estimate the importance of safety criteria for test case design, • recognize the limits of testing capabilities, • use gained experience to select valuable automated tests, • recognize tests not to be automated. Non specialist competencies (15% of the total workload): • This module facilitates communication structures used in business like Wikis and Discussion boards to show challenges working in global teams.
Module units	• Unit Advanced Testing Methods - lectures • Unit Advanced Testing Methods - exercises

Unit teaching modes	Lectures: Interactive group lecturing Exercises: Teamwork in small groups
Total workload (h)	150
Language	English
Module frequency	Annual, Summer term
Module coordination	Dr. Torsten Schönfelder, Deutsche Lufthansa

Unitbeschreibung zum Modul 8.2

Unit name	Lectures Advanced Testing Methods
Code	
Module name	Advanced Testing Methods
Lecturers	Dr. Torsten Schönfelder, Deutsche Lufthansa
Contents	• Planning for verification and validation • Design for testability • Testing strategies • Testing procedures • Testing of object-oriented systems • Testing patterns • Testing of and with safety criteria • Environment simulation • Testing tools
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	Current Software Engineering literature announced at the beginning of the semester
Module examination	Written examination with 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 8.2

Unit name	Advanced Testing Methods Exercises
Code	
Module name	Advanced Testing Methods
Lecturers	Dr. Torsten Schönfelder, Deutsche Lufthansa
Contents	• Lab exercises with software tools pertaining to the contents described in the unit Advanced Testing Methods – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small software development groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	Current Software Engineering literature announced at the beginning of the semester
Module examination	none
Module examination assessment	

Modulbeschreibung zum Modul 9

Module title	Advanced IT-Security
Module number	9
Study program	M.Sc. Program High-Integrity Systems
Module code	
Level	M.Sc.
Applicability	Usable in other Computer Science Master programs
Duration	1
Status	Mandatory
Recommended semester	1st/2nd semester
Credits	5
General recommended prerequisites	None
Recommended prerequisites	• Good knowledge in introductory computer science, • Programming skills in C or Java, • Basic System Administration Skills in Windows and Unix • Theoretical foundations of computer science, networks, operating systems
Requirements for module examination	None
Examination type	Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester
Education goals/ capabilities	Upon completion of this course, the student is able to: • to identify, analyze, and perhaps solve network-related security problems in computer systems. • to understand security problems in the combination of the Internet with Intranets. • to comprehend the need to protect all architectural levels. • to get an understanding of how to coordinate hardware and software to provide data security against internal and external attacks. Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	• Advanced IT-Security lectures

	• Advanced IT-Security exercises
Unit teaching modes	Lectures: Interactive group lecturing Exercises: Teamwork in small groups
Total workload (h)	150
Language	English
Module frequency	Annual Summer term
Module coordination	Prof. Dr. Martin Kappes

Unitbeschreibung zum Modul 9

Unit name	Advanced IT-Security - Lectures
Code	
Module name	Advanced IT-Security
Lecturers	Prof. Dr. Martin Kappes
Contents	• Introduction • Cryptography, Computational Complexity and Computability • Security threats in computer networks and countermeasures and security protocols on all layers of the reference model • Firewalls, VPNs • Anomaly Detection • Further Topics
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	Current literature to be announced at the beginning of the semester
Module examination	Oral examination of at least 15 and maximum 45 minutes duration at the end of the semester
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 9

Unit name	Advanced IT-Security - Exercises
Code	
Module name	Advanced IT-Security
Lecturers	Prof. Dr. Martin Kappes
Contents	• Lab exercises with software tools pertaining to the contents described in the unit Advanced IT-Security – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small software development groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	See unit Advanced IT-Security lectures
Module examination	none
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 10.1

Module title	Human-Machine Interaction
Modul number	10.1
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	4 SWS, Group Project
Level	Master
Applicability	Applicable in other computer science master curricula especially master program BaSys – Intelligente Systeme
Duration	1
Status	Elective module
Recommended semester	$1^{st}/2^{nd}$ semester
Credits	5
General prerequisites	none
Prerequisites	Recommended: knowledge in software and systems engineering, knowledge in programming
Requirements for module examination	The project should be worked out in a team of students (no more than four) with a 2-weekly written report of each participant describing essential aspects of the process from the point of view of each participant.
Module examination	Delivery of a written paper describing the theoretical concept of a HMI-project in combination with a working demonstrator showing the working of the theory. The format of the paper has to be in accordance with a paper template of a typical scientific conference. The project should include at least one realized empirical usability test with the demonstrator.
Education goals/ capabilities	• Overall goal is to gain basic knowledge about HMI as part of a systems engineering process, psychological conditions of a user, how to describe the behavior of user, how to derive requirements for an interface, and how to test the usability of an interface • Transfering the gained knowledge into a theoretical model solving a concrete problem • Transfering the theoretical model into a working demonstrator • Validating the demonstrator with the aid of usability tests

	Non specialist compentencies (25% of total workload): • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper
Contents	•HMI as part of a systems engineering process •Psychological aspects of human - computer interaction •Behavior modeling and interface design •HMI prototyping •Usability tests
Teaching mode	After an introduction the student teams work in a project. They have to use official textbooks and/ or scientific papers to back up their knowledge. The professor can be interviewed on demand.
Total workload (h)	150
Language	English
Module frequency	Annual, Summer term
Module coordination	Prof. Dr. Gerd Doeben-Henisch
Hints	

Unitbeschreibung

Unit name	Human Machine Interacttion - Project
Code	
Module name	Human Machine Interaction
Lecturers	Prof. Dr. Matthias Deegener, Prof. Dr. Gerd Döben-Henisch
Contents	•HMI as part of a systems engineering process •Psychological aspects of human - computer interaction •Behavior modeling and interface design •HMI prototyping •Usability tests
Teaching mode	R&D project with small groups (4 students max.)
Weekly hours	4
Total workload (h)	150
Attendance	60
Exam incl. exam preparation	10
Practical part	60
Self study	80
Language	English
Literature	Current literature, e.g. research papers, will be announced at the begin of the semester
Module examination	See module description
Module examination assessment	Graded according to published grading scheme
Hints	

