FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Module handbook

for the consecutive study program

High Integrity Systems

Master of Science (M.Sc.)
Department 2: Department of Computer Science and Engineering

Wissen durch Praxis stärkt

Table of Contents

1. Qualification Objectives (General Program Profile)

Die Beschreibung der Qualifikationsziele folgt dem Qualifikationsrahmen für Deutsche Hochschulabschlüsse für das Master-Niveau und enthält die Rubriken Wissensverbreiterung, Wissensvertiefung, Wissensverständnis, Nutzung und Transfer, Wissenschaftliche Innovation, Kommunikation und Kooperation sowie Wissenschaftliches Selbstverständnis.

Wissensverbreiterung

Die Absolventinnen und Absolventen verfügen über fortgeschrittenes Wissen der Informatik im Zusammenhang mit der Entwicklung und Erforschung kritischer Systeme und sind in der Lage, sich an den aktuellen wissenschaftlichen und technischen Entwicklungen des Gebiets zu beteiligen.

Wissensvertiefung

Die Absolventinnen und Absolventen beherrschen Techniken des wissenschaftlichen Schreibens und des wissenschaftlichen Vortrags, Instrumente des Selbst- und Projektmanagements in allen Phasen der Entwicklung softwaregestützter, kritischer Systeme, sowie der wissenschaftlichen Informationsbeschaffung und –verarbeitung, einschließlich aller relevanten Softwaretools. Sie haben gelernt, Anforderungen, Probleme und Ergebnisse ihrer Arbeit in englischer Sprache zu formulieren. Sie beherrschen je nach den gewählten Wahlmodulen die wesentlichen Methoden der Anforderungsanalyse, des Entwurfs, der Implementierung, des Tests und der Qualitätssicherung, und der Zertifizierung von sicherheitskritischen, missionskritischen und geschäftskritischen computergestützten Systemen. Die erworbenen Methoden qualifizieren die Absolventinnen und Absolventen für die angestrebten beruflichen Tätigkeitsfelder (Forschung & Entwicklung, Qualitätssicherung, Beratung, Projektleitung, Management). Sie kennen die Grundlagen angrenzender Fachgebiete und beziehen diese Kenntnisse in ihre Tätigkeit ein. Sie sind sich der ethischen und gesellschaftlichen Wirkungen ihrer Tätigkeit bewusst.

Wissensverständnis

Bei der Lösung konkreter Forschungs- und Entwicklungsaufgaben wenden sie ihr Wissen an, erkennen die Grenzen von Technologien und Wissenschaft und sind in der Lage, diese anforderungsgerecht zu beurteilen und zu bewerten. Dabei wenden sie das Fachwissen und Erfahrungen an, die sie in ihrem Studium je nach gewählten Wahlmodulen in theoretischen und praxisorientierten Modulen erworben haben.

Nutzung und Transfer

Die Absolventinnen und Absolventen sind in der Lage, sich relevante Informationen zu beschaffen, diese zu verarbeiten und darauf basierende, fundierte Entscheidungen zu treffen. Sie kennen die relevanten Organisations- und Teamstrukturen und –dynamiken und sind in der Lage, im Rahmen eines Teams eine gemeinsame Aufgabenstellung erfolgreich zu bearbeiten.

Wissenschaftliche Innovation

Die Absolventinnen und Absolventen können die Anforderungen an eine technisch/wissenschaftliche Aufgabenstellung beurteilen, Lösungsansätze erforschen und entwickeln und selbstständig umsetzen.

Sie können daraus offene Fragestellungen ableiten und hierfür neue Lösungsansätze auf Basis des aktuellen Standes der Forschung entwickeln. Im Rahmen von Projektarbeiten und der Masterthesis haben sie gelernt, ihre Ergebnisse wissenschaftlich zu dokumentieren, zu präsentieren und vor einem Fachpublikum begründet zu rechtfertigen.

Kommunikation und Kooperation

Die Absolventinnen und Absolventen sind in der Lage, sich in fremde Fach- und Wissenschaftsgebiete einzuarbeiten und zusammen mit Fachleuten fremder Fachgebiete Lösungen für interdisziplinäre Problemstellungen zu finden. Die Absolventen verfügen damit sowohl über die interpersonelle Kompetenz des Arbeitens im Team mit Fachleuten der eigenen Disziplin, als auch der interdisziplinären Teamarbeit. Im Laufe verschiedener Arbeitssituationen während ihres Studiums haben sie kooperatives Lern- und Arbeitsverhalten erworben.

Wissenschaftliches Selbstverständnis/ Professionalität

Die Absolventinnen und Absolventen erkennen die Anforderungen von Unternehmen, staatlichen Organisationen und der Gesellschaft an die Entwicklung kritischer Systeme. Sie sind darauf vorbereitet, Projekt- oder Führungsverantwortung zu übernehmen. Sie entwickeln ihre Sensibilität für die Denkweisen fachfremder Disziplinen und lernen, technische und wissenschaftliche Zusammenhänge im Rahmen unterschiedlicher wissenschaftlicher Disziplinen verständlich zu machen.

Durch den Einblick, den sie in ihrer Fachdisziplin und interdisziplinär erworben haben, sind sie insbesondere darauf vorbereitet, tiefer gehende fachliche Expertise anzufordern und in ihre Aufgaben einzubinden. Sie besitzen damit die entsprechenden Kompetenzen, die sie für die selbständige Arbeit in der Informatik qualifizieren.

Die Absolventinnen und Absolventen erkennen und reflektieren an sie gestellte fachliche Anforderungen ebenso, wie ihre berufliche Verantwortung für Menschen, Gesellschaft und Ökologie

2. Module Overview of Degree Program WS

High Integrity Systems (M.Sc.) **Module overview (start of studies in winter semester)**							Credit points (CP): 120	
Semester 4 Winter semester	Master thesis with colloquium 30 CP						30	
Semester 3 Winter Semester	Compulsory Elective Subjects VI 5 CP	Compulsory Elective Subjects VII 5 CP	Compulsory Elective Subjects VIII 5 CP	Formal Specification and Verification 5 CP	HIS project 10 CP		30	
Semester 2 Summer Semester	Mathematics Update 5 CP	Compulsory Elective Subjects III 5 CP	Compulsory Elective Subjects IV 5 CP	Compulsory Elective Subjects V 5 CP	Advanced IT-Security 5 CP	Data Mining Methods 5 CP	30	
Semester 1 Winter Semester	Safety Critical Computer Systems 5 CP	Advanced Formal Modeling 5 CP	Introductory Data Analysis 5 CP	Implementation of DBMS 5 CP	Compulsory Elective Subjects I 5 CP	Compulsory Elective Subjects II 5 CP	30	

3. Module Overview of Degree Program SS

High Integrity Systems (M.Sc.) Module overview (start of studies in summer semester)							Credit points (CP): 120	
Semester 4 Winter semester	Master thesis with colloquium 30 CP						30	
Semester 3 Summer Semester	Com-pulsory Elective Subjects VI 5 CP	Com-pulsory Elective Subjects VII 5 CP	Com-pulsory Elective Subjects VIII 5 CP	Formal Specifica-tion and Verifica-tion 5 CP	HIS project 10 CP		30	
Semester 2 Winter Semester	Mathe-matics Update 5 CP	Advanced Formal Modeling 5 CP	Intro-ductory Data Analysis 5 CP	Implemen tation of DBMS 5 CP	Com-pulsory Elective Subjects I 5 CP	Com-pulsory Elective Subjects II 5 CP	30	
Semester 1 Summer Semester	Safety Critical Computer Systems 5 CP	Com-pulsory Elective Subjects III 5 CP	Com pulsory Elective Subjects IV 5 CP	Com pulsory Elective Subjects V 5 CP	Advanced IT-Security 5 CP	Data Mining Methods 5 CP	30	

