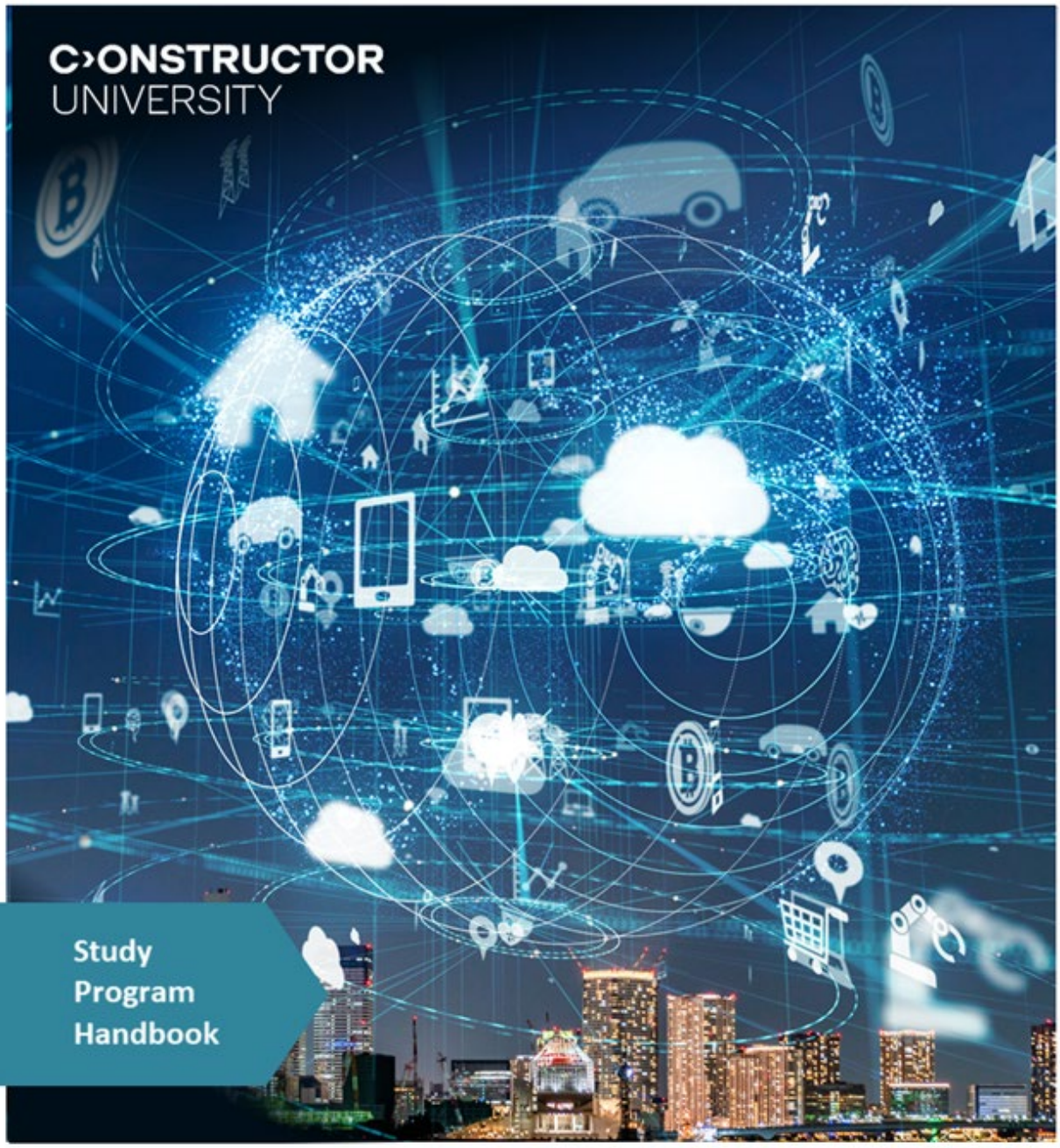


Computer Science and Software Engineering

Master of Science

Subject-specific Examination Regulations for Computer Science and Software Engineering (CSSE)

The subject-specific examination regulations for CSSE are defined by this program handbook and are valid only in combination with the General Examination Regulations for Master degree programs ("General Master Policies").