Modulbeschreibung zum Modul 10.2

Module title	Smart Sensor Network Systems
Module number	10.2
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	4 weekly hrs: Group Project
Level	M.Sc.
Applicability	Applicable in other computer science master curricula especially master program BaSys – Intelligente Systeme
Duration	1
Status	Elective module
Recommended semester	1st/2nd semester
Credits	5
Conditions for module participation	none
Recommended prerequisites	Recommended: • Knowledge in software and systems engineering, • C/C++ - programming • Numerical analysis
Conditions for examination	The project should be worked out in a team of students (no more than four) with a 2-weekly written report of each participant describing essential aspects of the process from the point of view of each participant.
Module examination	Oral examination of at least 15 minutes and maximum 30 minutes duration, based on a written report and an oral presentation of project results.
Education goals/ capabilities	• Upon completion of this course, the student is able to: • understand the interface between computer science and the physical environment, • assess the challenges of the measuring process and the possible errors, • set up and program a Wireless Sensor Network and interface it with a standard network and/or the Internet,

	• participate in the solution of measuring tasks by cooperation with specialists of other disciplines. Non specialist compentencies (25% of total workload): • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper
Module Contents	• Introduction to measuring technology for computer scientists • Data acquisition basics • The measuring chain • Data acquisition challenges and error propagation • Intelligent sensor concepts • Wireless sensor networks (WSN) • WSN operating systems • Real-time aspects of WSNs • Signal analysis basics
Unit teaching modes	Project
Total workload (h)	150
Language	English
Module frequency	Annual, Summer term
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung

Unit name	Smart Sensor Network Systems - Project
Code	
Module name	Smart Sensor Systems
Lecturers	Prof. Dr. Matthias Wagner
Contents	• Introduction to measuring technology for computer scientists • Data acquisition basics • The measuring chain • Data acquisition challenges and error propagation • Intelligent sensor concepts • Wireless sensor networks (WSN) • WSN operating systems • Real-time aspects of WSNs • Signal analysis basics
Teaching mode	R&D project with small groups (4 students max.)
Weekly hours	4
Total workload (h)	150
Attendance	60
Exam incl. exam preparation	10
Practical part	60
Self study	80
Language	English
Literature	HolgerKarl, Andreas Willig: Protocols and Architectures for Wireless Sensor Networks, Wiley, 2005 Current literature, e.g. research papers, will be announced at the begin of the semester
Module examination	See module description
Module examination assessment	Graded according to published grading scheme
Hints	

Modulbeschreibung zum Modul 11

Module title	Data Mining Methods
Module number	11
Study program	M.Sc. Program High Integrity Systems
Module code	
Level	M.Sc.
Applicability	yes
Duration	1
Status	Mandatory module
Recommended semester	2
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Basic skills in statistics as they are offered in the Bachelor program Informatik, i.e students should be able to perform the most important methods of inferential statistics in line with some real-world problems, the students should be able to interpret and assess the results of basic statistical methods
Requirements for module examination	50% Regular attendance at exercise groups, (unit Data Mining Methods – Exercises) solutions to 40% of weekly exercises in unit Data Mining Methods - Exercises short written exposé as stated in unit Data Mining Methods - Exercises
Examination type	Written (computer) examination of 90 minutes duration
Education goals/ capabilities	• Awareness of different data types, data scales, data use as endogenous and exogenous • Skills in data recovery and data pre-processing • Theoretical understanding of statistical methods for information extraction • Capacity to use the computer to solve problems in real world data mining problems • Capacity to under stand and judge results of statistical analysis in the context of data mining • Awareness of dangers of misuse and misinterpretation • Capacity to communicate using statistical language, i.e. explain procedures, results of an analysis and a critique of the results

	Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	Data Mining Methods - lectures Data Mining Methods - exercises
Unit teaching modes	Lectures using multimedia presentation techniques Exercises with a PC and statistical programming language in Computer pool to solve problems
Total workload (h)	150
Language	English
Module frequency	Annual, Summer term
Module coordination	Prof. Dr. Andreas Orth
Hints	

Unitbeschreibung zum Modul 11

Unit name	Data Mining Methods - Lectures
Code	
Module name	Data Mining Methods
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Introduction to Data Mining (data types, data scales, roles of variables in an analysis, methods pertaining to different scales and types of variables, the data mining workflow, Loss functions) • Introduction to a statistical programming language (alternatively: R (or S-plus), SAS, SPSS, etc.) • Theory behind important methods data mining and inference selection out of o linear modelling – GLM, GLIM, mixed effects modelling, variable selection methods - , o Methods for Classification – prototype-methods, k-nearest neighbour classifiers, Linear Discriminant Analysis, logistic regression, seperating hyperplanes, support vector machines etc. o Additive Models, Trees, Boosting Methods, Additive Trees o Neural nets o Unsupervised Learning • Variance Estimation and Validation methods (selection out of bootstrapping, jackknifing, cross-validation, Baysian methods, EM-algorithm, MCMC)
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• SAS – Online Documentation • R project-Documentation • Hastie, Tibshirani & Friedman: The Elements of Statistical

	Learning: Data Mining, Inference, and Prediction. Springer (2001) • Berthold & Hand: Intelligent Data Analysis: An Introduction. (1999) • John Fox: Applied Regression Analysis and Generalized Linear Models. Sage Publications (1998) • Efron, B; Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman&Hall/CRC (1993) • Christopher Bishop: Neural Networks for Pattern Recognition. (1995)
Module examination	Prerequisite: Successful participation at Unit: Data Mining Methods - Exercises Written (computer) examination of 90 minutes duration

Unitbeschreibung zum Modul 11

Unit name	Data Mining Methods - Exercises
Code	
Module name	Data Mining Methods
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Computer Exercises pertaining to the contents described in the unit Data Mining Methods – lectures • short written exposé of one real world problem, inlcuding reasoning on why which methods were selected, inlcuding interpretation and critique of results obtained • lesson's learned session after written exposé • exam preparation session for the Module examination
Teaching mode	Using PC and statistical programming language in Computer pool to solve problems
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• literature as in Data Mining Methods – Lectures, in addition: • Davison, A.C.; Hinkley, D.V.: Bootstrap Methods and their Applications. Cambridge University Press (1997) • C. R. Robert, G. Casella: Introducing Monte Carlo Methods with R. Springer (2010) • John Fox, Stanford Weisberg: An R companion to Applied Regression. Sage Publications (2011) • Data Mining Group (2011): http://www.dmg.org/ (Zugriff 11.8. 2011) •
Module examination	50% Regular attendance at exercise groups, (unit Data Mining Methods – Exercises) solutions to 40% of weekly exercises in unit Data Mining Methods - Exercises Compilation of a short written exposé (see above)