4. ECTS/Workload Overview

No.	Module Title	ECTS CP	Duration [Sem.]	Assessment type	Language	weighting
	1st Semester					
1	Safety Critical Computer Systems (for students starting in the winter semester)	5	1	Written examination (90 minutes)	English	1/24
7	Mathematics Update (for students starting in the summer semester)	5	1	Written examination (90 minutes)	English	1/24
2	Advanced Formal Modeling	5	1	Written computer-based examination (90 minutes) ER*	English	1/24
3	Introductory Data Analysis	5	1	Written computer-based examination (90 minutes) ER*	English	1/24
4	Compulsory Elective Subjects I					
4.1.	Advanced Real-Time Systems	5	1	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)	English	1/24
4.2.	Machine Learning	5	1	Written examination (90 minutes)	English	1/24
5	Implementation of Database Management Systems (DBMS)	5	1	Written examination (90 minutes)	English	1/24
6	Compulsory Elective Subjects II					
6.1	Pattern Oriented Software Architecture	5	1	Written examination (90 minutes)	English	1/24
6.2	Quantum Information Science	5	1	Written examination (90 minutes)	English	1/24
	2nd Semester					
1	Safety Critical Computer Systems (for students starting in the summer semester)	5	1	Oral examination (min. 15, max. 45 minutes)	English	1/24
7	Mathematics Update (for students starting in the winter semester)	5	1	Written examination (90 minutes)	English	1/24
8	Compulsory Elective Subjects III					
8.1	Advanced Distributed Systems	5	1	Written examination (90 minutes)	English	1/24

No.	Module Title	ECTS CP	Duration [Sem.]	Assessment type	Language	weighting
8.2	Advanced Testing Methods	5	1	Written examination (90 minutes)	English	1/24
9	Advanced IT-Security	5	1	Written examination (120 minutes)	English	1/24
10	Compulsory Elective Subjects IV					
10.1	Human Machine Interaction	5	1	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)	English	1/24
10.2	Smart Sensor Network Systems	5	1	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)	English	1/24
10.3	Advanced Data Structures and Algorithms	5	1	Project report (submission period 6 weeks) with presentation (at least 10, at most 20 minutes)	English	1/24
11	Data Mining Methods	5	1	Written computer-based examination (90 minutes) ER*	English	1/24
12	Compulsory Elective Subjects V					
12.1	System Theory and Modeling	5	1	Written examination (90 minutes)	English	1/24
12.2	Transaction Management	5	1	Written examination (90 minutes)	English	1/24
12.3	Learning from Data	5	1	Oral examination (min. 15, max. 45 minutes)	English	1/24
3rd Semester						
13	Compulsory Elective Subjects VI					
13.1	Multivariate Data Analysis	5	1	Written computer-based examination (90 minutes) ER*	English	1/24
13.2	Simulation Methods	5	1	Written examination (90 minutes) ER*	English	1/24
13.3	Artificial Intelligence	5	1	Written examination (90 minutes)	English	1/24
14	Compulsory Elective Subjects VII					

No.	Module Title	ECTS CP	Duration [Sem.]	Assessment type	Language	weighting
14.1	Standards and Certification	5	1	Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes)	English	1/24
14.2	Current Topics in High Integrity Systems	5	1	Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes)	English	1/24
14.3	Internet of Things	5	1	Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes)	English	1/24
15	Formal Specification and Verification	5	1	Written computer-based examination (90 minutes) ER*	English	1/24
16	Compulsory Elective Subjects VIII					
16.1	Selected Subjects in Current Web Engineering	5	1	Written examination (90 minutes)	English	1/24
16.2	Mobile Systems and Applications	5	1	Written examination (90 minutes)	English	1/24
16.3	Cloud Computing	5	1	Written assignment (processing time 6 weeks) with presentation (min. 15, max. 30 minutes)	English	1/24
17	High Integrity Systems Project	10	1	Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes)	English	1/12
4th semester						
18	Master Thesis with Colloquium	30	1	Master-Thesis (processing time 20 weeks) with Colloquium (min. 30, max. 60 minutes)	English	1/4

*ER = Module Examination Requirements

4. Module Descriptions

Module 1

Module title	Safety Critical Computer Systems
Module number	1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Barrierefreie Systeme (M.Sc.)
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Bachelor-level modules Mathematics, Programming Languages, Software Engineering
Module prerequisites	none
Module examination requirements	none
Module examination	Written examination (90 minutes)

Learning outcomes and skills	Upon completion of this course, the student is able to: • distinguish between reliability and safety, • critically read accident reports, • perform a hazard analysis for 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. • respect cultural and social aspects of project work in international R&D teams • present • apply team leading skills • practice scientific literature research and handling • apply time and project management skills
Module contents	Safety Critical Computer Systems – Lectures Safety Critical Computer Systems - Exercises
Module teaching methods	Lectures: Interactive Teaching Exercises: Teamwork in small development groups
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 1.1

Unit title	Safety Critical Computer Systems – Lectures
Code	
Module title	Safety Critical Computer Systems
Unit 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 methods	Interactive Lectures
Semester periods (hours) per week	2
Workload (h)	70 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Matthias Wagner

Recommended reading	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.
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 1.2

Unit title	Safety Critical Computer Systems - Exercises
Code	
Module title	Safety Critical Computer Systems
Unit 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 methods	Teamwork in small development groups, exercise tasks
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30 h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Matthias Wagner
Recommended reading	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.
Assessment type and form	
Assessment grading	
Comments	n/a

Module 2

Module title	Advanced Formal Modeling
Module number	2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other Computer Science Master programs
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Basic knowledge of propositional and first order logic
Module prerequisites	None
Module examination requirements	Lab exercises with written assignment (processing time 80 hours)
Module examination	Written computer-based examination (90 minutes)

Learning outcomes and skills	The module provides an insight into the theory and practice of for- mal modeling. It focuses on the role of logic in deductive software verification and in knowledge representation. In this module the stu- dents learn how to model formal properties of software and prove its correctness using logic. They learn how to model knowledge using logic-based knowledge representation formalisms. Main goals are: • Understanding the use of different logics in formal modeling. • Understanding the logical foundations of formal methods and logic-based knowledge representation formalisms. • Ability to prove correctness of simple code fragments. • Ability to formalize and reason using logic. • Obtaining practical skills in using a theorem prover and a for- mal verification tool. • Understanding the limitations of logic. • Non specialist competencies: Scientific working style
Module contents	Advanced Formal Modeling – Lectures Advanced Formal Modeling – Exercises
Module teaching methods	Lectures, Exercises
Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Barįs Sertkaya
Comments	n/a

Unit 2.1

Unit title	Advanced Formal Modeling - Lectures
Code	
Module title	Advanced Formal Modeling
Unit contents	The lectures cover the theoretical aspects and provide examples mainly on the following topics: • Mathematical and logical foundations – Sets, relations and functions - Propositional Logic, First Order Logic • Verification of sequential programs • Verification of Java programs using Java Modeling Language • Logic-based knowledge representation and reasoning • Description Logics, Web Ontology Language OWL
Teaching methods	Lecture
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Ruth Schorr, Prof. Dr. Barjs Sertkaya

Recommended reading	• Logic in Computer Science Modelling and Reasoning about Systems. M. Huth, M. Ryan. 2004 (2nd Edition). ISBN: 9780521543101 • Understanding Formal Methods. J.-F. Monin. Springer, 2003. ISBN-10: 1852332476 • Deductive Software Verification The KeY Book. From Theory to Practice. W. Ahrendt, B. Beckert, R. Bubel, R. Hhnle, P. H. Schmitt, M. Ulbrich. ISBN 978-3-319-49811-9 • The Description Logic Handbook: Theory, Implementation, and Applications. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. F. Patel-Schneider.Cambridge University Press, 2003.
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 2.2

Unit title	Advanced Formal Modeling - Exercises
Code	
Module title	Advanced Formal Modeling
Unit contents	The exercises • deepen the knowledge for the theoretical foundations of modeling and proving with logic, and • help to gain practical experience with theorem provers and formal verification tools.
Teaching methods	Exercises
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Ruth Schorr, Prof. Dr. Barjs Sertkaya
Recommended reading	See unit Advanced Formal Modeling – Lectures
Assessment type and form	Lab exercises with written assignment (processing time 80 hours)
Assessment grading	Pass/fail
Comments	n/a

Module 3

Module title	Introductory Data Analysis
Module number	3
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	n/a
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/150 h
Recommended previous knowledge	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.
Module prerequisites	None
Module examination requirements	• Lab exercises with written assignment (processing time 74 hours) • written exposé (processing time 6 hours)
Module examination	written computer-based examination, 90 minutes