Upon graduation students in this program will receive a Master of Science (MSc) degree with a scope of 120 ECTS credit points (CP) (for specifics see chapter 3 of this handbook).

Valid for all students starting their studies in Fall 2026

Version	**Valid as of**	**Decision**	**Details**
Fall 2026- V1.1		Nov 19, 2026	Editorial: Schematic Plan
Fall 2026-V1	Sep 01, 2026		
		May 26, 2022	Originally approved by the Academic Senate

Contents

1 Program Overview 1

1.1 Concept 1

1.2 Qualification Aims 2

1.2.1 Educational Aims 2

1.2.2 Intended Learning Outcomes 3

1.3 Target Audience 4

1.4 Career Options 4

1.5 Admission Requirements 5

1.6 More information and contacts 6

2 The Curriculum 7

2.1 The Curriculum at a Glance 7

2.2 Schematic Study Scheme 8

2.3 Study and Examination Plan 9

2.4 Technical CORE Modules 12

2.4.1 Software Engineering Modules 12

2.4.2 Cybersecurity Modules 12

2.4.3 Artificial Intelligence Modules 13

2.4.4 Breakthrough Area Modules 13

2.5 Management Modules 13

2.6 Leadership / Academic Skills Modules 14

2.7 Project, Capstone Project & Master Thesis 14

3 Computer Science and Software Engineering Graduate Program Regulations 15

3.1 Scope of these Regulations 15

3.2 Examination Concept 15

3.3 Degree 15

3.4 Graduation Requirements 16

3.5 Other Program-specific Policies & Practices 17

4 Module Descriptions 18

4.1 Software Engineering Modules 18

4.1.1 Software Construction, Architecture and Engineering 18

4.1.2 Quality Engineering 21

4.1.3 Advances in Software Engineering 23

4.1.4 Advanced AI-Native Software Engineering with Constructor Fabric 26

4.1.5 Advanced Databases 29

4.1.6 Artificial Intelligence in Software Engineering 32
4.2 Cybersecurity Modules 35
4.2.1 Cryptography 35
4.2.2 System Security 38
4.2.3 Network Security 41
4.2.4 Cybercriminology 44
4.3 Artificial Intelligence Modules 46
4.3.1 Deep Learning 46
4.3.2 Intelligent Autonomous Systems 49
4.3.3 Symbolic Artificial Intelligence 51
4.3.4 Text Analysis and Natural Language Processing 53
4.3.5 Data Analytics 55
4.3.6 Machine Learning 58
4.4 Breakthrough Area Modules 61
4.4.1 Quantum Informatics 61
4.5 Management Modules 64
4.5.1 Agile Product Development & Design 64
4.5.2 Product Innovation & Marketing 67
4.5.3 Transformational Change Management 69
4.6 Leadership/ Academic Skills Modules 71
4.6.1 Entrepreneurship and Intrapreneurship 71
4.6.2 Communication & Presentation Skills for Executives 74
4.6.3 Organizational Behavior 77
4.6.4 Academic Writing Skills/Intercultural Training 79
4.6.5 Agile Leadership and Strategic Management 82
4.6.6 Customer-centric Mindset and Agile Delivery Management 84
4.7 Research Project, Capstone Project & Master Thesis 86
4.7.1 Research Project 86
4.7.2 Capstone Project I 88
4.7.3 Capstone Project II 91
4.7.4 Capstone Project III 94
4.7.5 Research Lab I 97
4.7.6 Research Lab II 99
4.7.7 Internship 101
4.7.8 Master Thesis CSSE 104
5 Appendix 107

5.1 Intended Learning Outcomes Assessment Matrix 107

1 Program Overview

1.1 Concept

Computer science is one of the most impactful and lively research disciplines as digitalization is the backbone of industry and society as well as academia. There is enormous progress that is driven especially by artificial intelligence including machine learning and cyber-physical systems, but there are also new challenges, e.g., dealing with malicious uses and threads, i.e., the need for cybersecurity. Software, hence, software engineering, and more generally digital companies play a key role in this domain. Leading companies have a crucial need for a new breed of digital experts. The complexity of software and digitalization demands a new generation of experts with crosscutting technical management and leadership skills. At the same time, disruption is often driven by small start-ups that require not only technical skills in developing software, but the management and entrepreneurial skills to make their mark.