Module examination assessment	Graded according to published grading scheme

Modulbeschreibung zum Modul 12.1 System Theory and Modeling

Module title	System Theory and Modeling
Module number	12.1
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Applicable in other M.Sc. Programs in computer science
Duration	1 semester
Status	Elective subject
Recommended semester	2^{nd} /1^{st} semester
Credits	5 CP
General recommended prerequisites	none
Recommended prerequisites	• Good knowledge in discrete mathematics, analysis, numerical methods
Requirements for module examination	None
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	Upon completion of this course, the student is able to: • understand the foundations of systems theory, • comprehend the importance of HW/SW system modeling, • assess different modeling techniques, • apply system modeling techniques to real world application prototype examples. Non specialist compentencies (15% of total workload): • Cultural and social aspects of project work in international R&D teams • Scientific literature research and handling
Module units	System Theory and Modeling – lectures System Theory and Modeling - exercises
Unit teaching modes	Interactive lectures using multimedia presentation techniques Exercises: Teamwork

Total workload (h)	150
Language	English
Module frequency	Annuel, Summer term
Module coordination	Prof. Dr. Manfred Hannemann
Hints	

Unitbeschreibung zum Modul 12.1

Unit name	System Theory and Modeling - Lectures
Code	
Module name	System Theory and Modeling
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Matthias Wagner

Contents	• Systems theory
	  • Principles
	  • System Analysis
	  • Structures and Classes
	  • Complexity and Catastrophes
	  • Neighboring disciplines
	• Modeling
	  • Types and categories
	  • Math Tools
	  • State Models
	  • Functional Modeling
	  • Process Modeling
	• Applications
	• Dynamical Systems
	• Controlling
	• Synergetics
	• Prototype Examples

Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• B. P. Zeigler et al.: Theory of Modeling and Simulation, 2nd ed. Academic Press, 2000 • A. B. Shiflet; G. W. Shiflet: Introduction to Computational Science. Princeton University Press, 2006 • St. L. Campbell et al: Modeling and Simulation in Scilab/Scicos. Springer, 2006 • L.v. Bertalanffy: General System Theory, George Braziller Inc., New York, 1968

	• H. Anton, Calculus, A new horizon, Sixth Edition, John Wiley and Sons, New York, 1999
Module examination	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 12.1

Unit name	System Theory and Modeling - Exercises
Code	
Module name	System Theory and Modeling
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Matthias Wagner
Contents	• Lab exercises with software tools pertaining to the contents described in the unit System Theory and Modeling – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small R&D groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• See unit System Theory and Modeling
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 12.2 Transaction Management

Module title	Transaction Management
Module number	12.2
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Applicable in other M.Sc. Programs in computer science
Duration	1 semester
Status	Elective subject
Recommended semester	2^{nd} /1^{st} semester
Credits	5 CP
General recommended prerequisites	none
Recommended prerequisites	• Good knowledge in the use of database systems and programming with higher programming languages This corresponds to the following bachelor module: • Databases • Programming
Requirements for module examination	None
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	• Understanding the concept of a transaction. • Understanding how the deployment of transaction systems can increase the robustness of a system without adding additional complexity to the application development. • Knowledge of algorithms to handle problems resulting from concurrent access to data and errors resulting from system failures. Non specialist compentencies (15% of total workload): • Cultural and social aspects of project work in international R&D teams • Scientific literature research and handling
Module units	Transaction Management – lectures

	Transaction Management - exercises
Unit teaching modes	Interactive lectures Exercises: Teamwork in R&D-groups
Total workload (h)	150
Language	English
Module frequency	Annuel, Summer term
Module coordination	Prof. Dr. Justus Klingemann
Hints	

Unitbeschreibung zum Modul 12.2

Unit name	Transaction Management - Lectures
Code	
Module name	Transaction Management
Lecturers	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Contents	• Concept of transactions • Theory of serialization • Concurrency Control • Recovery • Distributed transactions • Extended transaction models
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• G. Vossen, G. Weikum: • Transactional Information Systems, Morgan Kaufmann • Garcia-Molina, H., J. D. Ullman and J. D. Widom, *Database Systems: The Complete Book*, Pearson/Prentice Hall. • • Kifer, M., A. Bernstein and P.M. Lewis, *Database Systems: An Application-Oriented Approach*. Addison Wesley / Pearson. • Ramakrishnan, R. and J. Gehrke, *Database Management Systems*, McGraw-Hill, hardcover as well as eBook
Module examination	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul 12.2

Unit name	Transaction Management - Exercises
Code	
Module name	Transaction Management
Lecturers	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Contents	• Lab exercises with software tools pertaining to the contents described in the unit Transaction Management – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small R&D groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• See unit Transaction Management
Module examination	none
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 13.1

Module title	Multivariate Data Analysis
Module number	13.1
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	yes
Duration	1 semester
Status	Elective subject
Recommended semester	3rd semester
Credits	5 CP
General recommended prerequisites	successful participation in courses Introductory Data Analysis - Exercises successful participation in courses Data Mining Methods - Exercises
Recommended prerequisites	Understanding of univariate and bivariate methods as taught in module Introductory Data Analysis, Experience in applying statistical methods to real world data
Requirements for module examination	50% Regular attendance at exercise groups, (unit Multivariate Data Analysis – Exercises) solutions to 40% of weekly exercises in unit Multivariate Data Analysis - Exercises short written exposé as stated in unit Multivariate Data Analysis - Exercises
Examination type	Written (computer) examination of 90 minutes duration
Education goals/ capabilities	• Understanding of structure of data from automated processes • Understanding of Data pre-processing methods (data compression, data alignment, data transformations etc.) • Understanding of collinearity problem and ways to deal with it • Capacity to apply technologies to real world situations • Capacity to analyse a data analysis project, determine pre-processing steps, try out different statistical technologies • Interpret results in the context of an application and a given problem setting • Draw conclusions and communicate results and procedures of

	a data analysis project Non specialist compentencies (15% of total workload): • Cultural and social aspects of project work in international R&D teams • Presentation skills • Communicate with a customer to understand a problem setting • Scientific literature research and handling
Module units	Multivariate Data Analysis – lectures Multivariate Data Analysis - exercises
Unit teaching modes	Lectures using multimedia presentation techniques Group work
Total workload (h)	150
Language	English
Module frequency	biannual
Module coordination	Prof. Dr. Andreas Orth
Hints	