Learning outcomes and skills	• 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 understand 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 • Scientific work style
Module contents	Introductory Data Analysis – Lectures Introductory Data Analysis – Exercises
Module teaching methods	Lectures using multimedia presentation techniques Exercises on PC using spreadsheets and statistical software tool
Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Andreas Orth
Comments	n/a

Unit 3.1

Unit title	Introductory Data Analysis - Lectures
Code	
Module title	Introductory Data Analysis
Unit contents	• Descriptive statistics (characteristics and plots, univariate and bivariate methods) • Probability 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-parametric tools • Common Errors (how not to proceed)
Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth
Recommended reading	• Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley. • Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them)
Assessment type and form	
Assessment grading	

Comments	n/a

Unit 3.2

Unit title	Introductory Data Analysis - Exercises
Code	
Module title	Introductory Data Analysis
Unit 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 • lessons learned session after written exposé • exam preparation session for the Module examination
Teaching methods	Use of PC in computer pool to solve problems
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth
Recommended reading	• Montgomery, Runger: Applied Statistics and Probability for Engineers, Wiley. • Good P.I.; Hardin J. W.: Common Errors in Statistics (and How to Avoid them)
Assessment type and form	• Lab exercises with written assignment • written exposé (processing time 6 hours))
Assessment grading	pass/fail
Comments	n/a

Module 4.1

Module title	Advanced Real Time Systems
Module number	4.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Barrierefreie Systeme (M.Sc.), Computer Science Master “Allgemeine Informatik”
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	Knowledge in operating systems, programming languages, automata theory, formal languages, hardware architectures, modelling, and simulation
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)

Learning outcomes and skills	• Extending the basic knowledge of real-time systems by reading a typical real-time research-paper • Transfering the knowledge into theoretical model solving a concrete problem • Transfering the theoretical model into a working software • Validating the software • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Cultural and social aspects of project work in international R&D teams • Scientific literature research and handling • Writing a paper • Time and project management skills • Project documentation
Module contents	Advanced Real Time Systems-Project
Module teaching methods	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.
Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Matthias Deegener, Prof. Dr. Karsten Weronek
Comments	

Unit 4.1.1

Unit title	Advanced Real Time Systems - Project
Code	
Module title	Advanced Real Time Systems
Unit 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 methods	R&D project with small groups (4 students max.)
Semester periods (hours) per week	4
Workload (h)	150h
Class hours	60h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	80h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Karsten Weronek
Recommended reading	Current literature, e.g. research papers, will be announced at the begin of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Module 4.2

Module title	Machine Learning
Module number	4.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Barrierefreie Systeme (M.Sc.), Computer Science Master “Allgemeine Informatik”
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	Good knowledge of mathematics (Algebra, calculus, statistics)
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes duration

Learning outcomes and skills	**Professional expertise:** The students acquire a basic understanding of a standard approach in the field of machine learning, the general terminology and the principles of the field. In addition they get a practical understanding of the relevant mathematical, statistical and numerical aspects of the field with respect to applications. The students are capable • to apply this knowledge independently for problems in different application area, and • to implement it on an appropriate software platform. **Personal.expertise:** The students are self-reliantly able to elaborate on complex theoretical models and to follow the state-of-the-art of the research field. They are capable to present worked-out solution strategies as well to experts of the field as to members of other disciplines. Because of the complexities of the requirements they are able to employ an effizient and evolutionary approach keeping the target in sight.
Module contents	Machine Learning - Lectures Machine Learning - Exercises
Module teaching methods	Lectures with exercises
Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Ute Bauer-Wersing
Comments	Course is supported by e-learning platform.

Unit 4.2.1

Unit title	Machine Learning- Lectures
Code	
Module title	Machine Learning
Unit contents	The lectures introduce the basic and advanced views, problems, methods and techniques of machine learning. Topics: • Introduction and historic development of machine learning • Biological motivation with the brain as example • Formalization and modeling with neural networks • Problem overview (Classification, regression, clustering) • Supervised and unsupervised learning • Perceptron, linear separability, relation to Bayes-classification • Linear Nets, online PCA and dimension reduction, feature selection • Multilayer perceptron • Error-back propagation, optimization • Structural concepts (Capability for generalization, estimation of approximation errors, MLE) • Concepts of the gradient-based learning algorithm, online vs. batch learning, cross validation • Deep Learning, convolutional networks etc.
Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2

Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	40h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Ute Bauer-Wersing
Recommended reading	C. M. Bishop (2006). Pattern Recognition and Machine Learning. Springer Verlag. T. Hastie, R. Tibshiranie, J. Friedmann (2001). The Elements of Statistical Learning. Springer Verlag. E. Alpaydin (2008). Maschinelles Lernen. Oldenbourg Wissenschaftsverlag. S. Sharma (1996). Applied Multivariate Techniques. Wiley & Sons. At semester start relevant current research literature will be named.
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Unit 4.2.2

Unit title	Machine Learning - Exercises
Code	
Module title	Machine Learning
Unit contents	Lecture topics, i.e. theories and concepts, are complemented by practical exercises using an appropriate software.
Teaching methods	Use of PC in computer pool to solve problems, exercise tasks
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Ute Bauer-Wersing
Recommended reading	see unit lectures
Assessment type and form	
Assessment grading	
Comments	n/a

Module 5

Module title	Implementation of DBMS
Module number	5
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programs
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	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
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination , 90 minutes

Learning outcomes and skills	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. • Working in teams • Communication in international teams
Module contents	Implementation of DBMS – lectures Implementation of DBMS - exercises
Module teaching methods	Interactive lectures Teamwork in lab exercises
Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Christian Rich
Comments	n/a

Unit 5.1

Unit title	Implementation of DBMS - Lectures
Code	
Module title	Implementation of DBMS
Unit 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 methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10h
Total time of individual study (h)	30h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	• 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.

Assessment type and form	
Assessment grading	
Comments	n/a

Unit 5.2

Unit title	Implementation of DBMS - Exercises
Code	
Module title	Implementation of DBMS
Unit contents	Exercise tasks based on module contents
Teaching methods	Exercises, team work
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	See Unit Implementation of DBMS – Lectures
Assessment type and form	
Assessment grading	
Comments	n/a

Module 6.1

Module title	Pattern Oriented Software Architecture
Module number	6.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programs
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	elective module
ECTS (cp) / Workload (h)	5 CP/ 150h.
Recommended previous knowledge	Good knowledge in principles and procedures of software engineering, programming skills in object-oriented programming languages
Module prerequisites	None
Module examination requirements	
Module examination	Written examination (90 minutes)
Learning outcomes and skills	• 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. • team work: Students acquire the ability to work with others toward a same goal, participating actively, sharing responsibility and rewards, and contributing to the capability of the teamwork. • communication in international teams
Module contents	Pattern Oriented Software Architecture – Lectures Pattern Oriented Software Architecture – Exercises
Module teaching methods	Interactive lectures Lab exercises in teamwork

Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. Eicke Godehardt
Comments	n/a

Unit 6.1.1

Unit title	Pattern Oriented Software Architecture - Lectures
Code	
Module title	Pattern Oriented Software Architecture
Unit 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 methods	Interactive Group Lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer
Recommended reading	• E. Gamma et. Al: Design Patterns Addison-Wesley, 1998 • Buschmann et al: Pattern Oriented Software Architecture. Addison Wesley 1996
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 6.1.2

Unit title	Pattern Oriented Software Architecture - Exercises
Code	
Module title	Pattern Oriented Software Architecture
Unit contents	• Lab exercises pertaining to the contents described in the unit Pattern Oriented Software Architecture – lectures • Lessons learned session after solved problems
Teaching methods	Teamwork in small R&D groups, exercise tasks
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Eicke Godehardt, Prof. Dr. Jörg Schäfer
Recommended reading	See Unit Pattern-Oriented Software Architecture - Lectures
Assessment type and form	
Assessment grading	
Comments	

Module 6.2

Module title	Quantum Information Science
Module number	6.2
Module code	Module code
Study program	High-Integrity Systems (M.Sc.)
Module usability	Other computer science Master programs
Module duration	One Semester
Recommended semester	1st /2nd semester
Module type	Elective module
ECTS (cp) / Workload (h)	5 cp / 150 h
Recommended previous knowledge	Analysis and Algebra with complex numbers helpful
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination (90 minutes)
Learning outcomes and skills	After having successful passed of this module, students are able • to describe the fundamental concepts and methods in Quantum Information Science, • to name the different building blocks, to explain their duties and functionality and to describe them in a formal mathematical manner, • to elucidate selected algorithms for this computational model, • to analyze, modify and apply existing quantum algorithms by using mathematical and computation foundation, • to constitute current forms of relevant quantum-systems
Module contents	Quantum Information Science Lectures, Quantum Information Science Exercises
Module teaching methods	On-Site Lecture, with up to 60% parts may be online, exercises, homework, reading assignments