The Master of Science (MSc) in Computer Science and Software Engineering (CSSE) at Constructor University is a consecutive master program that complements a broad spectrum of research-oriented technological education with essential management and leadership skills to educate the future technology leaders in research and industry. To prepare students for this role of technology leaders in research and industry, it offers a solid training in Software Engineering regarding development and management, as well as technical core courses in three subject areas that are presently at the utmost importance

- Software Engineering,
- Cybersecurity, and
- Artificial Intelligence.

These offers mirror the research activities at Constructor University and the faculty involved. Thereby, excellent teaching competence is guaranteed and hands-on experiences from the forefront of the state of the art in research and industry are provided. In addition, breakthrough applications such as Quantum Informatics will be covered.

As a consecutive Master program, the Computer Science and Software Engineering program is targeted at strong graduates of undergraduate programs related to the computer science disciplines. Core knowledge in the field is a mandatory requirement to enter the MSc CSSE program. Upon graduating, students will have obtained a portfolio of skills in highly relevant areas of computer science, namely Software Engineering, Artificial Intelligence, and Cybersecurity. Students will develop their creative and constructive abilities to produce, develop, and evaluate solutions for technical challenges. They will acquire knowledge about the state of the art in a selected subject area, and they learn the skills necessary to approach, develop, and document small independent projects dealing with the latest state of the art in research, (industrial) applications and even start-ups.

To strengthen the educational concept, the program will make use of flipped classroom teaching that will enable, wherever applicable, a student-centric and hands-on experience. Team-based work on software projects and beyond further profits from agile development concepts. Together with state-of-the-art equipment in soft- and hardware, the program allows seamless collaboration among students and instructors and naturally adapts to conditions that may derive from pandemic emergencies.

Overall, by completing the master study, students will acquire the core expertise of digital leaders, with a solid technological backbone developed along three complementary areas, with additional core management and leadership skills that characterize the educational journey. They will acquire the essential soft skills for an active digital technology leadership in the contemporary global and multiethnic society, thanks to the international environment that characterizes Constructor University. Overall, this education will enable graduates to enter research via Ph.D. programs and to succeed in the job market in high profile roles.

1.2 Qualification Aims

1.2.1 Educational Aims

Digitalization is the backbone of industry and society. Software and digital companies play a key role. Leading companies have a crucial need for a new breed of digital experts. The complexity of software and digitalization demands a new generation of experts with deep technological knowledge but also crosscutting technical management and leadership skills.

The Computer Science and Software Engineering program aims to provide an in-depth understanding of the essential aspects of designing, maintaining, and analyzing digital systems. Students will acquire the skills necessary to apply methods and tools to successfully and responsibly engineer software. The program seeks to expand the participant's competencies and capabilities in the subject areas Software Engineering, Cybersecurity and Artificial Intelligence, which play a dominant role in industries and research. To leverage technology excellence, one of these areas is selected by each student as the main specialization. The curriculum further complements this subject specific education by teaching modern cross-disciplinary leadership and management competencies to tomorrow's digital leaders.

Students are introduced to practical and research-oriented work through practical educational offers in a capstone project, an internship, or research labs and an elective research project. These offers as well as the final thesis are supported by frequent individual feedback sessions and personal guidance. This facilitates and quickens the students' career development and helps them to become valuable assets in industries and research within a short period of time.