Unitbeschreibung zum Modul 13.1

Unit name	Multivariate Data Analysis - Lectures
Code	
Module name	Data Mining Methods
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Introduction to process data analysis • Multivariate Methods, Data Compression Methods (e.g. Wavelets) • Collinearity and coefficient shrinking methods (projection methods) • Working with 3d-data cubes – batch data, alignment techniques • Criteria and Questions to ask in a data analysis project • Organizing the workflow of a consulting project and enhancing efficiency
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• Hastie, Tibshirani & Friedman: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer (2001) • L Eriksson et. al Multi- and Megavariate Data Analysis Part I Basic Priciples and Applications & Part II Advanced Applications and Method Extensions. Umetrics Academy (2006)
Module examination	Prerequisite: Successful participation at Unit: Multivariate Data Analysis - Exercises Written (computer) examination of 90 minutes duration

Unitbeschreibung zum Modul 13.1

Unit name	Multivariate Data Analysis - Exercises
Code	
Module name	Multivariate Data Analysis
Lecturers	Prof. Dr. Andersson, Prof. Dr. Behl, Prof. Dr. Falkenberg, Prof. Dr. Orth
Contents	• Computer Exercises pertaining to the contents described in the unit Data Mining Methods – lectures • group work on real world problem, “From the offer to the invoice” including time estimation, analysing customer requirements • lesson’s learned session after group work • exam preparation session for the Module examination
Teaching mode	Using PCs in the Computer pool to solve problems Group work
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	• literature as in unit Multivariate Data Analysis – Lecture, in addition: • Aalst van der, W. und K. van Hee (2002): Workflow Management - Models, Methods, and Systems, MIT Press, Cambridge, Massachusetts
Module examination	• 50% Regular attendance at exercise groups, (unit Multivariate Data Analysis – Exercises) • solutions to 40% of weekly exercises in unit Data Mining Methods - Exercises • Participation in simulated data analysis project “From the offer to the invoice”
Module examination assessment	Graded according to published grading scheme

Modulbeschreibung zum Modul 13.2 Simulation Methods

Module title	Simulation Methods
Module number	13.2
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Lectures Exercises
Level	M.Sc.
Applicability	Applicable in other M.Sc. Programs in computer science
Duration	1 semester
Status	Elective subject
Recommended semester	3rd semester
Credits	5 CP
General recommended prerequisites	none
Recommended prerequisites	Good knowledge in discrete mathematics, calculus, numerical methods, contents of module System Theory and Modeling
Requirements for module examination	Successful participation in unit exercises 50% Regular attendance at exercise groups, (unit Simulation Methods – Exercises) solutions to 40% of weekly exercises in unit Simulation Methods - Exercises
Examination type	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	Upon completion of this course, the student is able to: • assess the growing importance of simulation for high-integrity systems, • understand the interaction between simulation and experimental verification, • get an overview over simulation methods, • get experience in using simulation tools, • recognize the limitations of simulation work.
Module units	Simulation Methods – lectures

	Simulation Methods - exercises
Unit teaching modes	Interactive lectures using multimedia presentation techniques Exercises: Teamwork
Total workload (h)	150
Language	English
Module frequency	Summer term
Module coordination	Prof. Dr. Manfred Hannemann

Unitbeschreibung zum System Theory and Modeling 13.2: Simulation Methods - Lectures

Unit name	Simulation Methods - Lectures
Code	
Module name	Simulation Methods
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Matthias Wagner
Contents	• Methods Approximation techniques Types and categories Software tools Numerical methods Visualization • Validation Simulation and Measurement • Applications
Teaching mode	Interactive group lecturing
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. exam preparation	10
Practical part	
Self study	30
Language	English
Literature	• L.v. Bertalanffy: General System Theory, George Braziller Inc., New York, 1968 • H. Anton, Calculus, A new horizon, Sixth Edition, John Wiley and Sons, New York, 1999 • B. P. Zeigler et al.: Theory of Modeling and Simulation, 2nd ed. Academic Press, 2000 • A. B. Shiflet; G. W. Shiflet: Introduction to Computational Science. Princeton University Press, 2006 • St. L. Campbell et al: Modeling and Simulation in Scilab/Scicos. Springer, 2006
Module examination	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme

Unitbeschreibung zum Modul Simulation Methods 13.2: Simulation Methods - Exercises

Unit name	Simulation Methods - Exercises
Code	
Module name	Simulation Methods
Lecturers	Prof. Dr. Manfred Hannemann, Prof. Dr. Matthias Wagner
Contents	• Lab exercises with software tools pertaining to the contents described in the unit System Theory and Modeling – lectures • practical teamwork on real world problems • lesson's learned session after group work
Teaching mode	Teamwork in small R&D groups
Weekly hours	2
Total Workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	30
Self study	50
Language	English
Literature	See unit Simulation Methods
Module examination	Successful participation in unit exercises 50% Regular attendance at exercise groups, (unit Simulation Methods – Exercises) solutions to 40% of weekly exercises in unit Simulation Methods - Exercises
Module examination assessment	Not differentiated
Hints	

Modulbeschreibung zum Modul 14.1

Module title	Standards and Certification
Module number	14.1
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Seminar
Level	M.Sc.
Applicability	
Duration	1
Status	Elective Subject
Recommended semester	3rd semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Safety Critical Computer Systems
Requirements for module examination	none
Examination type	• Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. • The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Education goals/ capabilities	• Upon completion of this course, the student is able to: • assess the growing pressure to standardize the development of high-integrity systems, • understand the growing importance of software safety, • survey the body of standards, • distinguish between standards of different application fields, • understand the history of engineering for safety, • achieve the ability for certification work, • understand the roles of management and staff in certification work. Training for non-specialist competencies (25% of the total workload): Students learn