Module language	English
Module availability	Winter semester
Module coordination	Prof. Dr. rer. nat. Karsten Weronek
Comments	

Unit 6.2.1

Unit title	Quantum Information Science Lectures
Code	
Module title	Quantum Information science
Unit contents	This unit gives an introduction into the behavior and mathematical description of two-status quantum systems and its practical and theoretical usage for Quantum Information Science. This comprises (for example but not limited to): • Quantum Physics Fundamentals (Wave functions, Schrödinger's Equation, measurement of quantum states (observables)) • Mathematical description of single and multiple two state system (complex Analysis and Algebra, wave functions, Dirac notation, Bloch sphere) • Relevant properties of Quantum Systems (e.g. non-determinism, Quantum Entanglement, superposition principle, coherence, Non-Cloning Theorem, Quantum Teleportation) • No-Communication-Theorem (Bell's equations, non-locality) • Building blocks (CBits and QuBits, Quantum gates and circuits) • Examples of quantum computer algorithms (e.g. Shor, Quantum-Fourier-Transformation, quantum search, Grover, Deutsch/Josza etc.) • Applications in Quantum Computing (e.g. quantum cryptology) • Simulation/Application of quantum computers (e.g. using MS-Q#, IBM-Quantum SDK, Goolge Cirk) Optional topics are e.g.: • Fundamentals of computability (e.g. Church-Turing Thesis, BQP) • Actual availability of quantum computers • Selected items of Quantum Information Theory • Post-Quantum cryptology
Semester periods (hours) per week	2
Workload (h)	70
Class hours	30
Total time of examination incl. preparation (h)	10h

Total time of individual study (h)	30
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Jörg Schäfer, Prof. Dr. rer. nat. Matthias Wagner, Prof. Dr. rer. nat. Karsten Weronek
Recommended reading	• N. D. Mermin, Quantum Computer Science, An Introduction; New York, Cambridge University Press, 2007, reprinted 2014. • M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information; New York: Cambridge University Press, 2000, 4th printing 2013 • R. J. Lipton and K. W. Regan, Quantum Algorithms via Linear Algebra, A Primer; London: The MIT Press, 2014
Assessment type and form of	
Assessment grading	
Comments	A comprehensive and up-to-date bibliography will be provided during the lecture.

Unit 6.2.2

Unit title	Quantum Information Science Exercises
Code	
Module title	Quantum Information Science
Unit contents	• Calculation of computing-relevant quantum systems • Design and analysis of quantum algorithms and circuits • Simulation/application of quantum computers (e.g. using MS-Q#, IBM-Quantum SDK, Goolge Cirk)
Teaching methods	Exercises, homework and reading assignments
Semester periods (hours) per week	2
Workload (h)	80
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Jörg Schäfer, Prof. Dr. rer. nat. Matthias Wagner, Prof. Dr. rer. nat. Karsten Weronek
Recommended reading	See unit lectures
Assessment type and form of	
Assessment grading	
Comments	

Module 7

Module title	Mathematics Update
Module number	7
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programs
Module duration	One semester
Recommended semester	2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	Undergraduate level of mathematics
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes
Learning outcomes and skills	• 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
Module contents	Mathematics Update – Lectures Mathematics Update – Exercises
Module teaching methods	Interactive Lectures Exercises with teamwork in small groups
Module language	English.
Module availability	summer semester (for students starting in the winter semester) winter semester (for students starting in the summer semester)
Module coordination	Prof. Dr. Doina Logofatu

Comments	n/a

Unit 7.1

Unit title	Mathematics Update - Lectures
Code	
Module title	Mathematics Update
Unit contents	• Linear Algebra • Geometry • Discrete Mathematics • Calculus • Scientific Computing
Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Jörg Schäfer, Prof. Dr. Doina Logofatu, Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr
Recommended reading	• 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
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 7.2

Unit title	Mathematics Update – Exercises
Code	
Module title	Mathematics Update
Unit contents	• Computer Exercises pertaining to the contents described in the unit Mathematics Update – Lectures • Lessons learned session
Teaching methods	Teamwork in small R&D groups, exercise tasks
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Jörg Schäfer, Prof. Dr. Doina Logofatu, Prof. Dr. Egbert Falkenberg, Prof. Dr. Ruth Schorr
Recommended reading	see Unit Mathematics Update – Lectures
Assessment type and form	None
Assessment grading	None
Comments	n/a

Module 8.1

Module title	Advanced Distributed Systems
Module number	8.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5CP/ 150h

Recommended previous knowledge	• 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

Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes
Learning outcomes and skills	• Understand 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.
Module contents	Advanced Distributed Systems – Lectures Advanced Distributed Systems – Exercises

Module teaching methods	Interactive Group Lecturing Teamwork exercises in small groups
Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Justus Klingemann
Comments	n/a

Unit 8.1.1

Unit title	Advanced Distributed Systems – Lectures
Code	
Module title	Advanced Distributed Systems
Unit contents	• Properties of distributed systems • Time and synchronization • Distributed algorithms • Middleware for distributed systems • Consistency and replication
Teaching methods	Interactive Group Lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Recommended reading	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
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 8.1.2

Unit title	Advanced Distributed Systems - Exercises
Code	
Module title	Advanced Distributed Systems
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit Distributed Systems – lectures • Practical teamwork on real world problems • Lessons learned session after group work
Teaching methods	Teamwork in small software development groups
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Recommended reading	See Unit Advanced Distributed Systems – Lectures
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Module 8.2

Module title	Advanced Testing Methods
Module number	8.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programs
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory Elective Module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Extended knowledge in software engineering, very good programming skills in procedural and object-oriented programming languages
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes
Learning outcomes and skills	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. This module facilitates communication structures used in business like Wikis and Discussion boards to show challenges working in global teams.
Module contents	Advanced Testing Methods – Lectures Advanced Testing Methods – Exercises

Module teaching methods	Interactive group lecturing Teamwork in small groups
Module language	English
Module availability	Summer semester
Module coordination	Dr. Torsten Schönfelder, Deutsche Lufthansa
Comments	n/a

Unit 8.2.1

Unit title	Advanced Testing Methods – Lectures
Code	
Module title	Advanced Testing Methods – Lectures
Unit 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 methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Dr. Torsten Schönfelder, Deutsche Lufthansa
Recommended reading	Current Software Engineering literature announced at the beginning of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 8.2.2

Unit title	Advanced Testing Methods – Exercises
Code	
Module title	Advanced Testing Methods
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit Advanced Testing Methods – lectures • practical teamwork on real world problems • Lessons learned session after group work
Teaching methods	Teamwork in small software development groups
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Dr. Torsten Schönfelder, Deutsche Lufthansa
Recommended reading	Current Software Engineering literature announced at the beginning of the semester
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Module 9

Module title	Advanced IT-Security
Module number	9
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	• 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
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 120 minutes
Learning outcomes and skills	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. • to communicate in international teams
Module contents	Advanced IT-Security – Lectures Advanced IT-Security – Exercises
Module teaching methods	Interactive Group Lecturing

	Teamwork Exercises in small groups
Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Martin Kappes
Comments	n/a

Unit 9.1

Unit title	Advanced IT-Security – Lectures
Code	
Module title	Advanced IT-Security
Unit 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 methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Martin Kappes
Recommended reading	Current literature to be announced at the beginning of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 9.2

Unit title	Advanced IT-Security - Exercises
Code	
Module title	Advanced IT-Security
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit Advanced IT-Security – lectures • practical teamwork on real world problems • Lessons learned session after group work
Teaching methods	Teamwork in small software development groups, exercises
Semester periods (hours) per week	2
Workload (h)	80 h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50 h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Martin Kappes
Recommended reading	See unit Advanced IT-Security lectures
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Module 10.1