Constructor University programs are offered in a highly intercultural environment. Students acquire intercultural competence as part of their education through everyday group work, class participation, and extracurricular activities. In this way, students gain practical intercultural competencies and build their confidence in an English-speaking work and study environment. Presenting a strong, confident appearance and communicating effectively in various cultural contexts are among the core abilities of internationally successful executives in any business area.

To summarize, graduates of Computer Science and Software Engineering will have obtained the following competences and skills:

- Subject-matter competence in a Computer Science specialization

  Graduates have acquired in-depth knowledge of one of the fields of software engineering, cybersecurity, or artificial intelligence. In doing so, they are not only able to define and interpret the doctrine of the field, but have also developed a detailed and critical understanding at the cutting edge of knowledge in the field.

- Computer Science and Software Engineering Competency

In general, graduates have a broadened and deepened knowledge in their formal, algorithmic, and applied competencies in Computer Science. This enables them to develop independent ideas as digital experts. Responding to the massive demand in industry and following the increasing interest in research software, graduates have also acquired broader knowledge in software engineering, enabling them to solve practical and scientific problems in the field.

- Learning, transfer and research skills

Graduates are able to learn new methodologies by means of theoretically underpinned approaches, lifelong and trend independent. This enables them to apply problem solutions in new and unfamiliar situations. They integrate learned skills in complex and multidisciplinary contexts, as it is increasingly necessary in industry and research. In particular, graduates are able to formulate research questions, select appropriate methods, and document and interpret research results.

- Management and Leadership Skills

Recognizing the ever-increasing need for management and leadership skills in business, industry and research, graduates have a broad and integrated knowledge and understanding of the fundamentals of management and leadership. Their knowledge corresponds to the standard literature in the field. In particular, they are able to solve related problems in the field of computer science and software engineering with professional plausibility.

- Teamwork and communication skills

Graduates are proficient in the specialized exchange of ideas in a group setting with the goal of collaborative development of digital software or hardware systems. This is reinforced by effective and reflective practice of communication and collaboration on both academic and non-academic topics.

- Personal and Professional Competence

Graduates will be able to develop a professional profile both in and out of academia and make, justify and reflect decisions based on theoretical and professional knowledge. They can critically examine their own behavior and assess social consequences. In doing so, they act appropriately to the situation, also in an international environment, and further develop their professional actions.

1.2.2 Intended Learning Outcomes

By the end of this program, students will be able to:

1. critically assess and creatively apply technological possibilities and innovations in the fields of computer science and software engineering;
2. critically assess and apply software engineering methodologies considering real life situations, organizations and industries;
3. use, adapt und improve modern artificial intelligence techniques related to data, planning and applications;
4. design, implement and exploit methods in cryptography and security related fields;
5. apply cross-disciplinary management methodologies to solve academic and professional problems;

6. critically assess and integrate a consistent tool set of leadership abilities into a professional work environment;
7. plan, conduct and document small research projects in the context of computer science and software engineering;
8. independently research, document and present a scientific topic with appropriate language skills;
9. use scientific methods as appropriate in the field of Computer Science and Software Engineering such as defining research questions, justifying methods, collecting, assessing and interpreting relevant information, and drawing scientifically founded conclusions that consider social, scientific and ethical insights;
10. develop and advance solutions to problems and arguments in their subject area and defend these in discussions with specialists and non-specialists;
11. engage ethically with academic, professional and wider communities and to actively contribute to a sustainable future, reflecting and respecting different views;
12. take responsibility for their own learning, personal and professional development and role in society, evaluating critical feedback and self-analysis;
13. apply their knowledge and understanding to a professional context;
14. take on responsibility in a diverse team;
15. adhere to and defend ethical, scientific and professional standards.

1.3 Target Audience

The program is designed for students of different geographical and cultural backgrounds. The program addresses graduates of computer science and closely related undergraduate programs who would like to focus or deepen their knowledge in the field of Computer Science and Software Engineering. Candidates who are dedicated to and interested in gaining theoretical and application-oriented knowledge in the fields of Software Engineering, Cybersecurity and Artificial Intelligence are particularly addressed by the program.