	• to search for, read, summarize and cite scientific literature on a large scale; •to read and interpret national and international standards; •to write a report as a scientific paper; •to give a scientific talk.
Module units	• Unit Seminar
Unit teaching modes	Seminar
Total workload (h)	150
Language	English
Module frequency	Alternating with modules 14.2, 14.3
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 14.1

Unit name	Certification and Standards - Seminar
Code	
Module name	Certification and Standards
Lecturers	Prof. Dr. Matthias Wagner
Contents	• International standards for safety critical computer systems • Overview over certification in various fields of application • Commonalities and differences of various standards
Teaching mode	Seminar
Weekly hours	2
Total workload (h)	150
Attendance (h)	36
Exam incl. exam preparation	
Practical part	
Self study	114
Language	English
Literature	Debra Herrmann Software Safety and Reliability IEEE Computer Society 1999 Current literature, e.g. journal papers, conference proceedings etc., will be announced at the begfinning of the semester
Module examination	• Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. • The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Module examination assessment	Graded according to published grading scheme

Modulbeschreibung zum Modul 14.2

Module title	Current Topics in High Integrity Systems
Module number	14.2
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Seminar
Level	M.Sc.
Applicability	
Duration	1
Status	Elective Subject
Recommended semester	3rd semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	Safety Critical Computer Systems
Requirements for module examination	none
Examination type	Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Education goals/ capabilities	• Upon completion of this course, the student is able to: • recognize important developments in the field of High Integrity Systems, • incorporate new methods into the software and systems development process • criticize new technology with respect to their usability in critical systems development. Training for non-specialist competencies (25% of the total workload): Students learn • to search for, read, summarize and cite scientific literature on a large scale; •to read and interpret national and international publications;

	●to write a report as a scientific paper; ●to give a scientific talk.
Module units	●Unit Seminar
Unit teaching modes	Seminar
Total workload (h)	150
Language	English
Module frequency	Alternating with modules 14.1, 14.3
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 14.2

Unit name	Seminar: Current Topics in High Integrity Systems
Code	
Module name	Current Topics in High Integrity Systems
Lecturers	All professors of the Master's program High Integrity Systems
Contents	•Current topics in Computer Science with respect to the •analysis •design, •development and • maintenance of High-Integrity Systems
Teaching mode	Seminar
Weekly hours	2
Total workload (h)	150
Attendance (h)	36
Exam incl. exam preparation	
Practical part	
Self study	114
Language	English
Literature	Current research literature
Module examination	Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Module examination assessment	Graded according to published grading scheme

Modulbeschreibung zum Modul 14.3

Module title	Internet of Things
Module number	14.3
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Seminar
Level	M.Sc.
Applicability	
Duration	1
Status	Elective Subject
Recommended semester	3rd semester
Credits	5
General recommended prerequisites	none
Recommended prerequisites	
Requirements for module examination	none
Examination type	Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Education goals/ capabilities	Upon completion of this course, the student is able to: • understand the basic technologies for the Internet of Things, • asses emerging technologies concerning their suitability, • get acquainted quickly with new technologies, and • develop new application fields. Training for non-specialist competencies (25% of the total workload): Students learn •to search for, read, summarize and cite scientific literature on a large scale; •to read and interpret national and international standards; •to write a report as a scientific paper; •to give a scientific talk.

Module units	Seminar
Unit teaching modes	Seminar
Total workload (h)	150
Language	English
Module frequency	Annual
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 14.3

Unit name	Internet of Things - Seminar
Code	
Module name	Internet of Things
Lecturers	Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Deegener, Prof. Dr. Matthias Wagner
Contents	The course will cover selected subjects from the following areas. The depth of coverage might vary. • Technological foundation of the Internet of Things • HW Basics • Field-Bus systems • Wireless sensor networks • Middleware and integration into the Internet • Example(s) of relevant algorithms • HMI • Application examples
Teaching mode	Seminar
Weekly hours	2
Total workload (h)	150
Attendance (h)	36
Exam incl. exam preparation	
Practical part	
Self study	114
Language	English
Literature	
Module examination	Paper written according to international scientific journal standards and oral presentation (30 minutes) according to international scientific conference standards. The grade is calculated by the arithmetic mean of the marks for the written report and oral presentation
Module examination assessment	Graded according to published grading scheme

Modulbeschreibung zum Modul M15

Module title	Formal Specification and Verification (M. Sc.)
Module number	15
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	2 weekly hrs Lectures Formal Specification and Verification 2 weekly hrs Exercises Formal Specification and Verification
Level	M.Sc.
Applicability	Usable in other Computer Science Master programs
Duration	1 Semester
Status	Mandatory module
Recommended semester	3rd semester
Credits	5
General recommended prerequisites	None
Recommended prerequisites	• Basic knowledge of propositional and predicate logic • Basic knowledge of algorithm design and analysis • Basic knowledge of automata theory
Requirements for module examination	None
Examination type	Written examination of 90 minutes duration
Education goals/ capabilities	• Understanding the principles of formal specification and verification. • Understanding the theory (models and logics) used in model checking. • Reasoning about safety, liveness and fairness properties in concurrent systems. • Specifying safety and liveness properties of concurrent systems using temporal logic and/or computational tree logic. • Verifying that a concurrent system satisfies certain safety and liveness properties using model checking algorithms. • Understanding the limitations of model checking. Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module units	Lectures Formal Specification and Verification

	Exercises Formal Specification and Verification
Unit teaching modes	Lectures and Exercises
Total workload(h)	150 h
Language	English
Module frequency	Summer and Winter term
Module coordination	Prof. Dr. Ruth Schorr
Hints	