Module title	Human Machine Interface
Module number	10.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes, particularly Barrierefreie Systeme (M.Sc.)
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Elective module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Knowledge in software and systems engineering, knowledge in programming
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)

Learning outcomes and skills	• 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 the user, how to derive requirements for an interface, and how to test the usability of an interface • Transferring the gained knowledge into a theoretical model solving a concrete problem • Transferring the theoretical model into a working demonstrator • Working with others toward a same goal, and contributing to the capability of the teamwork. • Creating an atmosphere of respect, helpfulness and cooperation. • Validating the demonstrator with the aid of usability tests • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper
Module contents	Human Machine Interaction - Project
Module teaching methods	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
Module language	English
Module availability	Each Summer semester
Module coordination	Prof. Dr. Gerd Doeben-Henisch
Comments	n/a

Unit 10.1.1

Unit title	Human Machine Interaction – Project
Code	
Module title	Human Machine Interaction
Unit 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 methods	R&D project with small groups (4 students max.)
Semester periods (hours) per week	4
Workload (h)	150 h
Class hours	60h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	80h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Gerd Döben-Henisch
Recommended reading	Current literature, e.g. research papers, will be announced at the beginning of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Module 10.2

Module title	Smart Sensor Network Systems
Module number	10.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes, particularly Barrierefreie Systeme (M.Sc.), Allgemeine Informatik (M.Sc.)
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Elective module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	• Knowledge in software and systems engineering, • C/C++ - programming • Numerical analysis
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with presentation (min. 10, max. 20 minutes)

Learning outcomes and skills	• 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 • Cultural and social aspects of project work in international R&D teams • Presentation skills • Team leading skills • Documentation • Writing a scientific paper
Module contents	Smart Sensor Network Systems – Project
Module teaching methods	Project
Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 10.2.1

Unit title	Smart Sensor Network Systems – Project
Code	
Module title	Smart Sensor Network Systems
Unit 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 methods	R&D project with small groups (4 students max.)
Semester periods (hours) per week	4
Workload (h)	150h
Class hours	60h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	80h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Matthias Wagner
Recommended reading	Holger Karl, 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
Assessment type and form	
Assessment grading	
Comments	n/a

Module 10.3

Module title	Advanced Data Structures and Algorithms
Module number	10.3
Study programme	High Integrity Systems (M.Sc.)
Module usability	Allgemeine Informatik (M.Sc.)
Module duration	One semester
Recommended semester	1st/2nd semester
Module type	Compulsory elective module
ECTS-Points (CP) / Workload (h)	5 CP / 150 hours
Prerequisites for participation in the module and the module examination	None
Prerequisites for the acquisition of credit points: a. preliminary examination b. Module examination	a. Computer-based exercises with written documentation (processing time 35 h) b. Project report (submission period 6 weeks) with presentation (at least 10, at most 20 minutes)
Learning outcomes and skills	Upon completion of the module students are able to: - analyze the complexity of data structures. - recognize, choose, and appropriate use advanced data structures. - analyze the complexity of algorithms. - deal with selected advanced algorithms, especially from the area of nature- and bio-inspired algorithms. - compare the efficiency/optimality of different algorithms. - implement and compare different approaches for a given real application. - deliver practical oriented solutions. - perform statistical tests. Training for non-specialist competencies. Students: - practice scientific project management. - communicate and work in team. - research and write scientific text. - present their results in a scientific colloquium.
Module contents	Advanced Data Structures and Algorithms – Lecture Advanced Data Structures and Algorithms – Exercise
Module teaching methods	Interactive group lecturing with exercises
Module language	English
Module availability	Each summer semester
Module coordination	Prof. Dr. Doina Logofatu
Comments	None

Unit 10.3.1

Unit title	Advanced Data Structures and Algorithms – Lecture
Code	
Module title	Advanced Data Structures and Algorithms
Unit contents	The course will cover selected topics form the following areas. The depth of coverage might vary. Advanced Data Structures and their Complexity: - Dynamic Graphs - Double-Ended Priority Queues - Binomial, Pairing and Fibonacci Heaps - Leftist, AVL, Red-Black, B+, B*, Splay, Priority Search, Segment, Multidimensional, Quad Trees - Compressed Binary, High Order Tries Advanced Algorithms and related innovative Applications: - Genetic Algorithms - Ant Colony and Swarm Intelligence Optimization - Genetic Programming - Quantum Computing and Quantum Annealing - Distributed/Parallel Evolutionary Algorithms - Other Nature- and Bio-Inspired Methods - Hybrid Evolutionary Algorithms - Memetic and Cloud Computing - Fuzzy Logic - Chaos Theory - Machine Learning The emphasis must be on examples for systems and applications in practice.
Unit teaching methods	Interactive group lecturing
Semester periods (hours) per week	2 SWS
Unit workload (h)	70h
Class hours (h)	30h
Total time of examination incl. preparation (h)	10h
Total time of individual study (h)	30h
Total time of practical training (h)	0h
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu

Recommended reading	- David E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison. Wesley, 1989 - John R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems), A. Bradford Book, 1992 - Dario Floreano, Claudio Matiussi, Bio-Inspired Intelligence: Theories, Methods, and Technologies, The MIT Press, 2008 Steven H. Strogatz, Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering, Westview Press, 2014 - El-Ghazali Talbi, Metaheuristics: From Design to Implementation, Wiley, 2009 Vijay V. Vazirani, Approximation Algorithms, Springer, 2004 Stephen Boyd, Lieven Vandenbeghe, Convex Optimization, Cambridge University Press, 2004 - Wan Fokkink, Distributed Algorithms: An intuitive Approach, The MIT Press, 2013 - Calos A. Varela, Programming Distributed Computing Systems: A Foundational Approach, The MIT Press, 2013
Assessment type and form of the unit	None
Assessment grading of the unit	None
Unit comments	None

Unit 10.3.2

Unit title	Advanced Data Structures and Algorithms – Exercises
Code	
Module title	Advanced Data Structures and Algorithms
Unit contents	Realization, analysation, and enhancement of applications and systems
Unit teaching methods	Teamwork in small groups (at most 4 students)
Semester periods (hours) per week	2 SWS
Unit workload (h)	80h
Class hours (h)	30h
Total time of examination incl. preparation (h)	0h
Total time of individual study (h)	50h
Total time of practical training (h)	0h
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu
Recommended reading	see Unit Advanced Data Structures and Algorithms – Lecture
Assessment type and form of the unit	Computer-based exercises with written documentation (processing time 35h)
Assessment grading of the unit	pass/fail
Unit comments	None

Module 11

Module title	Data Mining Methods
Module number	11
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	Semester 2
Module type	Compulsory module
ECTS (cp) / Workload (h)	5CP/ 150 h
Recommended previous knowledge	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
Module prerequisites	None
Module examination requirements	Software Exercises with documentation (processing time 80 hours)
Module examination	Written computer-based examination, 90 minutes

Learning outcomes and skills	• 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 understand 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 • Communication in international teams
Module contents	Data Mining Methods - Lectures Data Mining Methods - Exercises
Module teaching methods	Lectures using multimedia presentation techniques Exercises with a PC and statistical programming language in Computer pool to solve problems
Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Andreas Orth
Comments	n/a

Unit 11.1

Unit title	Data Mining Methods – Lectures
Code	
Module title	Data Mining Methods

Unit 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
	  ○ linear modelling – GLM, GLIM, mixed effects modeling, variable selection methods - ,
	  ○ Methods for Classification – prototype-methods, k-nearest neighbour classifiers, Linear Discriminant Analysis, logistic regression, separating hyperplanes, support vector machines etc.
	  ○ Additive Models, Trees, Boosting Methods, Additive Trees
	  ○ Neural nets
	  ○ Unsupervised Learning
	  ○ Variance Estimation and Validation methods (selection out of bootstrapping, jackknifing, cross-validation, Bayesian methods, EM-algorithm, MCMC)

Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth

Recommended reading	• 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)
Assessment type and form of	
Assessment grading	
Comments	n/a

Unit 11.2

Unit title	Data Mining Methods – Exercises
Code	
Module title	Data Mining Methods
Unit 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 methods	Using PC and statistical programming language in Computer pool to solve problems
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth
Recommended reading	• 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)
Assessment type and form	Software Exercises with documentation (processing time 80 hours)