Prior to admission, applicants have already completed their first degree in Computer Science or a closely related subject.

The program prepares students for key roles in the IT industry and for entering research in the subject fields. Part of this is the additional educational offer in the program that exposes students to management and leadership courses. The program also prepares students to develop their own start-ups. The program's educational approach supports exchange and discussion within the student community. Hence, the willingness to interact, to appreciate different teaching and learning formats, to accept challenges, and to develop professionally during study are important requirements for successful participation in the program.

1.4 Career Options

Computers are ubiquitous and essential for the functioning of our civilization. At the same time, their continuously growing complexity poses substantial challenges on all levels, from technology to society at large.

Computer Science researchers contribute new insights into concepts and their realization in a wide spectrum of disciplines. IT practitioners work in literally all areas of industry, business, government,

finance, energy, education, healthcare, aerospace, and many more. This work can be a core IT task, such as being an administrator responsible for some system, or applied work done in collaboration with domain experts. IT experts maintain databases and networks, set up web-based information services, deal with Big Data, increase cyber security, program robots, devise artificial intelligence models, ensure software quality, and provide consultancy, to name but a few.

Finally, Computer Science and Software Engineering graduates are desperately needed all over the world. So, graduates will not have to extensively search for a job, but the employers will seek for the graduates, allowing them to select from a rich choice of highly paid offers.

Constructor University's Career Services Center (CSC) and Alumni Office will help students in their career development. The Career Service Center (CSC) helps students in their career development. It provides students with high-quality training and coaching in CV creation, cover letter formulation, interview preparation, effective presenting, business etiquette, and employer research as well as in many other aspects, thus helping students identify and follow up on rewarding careers after graduating from Constructor University. For further information, please contact the Career Service Center (CSC) (https://constructor.university/student-life/career-services). Furthermore, the Alumni Office helps students establish a long-lasting and global network which is useful when exploring job options in academia, industry, and elsewhere. In addition, the broad industry network of the Constructor Group provides excellent access to leading technology enterprises.

1.5 Admission Requirements

The Computer Science and Software Engineering graduate program requires students to have completed an undergraduate program in computer science, software engineering, information technology or another discipline with at least 60 ECTS of computer science-related topics (such as mathematics, programming, design, software architecture). Students not fulfilling the main admission criterion of at least 60 ECTS of computer-science related topics can still be conditionally admitted based on a case-by-case basis decision. Part of the condition for admission can be the requirement to take further relevant courses out of the computer science related undergraduate programs at Constructor University. Regularly, these will be courses from the CHOICE or CORE area from these programs or mathematics courses from the Constructor track. Applicants need to prove a strong interest in the contents of the study program in a motivation letter.

Social commitment as well as extracurricular and voluntary activities during undergraduate studies, e.g. university service, clubs, varsity, social work, will be considered. Work experience is not a prerequisite.

Additionally, participants should possess elevated analytical, problem solving and verbal communication skills which must be substantiated in recommendation letters.

Study at Constructor University takes place in a highly intercultural environment. It is therefore necessary to be willing to join such a multicultural-international community and work together with students and faculty across various fields of interest at Constructor University.

Admission to Constructor University is selective and based on a candidate's university achievements, recommendations and self-presentation. Students admitted to Constructor University demonstrate exceptional academic achievements, intellectual creativity, and the desire and motivation to make a difference in the world.

The following documents need to be submitted with the application:

- Letter of motivation
- Curriculum vitae (CV)
- Official or certified copies of university transcripts
- Bachelor's degree certificate or equivalent
- Language proficiency test results (minimum score of 90 (TOEFL), 6.5 (IELTS) or 110 (Duolingo)).
- Copy of Passport
- Letter of recommendation (optional).

Formal admission requirements are subject to higher education law and are outlined in the Admission and Enrollment Policy of Constructor University.