Unitbeschreibung zum Modul 15

Unit name	Lectures Formal Specification and Verification
Code	
Module name	Formal Specification and Verification
Lecturers	Prof. Dr. Gerd Döben-Henisch, Prof. Dr. Ruth Schorr
Contents	The lectures provide an introduction to the main principles of model checking: • Modeling reactive systems by transition systems • Linear time properties and Büchi automata • Linear temporal logic and automata-based model checking • Computation tree logic • Timed automata
Teaching mode	Lectures
Weekly hours	2
Total workload (h)	70
Attendance (h)	30
Exam incl. Exam preparation	10
Practical part	0
Self study	30
Language	English
Literature	Baier, Christel and Katoen, Joost-Pieter: Principles of Model Checking, MIT Press, 2008. Current literature will be announced at the beginning of each semester.
Module examination	Written examination of 90 minutes duration
Module examination assessment	Graded according to published grading scheme
Hints	

Unitbeschreibung zum Modul 15

Unit name	Exercises Formal Specification and Verification
Code	
Modul name	Formal Specification and Verification
Lecturers	Prof. Dr. Gerd Döben-Henisch, Prof. Dr. Ruth Schorr
Contents	Exercises and examples to ensure that the students learn to solve problems using the methods from the lectures. To support the learning process continuous feedback is provided.
Teaching mode	Exercises
Weekly hours	2
Total workload (h)	80
Attendance	30
Exam incl. Exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	See Unit Lectures Formal Specification and Verification
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 16.1

Module Title	Selected Subjects in Current Web Engineering
Module number	16.1
Study program	M.Sc. Program High-Integrity Systems
Module code	
Units	2 weekly hrs: Lectures Selected Subjects in Current Web Engineering 2 weekly hrs: Exercises Selected Subjects in Current Web Engineering
Level	M.Sc. Advanced Level
Applicability	Applicable in other computer science and engineering master curricula
Duration	1 Semester
Status	Elective module
Recommended semester	3rd semester
Credits	5
Conditions for module participation	None
Recommended prerequisites	Basic understanding of Distributed Systems and basic command of a high level OO language (such as Java) for practical exercises.
Requirements for module examination	None
Module examination	Written examination at the end of the semester
Education goals/ capabilities	Web architectures play an important and ever increasing role in organizing IT on a large scale. Web applications and algorithms have an important impact on society and how information is processed and consumed. Upon completion of this course, the students • have a *basic* understanding of the fundamental principles of Web Engineering, such as Web-protocols and architectures, relevant algorithms, data semantics and (Web-) UI and how these relate to each other • are able to plan and architect information systems based on those principles • have a *deep* understanding of at least one selected subject

	from Web-protocols and architecture, relevant algorithms, data semantics and (Web-) UI (depending on the actual lecture and the student's interest) Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module content	Lectures Selected Subjects in Current Web Engineering Exercises Selected Subjects in Current Web Engineering
Teaching mode	Lectures and Exercises
Total workload (h)	150 h
Language	English
Module frequency	Bi-annual
Module coordination	Prof. Dr. Jörg Schäfer
Hints	

Unitbeschreibung zum Modul 16.1

Unit name	Selected Subjects in Current Web Engineering - Lectures
Code	
Module name	Selected Subjects in Current Web Engineering
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer

Content	The course will cover selected subjects from the following areas. The depth of coverage might vary:
	• Web-protocols and architectures
	  • Web History and Evolution
	  • HTTP and REST
	  • Web Standards (e.g. HTTP, HTML, CSS)
	  • Web Technologies (Web Applications, Web Services, Semantic Web)
	  • Architecting for QoS such as Scalability, Performance, no SPoF, Agility and Maintainability etc.
	  • Security
	  • Important architectural patterns
	• Relevant algorithms
	  • CAP-Theorem (including a discussion of consequences such asD vs BASE, NO-SQL movement etc.)
	  • Selected application algorithms for distribution such as e.g. MapReduce
	  • Selected application algorithms for search and data (web) mining such as e.g. Pagerank, Web crawling, search, social network analysis, opinion mining and sentiment analysis, Web usage (query log) mining, query log mining, etc,
	• Data semantics
	  • REST revisited (MIME Types)
	  • Vision of the Semantic Web and its main technologies such as e.g. RDF, OWL, SPARQL and RIF.
	  • Web Service Modeling Ontology and Web Services
	• (Web-) UI
	  • Architectural/Design principles for UI such as MVC

	• Rich Internet Applications (RIA) • Technologies for UI (e.g. HTML/CSS, CGI, Servlets/JSPs, Javascript, JSF, Silverlight, AJAX, HTML 5)
Teaching mode	Lectures
SWS	2
Total workload (h)	70
Attendance	30
Exam incl. exam preparation	10
Practical part	0
Self study	30
Language	English
Literature	• T. Berners-Lee, J. Hendler, and O. Lassila , The Semantic Web, Scientific American, 2001 • Seth Gilbert and Nancy Lynch. “Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services.” Sigact News, 33(2), 2002. • Carl Henderson, Building Scalable Web Sites: Building, Scaling, and Optimizing the Next Generation of Web Applications, O'Reilly Media; 1 edition, 2006 • I. Jacobs, N. Walsh (ed.), Architecture of the World Wide Web, W3C Recommendation 15 December 2004, http://www.w3.org/TR/webarch/, Parts 1,2,3 • Bing Liu, Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications), Springer; 2nd Edition, 2011 • Leon Shklar, Rich Rosen, Web Application Architecture: Principles, Protocols and Practices, Wiley; 2 edition, 2009 • Roy Thomas Fielding. Architectural Styles and the Design of Network- based Software Architectures. University of California, Irvine, 2000. Current literature recommendations will be given at each semester start.
Module examination	Written examination at the end of the semester
Module examination assessment	Graded according to published grading scheme
Hints	

Unitbeschreibung zum Modul 16.1

Unit name	Selected Subjects in Current Web Engineering - Exercises
Code	
Module name	Selected Subjects in Current Web Engineering
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Content	While the lectures provide the theoretical background, the exercises will enable students to verify their understanding and deepen their knowledge. To this end, selected research papers are studied in the exercises. In addition, the students will code example(s) in the course of this unit. Henceforth, the students will receive continuous feedback, which will support the educational objectives.
Teaching mode	Exercises
SWS	2
Total workload (h)	80
Attendance	30
Exam incl. exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	see Lectures Selected Subjects in Current Web Engineering
Module examination	None
Module examination assessment	NA
Hints	