Assessment grading	Pass/fail
Comments	n/a

Module 12.1

Module title	System Theory and Modeling
Module number	12.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5CP/ 150h
Recommended previous knowledge	Good knowledge in discrete mathematics, analysis, numerical methods
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination,90 minutes
Learning outcomes and skills	• 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. • Scientific literature research and handling
Module contents	System Theory and Modeling – Lectures System Theory and Modeling - Exercises
Module teaching methods	Interactive lectures using multimedia presentation techniques Exercises: Teamwork
Module language	English.
Module availability	Each summer semester

Module coordination	Prof. Dr. Doina Logofatu
Comments	n/a

Unit 12.1.1

Unit title	System Theory and Modeling – Lectures
Code	
Module title	System Theory and Modeling

Unit 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 methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner

Recommended reading	• 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 • 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 Current literature, e.g. journal papers, conference proceedings etc., will be announced at the beginning of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 12.1.2

Unit title	System Theory and Modeling – Exercises
Code	
Module title	System Theory and Modeling
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit System Theory and Modeling – lectures • Lessons learned session after group work
Teaching methods	Teamwork in small R&D groups
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner
Recommended reading	See Unit System Theory and Modeling – Lectures
Assessment type and form	
Assessment grading	
Comments	n/a

Module 12.2

Module title	Transaction Management
Module number	12.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	• Good knowledge in the use of database systems and programming with higher programming languages This corresponds to the following bachelor module: • Databases Programming
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes
Learning outcomes and skills	• 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. • Scientific literature research and handling
Module contents	Transaction Management – Lectures Transaction Management – Exercises
Module teaching methods	Interactive lectures Exercises: Teamwork in R&D-groups

Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Justus Klingemann
Comments	n/a

Unit 12.2.1

Unit title	Transaction Management – Lectures
Code	
Module title	Transaction Management
Unit contents	• Concept of transactions • Theory of serialization • Concurrency Control • Recovery • Distributed transactions • Extended transaction models
Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	• 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
Assessment type and form	
Assessment grading	

Comments	n/a

Unit 12.2.2

Unit title	Transaction Management – Exercises
Code	
Module title	Transaction Management
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit Transaction Management – lectures • practical teamwork on real world problems • Lessons learned session after group work
Teaching methods	Teamwork in small R&D groups
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Christian Rich
Recommended reading	See Unit Transaction Management – Lectures
Assessment type and form	
Assessment grading	
Comments	n/a

Module 12.3

Module title	Learning from Data
Module number	12.3
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	1st /2nd semester
Module type	Elective module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	• Good knowledge in mathematics (Linear algebra, calculus of one and multiple variables, statistics) • Good programming capabilities
Module prerequisites	None
Module examination requirements	Computer-based exercises with written assignment and presentation (min.10, max.20 minutes), processing time 80 hours
Module examination	Oral examination (min. 15 minutes, max. 45 minutes)
Learning outcomes and skills	Professional expertise: The students gain a principal understanding of the mathematical and epistemological basics of statistical learning theory and machine learning. They are capable to independently apply their knowledge in various problem fields, e.g. robotics, big data etc. In addition they know the most important application fields of statistical learning theory and are able to assess the ethical and societal dimensions of applications. The students have the opportunity to test their findings in a prototype on a relevant platform and write a scientific paper. Personal expertise: The students are capable to independently work out complex theoretical models and to follow the state-of-the-art of current research. They are capable to write scientific publications and to present the results of their research for experts and lay persons alike.

Module contents	Learning for Data – Lectures Learning from Data – Exercises
Module teaching methods	Interactive lectures Exercises: Seminar
Module language	English
Module availability	Summer semester
Module coordination	Prof. Dr. Jörg Schäfer
Comments	n/a

Unit 12.3.1

Unit title	Learning from Data– Lectures
Code	
Module title	Learning from Data
Unit contents	• Problem space • Is statistical or machine learning possible? (Outlook Vapnik-Chervonenkis theory, VC Dimension and Limit) • Statistical/Machine learning: Supervised, unsupervised, reinforcement learning, classification, regression • Probabilistic and non-probabilistic methods • Different models: o Linear model o Kernel methods o Support Vector Machines • Bayes methods o Gaussian processes o MCMC and particle filters • Neural networks o Genetic algorithms • Model selection o Overfitting and regularization o Bias-Variance tradeoff, error and noise • Training, testing, validation o Curse of dimensionality • Vapnik-Chervonenki theory, Hoeffding's lemma, VC dimension and VC inequality • Structural Risk Minimization (SRM), Occham's Razor • Overview of selected application fields • consequences of statistical learning theory • Philosophic implications (Are data more important than theories? Where are the limits of statistical learning theory? • Ethical implications for society, principal problems of algorithmic decisions (Information privacy, individual freedom, checks-and balances of citizens, corporations, states etc.) • Lectures are accompanied by practical exercises with respect to theories and concepts, supported by the use of appropriate software.
Teaching methods	Lectures

Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Prof. Dr. Jörg Schäfer
Recommended reading	Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. Learning From Data. AMLBook, 2012. Nello Cristianini and John Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, 1 edition, 2000. Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning: data mining, inference and prediction. Springer, 2 edition, 2008. G. James, D. Witten, T. Hastie, and R. Tibshirani. An Introduction to Statistical Learning: with Applications in R. Springer Texts in Statistics. Springer New York, 2014. Vladimir N. Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 12.3.2

Unit title	Learning from Data – Exercises and Seminar
Code	
Module title	Learning from Data
Unit contents	• Elaboration on individual aspects from the lectures in exercises and seminar contributions
Teaching methods	Seminar with accompanying exercises
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Ute Bauer-Wersing, Prof. Dr. Thomas Gabel, Prof. Dr. Jörg Schäfer
Recommended reading	See Unit Learning from Data – Lectures
Assessment type and form	Computer-based exercises with written assignment and presentation (min.10, max.20 minutes), processing time 80 hours
Assessment grading	Pass/fail
Comments	n/a

Module 13.1

Module title	Multivariate Data Analysis
Module number	13.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Understanding of univariate and bivariate methods as taught in module Introductory Data Analysis, Experience in applying statistical methods to real world data
Module prerequisites	None
Module examination requirements	Laboratory exercises with written assignment (total processing time 74 hours) written exposé (processing time 6 hours)
Module examination	written computer-based examination, 90 minutes
Learning outcomes and skills	• 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 • Scientific literature research and handling

Module contents	Multivariate Data Analysis – Lectures Multivariate Data Analysis - Exercises
Module teaching methods	Lectures using multimedia presentation techniques Group work
Module language	English
Module availability	each semester
Module coordination	Prof. Dr. Andreas Orth
Comments	n/a

Unit 13.1.1

Unit title	Multivariate Data Analysis – Lectures
Code	
Module title	Multivariate Data Analysis
Unit 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 methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth
Recommended reading	• 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)
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 13.1.2

Unit title	Multivariate Data Analysis – Exercises
Code	
Module title	Multivariate Data Analysis
Unit 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 • Lessons learned session after group work • exam preparation session for the Module examination
Teaching methods	Using PCs in the Computer pool to solve problems Group work
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Andersson, Prof. Dr. Falkenberg, Prof. Dr. Orth
Recommended reading	• 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
Assessment type and form	Laboratory exercises with written assignment (total processing time 74 hours) written exposé (processing time 6 hours)
Assessment grading	None

Comments	n/a

Module 13.2

Module title	Simulation Methods
Module number	13.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science Master programmes
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5CP/ 150 h
Recommended previous knowledge	• Good knowledge in discrete mathematics, calculus, numerical methods, • contents of module System Theory and Modeling
Module prerequisites	None
Module examination requirements	Lab exercises (total processing time 80 hours)
Module examination	Written examination, 90 minutes
Learning outcomes and skills	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 contents	Simulation Methods – Lectures Simulation Methods - Exercises
Module teaching methods	Interactive lectures using multimedia presentation techniques Exercises: Teamwork

Module language	English
Module availability	Each Summer semester
Module coordination	Prof. Dr. Doina Logofatu
Comments	n/a

Unit 13.2.1

Unit title	Simulation Methods – Lectures
Code	
Module title	Simulation Methods
Unit contents	• Methods Approximation techniques Types and categories Software tools Numerical methods Visualization • Validation Simulation and Measurement • Applications
Teaching methods	Interactive group learning
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	10h
Total time of individual study (h)	30h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner
Recommended reading	• 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 Current literature, e.g. journal papers, conference proceedings etc., will be announced at the beginning of the semester