For more detailed information about the admission visit:
Application Information | Constructor University

1.6 More information and contacts

For more information on the study program please contact the Study Program Coordinator:

Prof. Dr. Jürgen Schönwälder

Professor of Computer Science

Email: jschoenwaelder@constructor.university

or visit our program website: https://constructor.university/programs/graduate-education/computer-science-software-engineering

For more information on Student Services please visit:

https://constructor.university/student-life/student-services

2 The Curriculum

2.1 The Curriculum at a Glance

The curriculum of the Computer Science and Software Engineering master program is divided into four semesters and takes two years to complete. Each semester is composed of a mixture of core technical content, project/seminar work, management and leadership education and academic skills work, leading to a master's thesis that can cover academic research, industrial applications or developments towards a start-up.

The modules are grouped into several domains, as outlined in the Schematic Study Plan (see Figure 1).

To graduate, students take out of these modules a total of 120 ECTS with

- Technical CORE modules: 45 ECTS,
- Management modules: 15 ECTS,
- Leadership / Academic Skills modules: 15 ECTS,
- Industry / Research / Start-up Orientation modules: 15 ECTS,
- Master Thesis module: 30 ECTS.

Detailed module descriptions in their latest version are available in the catalogue on CampusNet.

2.2 Schematic Study Scheme

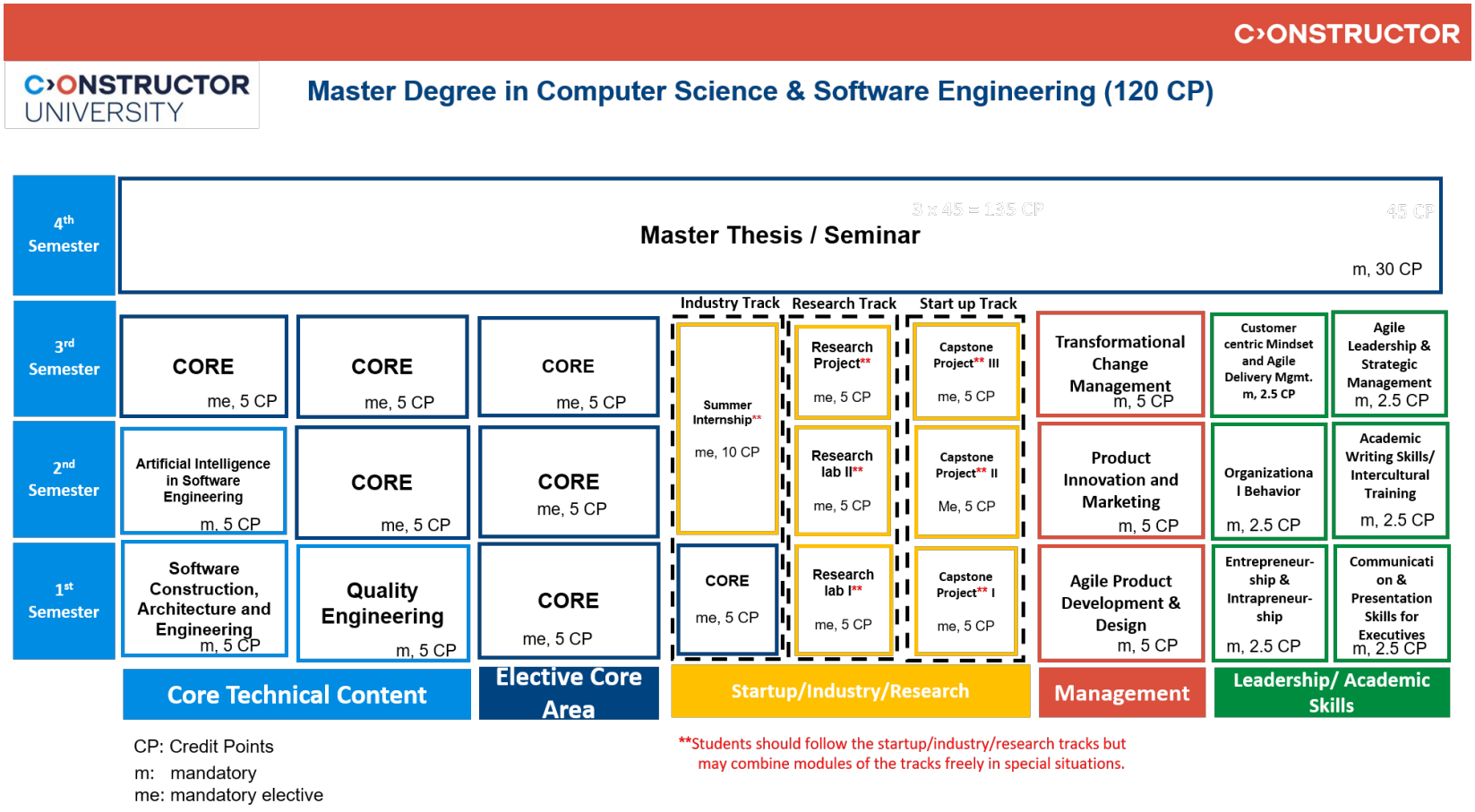


Figure 1: Schematic Study Scheme

2.3 Study and Examination Plan

MSc Degree in Computer Science and Software Engineering Matriculation Fall 2026							
Module Code	**Program-Specific Modules**	**Type**	**Assessment**	**Period[1]**	**Status[2]**	**Semester**	**CP**
Semester 1							**25**
	CORE modules						**15**
MCSSE-SE-01	**Module: Software Construction, Architecture and Engineering**				**m**	**1**	**5**
MCSSE-SE-01-A	Software Construction, Architecture and Engineering	Lecture	Written examination	Examination period			
MCSSE-SE-02	**Module: Quality Engineering**				**m**	**1**	**5**
MCSSE-SE-02	Quality Engineering	Lecture	Portfolio Assessment	During semester			
	Further CORE modules				**me**	**1**	**5**