Modulbeschreibung zum Modul 16.2

Module Title	Mobile Systems and Applications
Module number	16.2
Study program	M.Sc. Program High-Integrity Systems
Module code	
Units	2 weekly hrs: Lectures Mobile Systems and Applications 2 weekly hrs: Exercises Mobile Systems and Applications
Level	Advanced Level
Applicability	Applicable in other computer science and engineering master curricula
Duration	1 Semester
Status	Elective module
Recommended semester	3rd semester
Credits	5
Conditions for module participation	None
Recommended prerequisites	Good knowledge in programming with C and Java, foundations of computer networks
Requirements for module examination	None
Module examination	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	Upon completion of this course, the students • understand the role and specific challenges of mobile computing • understand the foundations of mobile computing including theoretical concepts, technologies and tools • are able to apply their skills and choose technologies accordingly • are able to develop and deploy mobile applications Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module content	Lectures Mobile Systems and Applications

	Exercises Mobile Systems and Applications
Teaching mode	Lectures and Exercises
Total workload (h)	150 h
Language	English
Module frequency	Annuel
Module coordination	Prof. Dr. Jörg Schäfer
Hints	

Unitbeschreibung zum Modul 16.2

Unit name	Mobile Systems and Applications - Lectures
Code	
Module name	Mobile Systems and Applications
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Content	The course will cover selected subjects from the following areas. The depth of coverage might vary: • What is Mobile Systems and Applications? • Mobile Computing ecosystem • Mobile and Ubiquituous Computing • Communication mechanisms, mobile networks • Context awareness, sensors • Location Based Services (LBS) • Mobile application platforms including (but not limited to) Android, iOS, Windows Mobile (.Net CF) • Data and service management • Component and modularity frameworks (e.g. OSGi and others) • Mobile Services • Security aspects of mobile applications
Teaching mode	Lectures
SWS	2
Total workload (h)	70
Attendance	30
Exam incl. exam preparation	10
Practical part	0
Self study	30
Language	English
Literature	• Reza B'Far, Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XM, Cambridge University Press (November 1, 2004) • Ed Burnette, Hello, Android: Introducing Google's Mobile Development Platform (Pragmatic Programmers), Pragmatic Bookshelf; Third Edition edition (August 4, 2010) • Kwok, Lau: Wireless Internet and Mobile Computing: Wireless Internet and Mobile Computing: Interoperability

	and Performance, Wiley and Sons Verlag, (September 4, 2007) • Jochen Schiller, Mobile Communications (2nd Edition), Addison Wesley; 2 edition (September 21, 2003) • Ivan Stojmenovic, Handbook of Wireless Networks and Mobile Computing, Wiley-Interscience; 1st edition (February 8, 2002) • Andrew Tanenbaum – Computer Networks, Prentice Hall; 5 edition (October 7, 2010) • Mark Weiser. The computer for the 21st century, ACM SIGMOBILE Mobile Computing and Communications Review - Special issue dedicated to Mark Weiser, Volume 3 Issue 3, July 1999. Current literature recommendations will be given at each semester start.
Module examination	Written examination at the end of the semester
Module examination assessment	Graded according to published grading scheme
Hints	

Unitbeschreibung zum Modul 16.2

Unit name	Mobile Systems and Applications - Exercises
Code	
Module name	Mobile Systems and Applications
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Content	While the lecture provides the theoretical background, the exercises will enable students to apply the concepts. The students will code practical mobile application(s) in the course of this unit. Henceforth, the students will receive continuous feedback, which will support the educational objectives.
Teaching mode	Exercises
SWS	2
Total workload (h)	80
Attendance	30
Exam incl. exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	see Lecture Mobile Systems and Applications
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 16.3

Module Title	Cloud Computing
Module number	16.3
Study program	M.Sc. Program High-Integrity Systems
Module code	
Units	2 weekly hrs: Lectures Cloud Computing 2 weekly hrs: Exercises Cloud Computing
Level	Advanced Level
Applicability	Applicable in other computer science and engineering master curricula
Duration	1 Semester
Status	Elective module
Recommended semester	3rd semester
Credits	5
Conditions for module participation	None
Recommended prerequisites	Good knowledge in software engineering, computer networks, databases and distributed applications and one high-level programming language.
Requirements for module examination	None
Module examination	Written examination of 90 minutes duration at the end of the semester
Education goals/ capabilities	Cloud Computing provides scalable IT resources "on demand" using technologies such as virtualization. Access to these resources is abstracted via APIs and frameworks - often based on Web-Services. It is expected that Cloud Computing has a major impact on IT infrastructure of enterprises and business models. Upon completion of this course, the students • understand the concepts and technologies fundamental for Cloud Computing • understand the economical and operational impact of Cloud Computing for providing IT-resources within the enterprise • is able to apply a structured, scientific process to evaluate architecture alternatives for Cloud Computing

	• are able to architect and implement Cloud Computing solutions. Non specialist competencies (15% of the total workload): • Team work • Communication in international teams
Module contents	Lectures Cloud Computing Exercises Cloud Computing
Teaching mode	Lectures and Exercises
Total workload (h)	150 h
Language	English
Module frequency	Annuel
Module coordination	Prof. Dr. Jörg Schäfer
Hints	

Unitbeschreibung zum Modul 16.3

Unit name	Cloud Computing - Lectures
Code	
Module name	Cloud Computing
Lecturer	Prof. Dr. Martin Kappes, Prof. Dr. Jörg Schäfer
Content	The course will cover selected subjects from the following areas. The depth of coverage might vary: • Definitions of Cloud Computing and Core Foundations of Cloud Computing • Virtualization technologies • SOA and Web-Services • Different Cloud Computing architectures (SaaS, PaaS, IaaS) • Different Cloud Computing vendor stacks including open source • Service Management for the cloud • Algorithms for Cloud Computing (e.g. MapReduce) • Security aspects of Cloud Computing • Operational aspects of Cloud Computing • Economical aspects of Cloud Computing
Teaching mode	Lectures
SWS	2
Total workload (h)	70
Attendance	30
Exam incl. exam preparation	10
Practical part	0
Self study	30
Language	English
Literature	• Nick Antonopoulos and Lee Gillam: Cloud Computing: Principles, Systems and Applications, Springer, 2010 • Charles Babcock, Management Strategies for the Cloud Revolution: How Cloud Computing Is Transforming Business and Why You Can't Afford to Be Left Behind, Mcgraw-Hill Professional; Edition: 1, 2010 • Iain D. Craig, Virtual Machines, Springer; Softcover reprint of hardcover 1st ed. 2006 edition 2011 • Tim Mather, Subra Kumaraswamy, and Shahed Latif: Cloud Security and Privacy: An Enterprise Perspective on