Assessment type and form	
Assessment grading	
Comments	n/a

Unit 13.2.2

Unit title	Simulation Methods – Exercises
Code	
Module title	Simulation Methods
Unit contents	• Lab exercises with software tools pertaining to the contents described in the unit Simulation Methods – lectures • Lessons learned session after group work
Teaching methods	Teamwork in small R&D groups
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Doina Logofatu, Prof. Dr. Matthias Wagner
Recommended reading	See Unit Simulation Methods – Lectures
Assessment type and form	Lab exercises (processing time 80 hours)
Assessment grading	Pass/fail
Comments	n/a

Module 13.3

Module title	Artificial Intelligence
Module number	13.3
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Barrierefreie Systeme (M.Sc.), Computer Science Master “Allgemeine Informatik”
Module duration	One semester
Recommended semester	3rd Semester
Module type	Compulsory elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	None
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes

Learning outcomes and skills	**Professional expertise:** The students acquire a basic understanding of problems, methods and techniques for the development and assessment of Artificial Intelligence (AI) systems. This includes knowledge of classical and state-of-the-art theoretical models of symbolic artificial intelligence and of software systems for the implementation of learned methods and algorithms. The students are capable • to independently design AI-based solutions for problems in different application areas, • to comprehend the functioning of selected learning algorithms and to independently implement those, and • to employ common software solutions for deliberate implementation of their design ideas. **Personal.expertise:** The students are capable to present and defend worked-out solution strategies as well to experts of the field as to members of other disciplines. They understand the basics of scientific work, master literature research and scientific presentation methods and use the e=learning platform.
Module contents	Unit 1: Artificial Intelligence - Lectures Unit 2: Artificial Intelligence - Exercises
Module teaching methods	Lectures with exercises
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Thomas Gabel
Comments	Course is supported by e-learning platform.

Unit 13.3.1

Unit title	Artificial Intelligence - Lectures
Code	
Module title	Artificial Intelligence
Unit contents	The first part of the lectures introduces the principles views, problems, methods and techniques of artificial intelligence. Topics: • Introduction and historical development of AI • The concept of agents in AI • Problem solution and search • Planning, logic, inference • Symbolic learning • Visualization and analysis of uncertain knowledge Based on this foundation the second part deepens special, selected topics of AI, f.i. game theory, deep learning, multi-agent systems, case-based inference, evolutionary algorithms, optimization.
Teaching methods	Interactive group lecturing
Semester periods (hours) per week	2
Workload (h)	70h
Class hours	30h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	40h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Thomas Gabel, Prof. Dr. Ute Bauer-Wersing

Recommended reading	S. Russell, P. Norvig: Artificial Intelligence: A Modern Approach At semester start and during the semester relevant current research literature will be named.
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Unit 13.3.2

Unit title	Artificial Intelligence - Exercises
Code	
Module title	Artificial Intelligence
Unit contents	Lecture topics, i.e. theories and concepts, are complemented by theoretical, application oriented and practical exercises using an appropriate software.
Teaching methods	Use of PC in computer pool to solve problems, exercise tasks
Semester periods (hours) per week	2
Workload (h)	80h
Class hours	30h
Total time of examination incl. preparation (h)	n/a
Total time of individual study (h)	50h
Total time of practical training (h)	n/a
Unit language	English
Lecturer	Prof. Dr. Thomas Gabel, Prof. Dr. Ute Bauer-Wersing
Recommended reading	see unit lectures
Assessment type and form	
Assessment grading	n/a
Comments	n/a

Module 14.1

Module title	Standards and Certification
Module number	14.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	n/a
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	None
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with oral presentation (min. 20 , max. 30 minutes)
Learning outcomes and skills	Students will be 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. • learn to search for, read, summarize and cite scientific literature on a large scale; • read and interpret national and international standards; • write a report as a scientific paper; • give a scientific talk.
Module contents	Certification and Standards – Seminar

Module teaching methods	Seminar
Module language	English
Module availability	Alternating with modules 14.2, 14.3, summer and winter semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 14.1.1

Unit title	Standards and Certification – Seminar
Code	
Module title	Standards and Certification
Unit contents	• International standards for safety critical computer systems • Overview over certification in various fields of application • Commonalities and differences of various standards
Teaching methods	Seminar
Semester periods (hours) per week	2
Workload (h)	150 h
Class hours	36
Total time of examination incl. preparation (h)	
Total time of individual study (h)	114 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Matthias Wagner
Recommended reading	Debra Herrmann Software Safety and Reliability IEEE Computer Society 1999 Current literature, e.g. journal papers, conference proceedings etc., will be announced at the beginning of the semester
Assessment type and form	
Assessment grading	
Comments	n/a

Module 14.2

Module title	Current Topics in High Integrity Systems
Module number	14.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	n/a
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsary Elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	None
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with oral presentation (min. 20 , max. 30 minutes)
Learning outcomes and skills	• 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. • 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 contents	Unit Current Topics in High Integrity Systems – Seminar

Module teaching methods	Seminar
Module language	English.
Module availability	Alternating with modules 14.1, 14.3, summer and winter semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 14.2.1

Unit title	Current Topics in High Integrity Systems
Code	
Module title	Current Topics in High Integrity Systems
Unit contents	Current topics in Computer Science with respect to the analysis, design, development and maintenance of High-Integrity Systems
Teaching methods	Seminar
Semester periods (hours) per week	2
Workload (h)	150h
Class hours	36h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	114h
Total time of practical training (h)	
Unit language	English
Lecturer	All professors of the Master's program High Integrity Systems
Recommended reading	Current research literature
Assessment type and form	
Assessment grading	
Comments	n/a

Module 14.3

Module title	Internet of Things
Module number	14.3
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	n/a
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective module
ECTS (cp) / Workload (h)	5 CP/ 150h
Recommended previous knowledge	None
Module prerequisites	None
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with oral presentation (min. 20 , max. 30 minutes)
Learning outcomes and skills	• 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. • 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 contents	Internet of Things – Seminar
Module teaching methods	Seminar

Module language	English
Module availability	annually
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 14.3.1

Unit title	Internet of Things – Seminar
Code	
Module title	Internet of Things
Unit 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 methods	Seminar
Semester periods (hours) per week	2
Workload (h)	150 h
Class hours	36
Total time of examination incl. preparation (h)	
Total time of individual study (h)	114 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Deegener, Prof. Dr. Matthias Wagner
Recommended reading	
Assessment type and form	
Assessment grading	
Comments	n/a

Module 15

Module title	Formal Specification and Verification
Module number	15
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	other computer science Master programmes
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	• Basic knowledge of propositional and predicate logic • Basic knowledge of algorithm design and analysis • Basic knowledge of automata theory
Module prerequisites	None
Module examination requirements	Laboratory exercises with written assignment (processing time: 80 hours)
Module examination	Written computer-based examination (90 minutes)
Learning outcomes and skills	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. Obtaining practical skills in using a Model Checking Tool . Understanding the limitations of model checking.