	- students choose 1 module from those listed below						
	Startup/ Research [3]						**5**
	Students choose a preferred Track						
MCSSE-CAP-01	**Module: Capstone Project 1**				**me**	**1**	**5**
MCSSE-CAP-01	Capstone Project 1	Project	Project Assessment	During semester			
MCSSE-RL-01	**Research Lab I**				**me**	**1**	**5**
MCSSE-RL-01	Research Lab I	Lab	Lab Report	During semester			
	Management Modules						**5**
MCSSE-MGT-01	**Module: Agile Product Development & Design**				**m**	**1**	**5**
MCSSE-MGT-01	Agile Product Development & Design	Lecture	Presentation	Examination period			
	Leadership / Academic Skills Modules						**5**
MCSSE-LAS-01	**Module: Entrepreneurship & Intrapreneurship**				**m**	**1**	**2.5**
MCSSE-LAS-01	Entrepreneurship & Intrapreneurship	Lecture	Presentations	During semester			
MDE-CAR-01	**Module: Communication & Presentation Skills for Executives**				**m**	**1**	**2.5**
MDE-CAR-01	Communication & Presentation Skills for Executives	Seminar	Oral Presentation	During semester			
Semester 2							**30**
	CORE modules						**15**
MCSSE-SE-06	**Module: Artificial Intelligence in Software Engineering**				**m**	**2**	**5**
MCSSE-SE-06-A	Artificial Intelligence in Software Engineering	Lecture	Oral examination	Examination period			
	Further CORE modules				**me**	**2**	**10**
	- students choose 2 modules from those listed below						
	Startup/ Research track [3]						**5**
	Students choose a preferred Track						
MCSSE-CAP-02	**Module: Capstone Project 2**				**me**	**2**	**5**
MCSSE-CAP-02	Capstone Project 2	Project	Project Assessment	During semester			
MCSSE-RL-02	**Module: Research Lab II**				**me**	**2**	**5**
MCSSE-RL-02	Research Lab II	Lab	Lab Report	During semester			
	Management Modules						**5**
MCSSE-MGT-02	**Module: Product Innovation & Marketing**				**m**	**2**	**5**
MCSSE-MGT-02	Product Innovation & Marketing	Lecture	Presentation	During semester			
	Leadership / Academic Skills Modules						**5**
MCSSE-LAS-02	**Module: Organizational Behavior**				**m**	**2**	**2,5**
MCSSE-LAS-02	Organizational Behavior	Lecture	Presentations	During semester			
MDE-CAR-02	**Module: Academic Writing Skills / Intercultural Training**				**m**	**2**	**2.5**
MDE-CAR-02	Academic Writing Skills / Intercultural Training	Seminar	Term Paper	Examination period			