	Risks and Compliance O'Reilly Media; Edition: 1, 2009 • Frank Munz, Middleware and Cloud Computing: Oracle on Amazon Web Services (AWS), Rackspace Cloud and Rightscale, munz & more; Edition: 1, 2011 • Jim Smith and Ravi Nair,Virtual Machines: Versatile Platforms for Systems and Processes (The Morgan Kaufmann Series in Computer Architecture and Design), Morgan Kaufmann; 1 edition 2005 • Georg Reese, Cloud Application Architectures: Transactional Systems for EC2 and Beyond, O'Reilly Media; Edition: 1, 2009 Current literature recommendations will be given at the semester start.
Module examination	Written examination at the end of the semester
Module examination assessment	Graded according to published grading scheme
Hints	

Unitbeschreibung zum Modul 16.3

Unit name	Cloud Computing - Exercises
Code	
Module name	Cloud Computing
Lecturer	Prof. Dr. Martin Kappes, Prof. Dr. Jörg Schäfer
Content	While the lectures provide the theoretical background, the exercises will enable students to apply the concepts. The students will read current research literature and vendor documentation and configure examples accordingly. Im addition, simple prototypes will be developed. Henceforth, the students will receive continuous feedback, which will support the educational objectives.
Teaching mode	Exercises
SWS	2
Total workload (h)	80
Attendance	30
Exam incl. exam preparation	0
Practical part	0
Self study	50
Language	English
Literature	see Lectures Cloud Computing
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 17

Module title	High Integrity Systems Project
Module number	17
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	4 weekly hrs: Group Project
Level	Master
Applicability	
Duration	1 semester
Status (	Mandatory
Recommended semester	3rd semester
Credits	10
Conditions for module participation	none
Recommended prerequisites	Recommended: Excellent knowledge in software and systems engineering, Very good programming skills Mathematics
Conditions for examination	The project should be worked out in a team of students (no more than four) with a 2-weekly written report of each participant describing essential aspects of the process from the point of view of each participant.
Module examination	Written report in the form of a scientific paper and an oral presentation of project results in the form of a scientific conference talk according to the rules of a scientific society, i.e. IEEE.
Education goals/ capabilities	Upon completion of this course, the student is able to: • develop a high-integrity software application with real-world requirements, • gain experience in all fields of software eand systems engineering and certification of high-integrity software, • and assess the problems of applying scientific knowledge in a real world R&D – situation. Training for non-specialist competencies (25% of the total workload): Students learn

	• to explore and to adapt to a R&D environment; • to organize a research team; • to use modern tools for project organization; • to make industrial presentations; • work in a group environment with distributed responsibilities; • to write a report as a scientific paper.
Module Contents	The project may be based on a collaboration with an industrial partner or an external research institute. The topic of the project covers several core areas of computer science with applications in High Integrity Systems.
Unit teaching modes	Project
Total workload (h)	300
Language	English
Module frequency	Each semester
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 17

Unit name	High Integrity Systems Project
Code	
Module name	High Integrity Systems Project
Lecturers	All professors of the High Integrity Systems program
Contents	• The project may be based on a collaboration with an industrial partner or an external research institute. • The topic of the project covers several core areas of computer science with applications in High Integrity Systems.
Teaching mode	R&D project with small groups
Weekly hours	4
Total workload (h)	300
Attendance	60
Exam incl. exam preparation	20
Practical part	150
Self study	220
Language	English
Literature	Current research literature
Module examination	None
Module examination assessment	
Hints	

Modulbeschreibung zum Modul 18

Module title	High Integrity Systems Master Thesis
Module number	18
Study program	M.Sc. Program High Integrity Systems
Module code	
Units	Thesis Project
Level	M.Sc.
Applicability	
Duration	5 months
Status	Mandatory
Recommended semester	4th semester
Credits	30
Conditions for module participation	All modules of the first 3 semesters with examinations passed
Recommended prerequisites	Contents and capabilities of all modules
Conditions for examination	Successful completeion of Master's Thesis
Module examination	Master's colloquium of at least 30 and maximum 60 minutes duration
Education goals/ capabilities	Upon completion of the master thesis, the student is able to: • develop completely an extensive high-integrity software application with real-world requirements, • work in a larger group environment with distributed responsibilities, • gain experience in all fields of software engineering and certification of high-integrity software, • and assess the problems of applying scientific knowledge in a real world R&D – situation. raining for non-specialist competencies (25% of the total workload): Students • practice scientific project management;. • use modern tools for project organization; • write the thesis as a comprehensive scientific report; • defend the thesis in a scientific colloquium.
Module Contents	The thesis project may be based on a collaboration with an industrial partner or an external research institute.

	The topic of the project covers several core areas of computer science with applications in High Integrity Systems.
Unit teaching modes	Research and Development project
Total workload (h)	900
Language	English
Module frequency	Each semester
Module coordination	Prof. Dr. Matthias Wagner
Hints	

Unitbeschreibung zum Modul 18

Unit name	High Integrity Systems Master Thesis
Code	
Module name	High Integrity Systems Master Thesis
Lecturers	All professors of the High Integrity Systems program
Contents	• The project may be based on a collaboration with an industrial partner or an external research institute. • The topic of the project covers several core areas of computer science with applications in High Integrity Systems.
Teaching mode	R&D project leading to the Master Thesis under supervision of 2 professors Research for the thesis may be done at • FH Frankfurt – University of Applied Sciences • IPIAG • External research institutes • Companies
Weekly hours	N/A
Total workload (h)	900
Attendance	N/A
Exam incl. exam preparation	20
Practical part	N/A
Self study	N/A
Language	English
Literature	Current research literature
Module examination	Review of Master's thesis
Module examination assessment	Graded according to published grading scheme
Hints