	Non specialist competencies: Communication in international teams
Module contents	Formal Specification and Verification – Lectures Formal Specification and Verification – Exercises
Module teaching methods	Lectures Exercises
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Ruth Schorr
Comments	n/a

Unit 15.1

Unit title	Formal Specification and Verification – Lectures
Code	
Module title	Formal Specification and Verification
Unit 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
Teaching methods	Lectures
Semester periods (hours) per week	2
Workload (h)	70 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Ruth Schorr
Recommended reading	Baier, Christel and Katoen, Joost-Pieter: Principles of Model Checking MIT Press, 2008. Ben-Ari, Mordechai: *Principles of the Spin Model Checker.* Springer, 2008.
Assessment type and form	
Assessment grading	

Comments	n/a

Unit 15.2

Unit title	Formal Specification and Verification –Exercises
Code	
Module title	Formal Specification and Verification
Unit contents	The exercises • deepen the knowledge for the theoretical foundations of model checking and • provide an introduction to model checkers, i.e. SPIN, NuSMV or nuXmv, with practical exercises
Teaching methods	Exercises
Semester periods (hours) per week	2
Workload (h)	80 h
Class hours	30 h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Ruth Schorr
Recommended reading	see Unit Formal Specification and Verification – Lectures
Assessment type and form	Laboratory exercises with written assignment (processing time: 80 hours)
Assessment grading	Pass/fail
Comments	

Module 16.1

Module title	Selected Subjects in Current Web Engineering
Module number	16.1
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science and engineering Master programmes
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective Module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Basic understanding of Distributed Systems and basic command of a high level OO language (such as Java) for practical exercises
Module prerequisites	None
Module examination requirements	None
Module examination	Written examination, 90 minutes

Learning outcomes and skills	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. At the end of the course the students will • 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 • be 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)
Module contents	Selected Subjects in Current Web Engineering - Lectures Selected Subjects in Current Web Engineering - Exercises
Module teaching methods	Lectures Exercises
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Jörg Schäfer
Comments	n/a

Unit 16.1.1

Unit title	Selected Subjects in Current Web Engineering – Lectures
Code	
Module title	Selected Subjects in Current Web Engineering

Unit contents	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 methods	Lectures
Semester periods (hours) per week	2
Workload (h)	70 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Recommended reading	• 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.
Assessment type and form	
Assessment grading	

Comments	n/a

Unit 16.1.2

Unit title	Selected Subjects in Current Web Engineering – Exercises
Code	
Module title	Selected Subjects in Current Web Engineering
Unit contents	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 methods	Exercises
Semester periods (hours) per week	2
Workload (h)	80 h
Class hours	30 h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer
Recommended reading	See Unit Selected Subjects in Current Web Engineering – Lectures
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Module 16.2

Module title	Mobile Systems and Applications
Module number	16.2
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Other computer science and engineering Master programmes
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective Module
ECTS (cp) / Workload (h)	5 CP/150 h
Recommended previous knowledge	Good knowledge in programming with C and Java, foundations of computer networks
Module prerequisites	None
Module examination requirements	None
Module examination	written examination, 90 minutes
Learning outcomes and skills	• 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
Module contents	Mobile Systems and Applications – Lectures Mobile Systems and Applications – Exercises
Module teaching methods	Lectures Exercises
Module language	English

Module availability	Each semester
Module coordination	Prof. Dr. Jörg Schäfer
Comments	n/a

Unit 16.2.1

Unit title	Mobile Systems and Applications – Lectures
Code	
Module title	Mobile Systems and Applications
Unit contents	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 methods	Lectures
Semester periods (hours) per week	2
Workload (h)	70 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner

Recommended reading	• 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.
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 16.2.2

Unit title	Mobile Systems and Applications – Exercises
Code	
Module title	Mobile Systems and Applications
Unit contents	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 methods	Exercises
Semester periods (hours) per week	2
Workload (h)	80 h
Class hours	30 h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Matthias Deegener, Prof. Dr. Justus Klingemann, Prof. Dr. Jörg Schäfer, Prof. Dr. Matthias Wagner
Recommended reading	See Unit Mobile Systems and Applications – Lecture
Assessment type and form	n/a
Assessment grading	n/a
Comments	n/a

Module 16.3

Module title	Cloud Computing
Module number	16.3
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	Cloud Computing – Lectures Cloud Computing – Exercises
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory Elective Module
ECTS (cp) / Workload (h)	5 CP/ 150 h
Recommended previous knowledge	Good knowledge in software engineering, computer networks, databases and distributed applications and one high-level programming language.
Module prerequisites	None
Module examination requirements	None
Module examination	Written assignment (processing time 6 weeks) with presentation (min. 15, max. 30 minutes)
Learning outcomes and skills	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.
Module contents	Cloud Computing – Lectures Cloud Computing – Exercises

Module teaching methods	Lectures Exercises
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Christian Baun
Comments	n/a

Unit 16.3.1

Unit title	Cloud Computing – Lectures
Code	
Module title	Cloud Computing
Unit contents	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 methods	Lectures
Semester periods (hours) per week	2
Workload (h)	70 h
Class hours	30 h
Total time of examination incl. preparation (h)	10 h
Total time of individual study (h)	30 h
Total time of practical training (h)	
Unit language	English
Lecturer	Prof. Dr. Christian Baun, Prof. Dr. Martin Kappes, Prof. Dr. Jörg Schäfer

Recommended reading	Christian Baun, Marcel Kunze, Jens Nimis and Stefan Tai, Cloud Computing: Web-Based Dynamic IT Services, Springer (2011). Tom White, Hadoop: The Definitive Guide (Englisch) Taschenbuch, O'Reilly, 2015. Thomas Erl, Zaigham Mahmood, Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, Prentice Hall, 2013. Nick Antonopoulos and Lee Gillam: Cloud Computing: Principles, Systems and Applications, Springer, 2010. Current literature recommendations will be given at the semester start.
Assessment type and form	
Assessment grading	
Comments	n/a

Unit 16.3.2

Unit title	Cloud Computing – Exercises
Code	
Module title	Cloud Computing
Unit contents	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. In addition, simple prototypes will be developed. Henceforth, the students will receive continuous feedback, which will support the educational objectives.
Teaching methods	Exercises
Semester periods (hours) per week	2
Workload (h)	80 h
Class hours	30 h
Total time of examination incl. preparation (h)	
Total time of individual study (h)	50 h
Total time of practical training (h)	
Unit language	English
Lecturer	See lecturers of Unit Cloud Computing – Lectures
Recommended reading	n/a
Assessment type and form	
Assessment grading	
Comments	n/a

Module 17

Module title	High Integrity Systems Project
Module number	17
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	
Module duration	One semester
Recommended semester	3rd semester
Module type	Compulsory module
ECTS (cp) / Workload (h)	10 LP/ 300 h
Recommended previous knowledge	Excellent knowledge in software and systems engineering, Very good programming skills Mathematics
Module prerequisites	Successfully completed modules of 40 CP
Module examination requirements	None
Module examination	Project (submission period 8 weeks) with presentation (min. 20, max. 30 minutes)

Learning outcomes and skills	• 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. • 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	High Integrity Systems – Project
Module teaching methods	Project
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 17.1

Unit title	High Integrity Systems Project
Code	
Module title	High Integrity Systems Project
Unit 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 methods	R&D project with small groups
Semester periods (hours) per week	4
Workload (h)	300 h
Class hours	60 h
Total time of examination incl. preparation (h)	20 h
Total time of individual study (h)	220 h
Total time of practical training (h)	
Unit language	English
Lecturer	All professors of the High Integrity Systems Master programme
Recommended reading	Current research literature
Assessment type and form	
Assessment grading	
Comments	n/a

Module 18

Module title	Master Thesis with Colloquium
Module number	18
Module code	
Study program	High Integrity Systems (M.Sc.)
Module usability	
Module duration	20 weeks
Recommended semester	Semester 4
Module type	Compulsory module
ECTS (cp) / Workload (h)	30 CP/ 900 h
Recommended previous knowledge	Sound acquisition of the contents and competences of the first three semesters of the study programme
Module prerequisites	Successful completion of all modules of the first three semesters of the study programme
Module examination requirements	Successful completion of all modules of the first three semesters of the study programme
Module examination	Master´s Thesis (processing time: 20 weeks) with Colloquium (min. 30, max. 60 minutes)
Learning outcomes and skills	• 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. • 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	Research and Development project

Module teaching methods	Thesis and colloquium
Module language	English
Module availability	Each semester
Module coordination	Prof. Dr. Matthias Wagner
Comments	n/a

Unit 18.1

Unit title	Master Thesis with Colloquium
Code	
Module title	Master Thesis with Colloquium
Unit 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. • Presentation of thesis' results
Teaching methods	R&D project leading to the Master Thesis under supervision of 2 professors Research for the thesis may be done at • Frankfurt University of Applied Sciences • Research groups • External research institutes • Companies Coaching by the supervisors
Semester periods (hours) per week	
Workload (h)	900 h
Class hours	
Total time of examination incl. preparation (h)	20 h
Total time of individual study (h)	880 h
Total time of practical training (h)	
Unit language	English
Lecturer	All professors of the High Integrity Master study program
Recommended reading	Current research literature
Assessment type and form	Master's Thesis (processing time 20 weeks) with Colloquium (min.30, max. 60 minutes)
Assessment grading	1-4= pass, 5= fail
Comments	n/a