Semester 3							**30**
	CORE modules						**15**
	Further CORE modules				**me**	**3**	**15**
	- students choose 3 modules from those listed below. One CORE module can be replaced by the Research Project module.						
	Startup/ Research / /Industry track [3]						**5**
	Students choose a preferred Track						
MCSSE-CAP-03	**Module: Capstone Project 2**				**me**	**3**	**5**
MCSSE-CAP-03	Capstone Project 3	Project	Project Assessment	During semester			
MCSSE-RP-01	**Research Project**				**me**	**3**	**5**
MCSSE-RP-01	Research Project	Project	Project Report	During semester			
MCSSE-INT-01	**Module: Internship**				**me**	**3/4**	**10**
MCSSE-INT-01	Internship	Project/ Internship	Internship Report	Summer Break			
	Management Modules						**5**
MCSSE-MGT-03	**Module: Transformational Change Management**				**m**	**3**	**5**
MCSSE-MGT-03	Transformational Change Management	Lecture	Presentation	During semester			
	Leadership / Academic Skills Modules				**m**		**5**
MCSSE-LAS-03	**Module: Agile Leadership and Strategic Management**				**m**	**3**	**2,5**
MCSSE-LAS-03	Agile Leadership and Strategic Management	Lecture	Presentations	During semester			
MCSSE-LAS-04	**Module: Customer-centric Mindset and Agile Delivery Management**				**m**	**3**	**2,5**
MCSSE-LAS-04	Customer-centric Mindset and Agile Delivery Management	Lecture	Presentation	During semester			
Semester 4							**30**
	Master Thesis						**30**
MCSSE-THE-01	**Module: Master Thesis MSc CSSE**				**m**	**4**	**30**
MCSSE-THE-01	Master Thesis	Thesis					
Total CP							**120**

[1] Each lecture period lasts 14 semester weeks and is followed by reading and examination days. Written examinations are centrally scheduled during weeks 15 and 16. For all other assessment types, the timeframes indicated in the above table stipulate the

[2] m = mandatory, me = mandatory elective

[3]For the Startup Track, students can take three 5 CP capstone project

For the industry track, students can take one 10 CP summer internship and one 5 CP technical core module

For the research track, students can take two 5 CP research lab modules and one CP research project

Further Core Modules							
Software Engineering							
MCSSE-SE-04	**Further Core Module: Advances in Software Engineering**				**me**	**3**	**5**
MCSSE-SE-04-A	Advances in Software Engineering	Lecture	Written examination	During semester			2.5
MCSSE-SE-04-B	Advances in Software Engineering - Lab	Lab	Project Assessment	During semester			2.5
MCSSE-SE-05	**Further Core Module: Advanced Databases**				**me**	**2**	**5**
MCSSE-SE-05-A	Advanced Databases	Lecture	Written examination	Examination Period			2.5
MCSSE-SE-05-B	Advanced Databases Lab	Lab	Laboratory Report	During semester			2.5
MCSSE-SE-07	**Further Core Module: Advanced AI-Native Software Engineering with Constructor Fabric**				**me**	**1**	**5**
MCSSE-SE-07-A	Advanced AI-Native Software Engineering with Constructor Fabric	Seminar	Project	During semester			
Cybersecurity							
Each student must choose at least 5 ECTS from this area. In order to specialize at least 20 ECTS must be chosen including all main content modules.							
MCSSE-CYB-01	**Main content: Cryptography**				**me**	**1**	**5**
MCSSE-CYB-01	Cryptography	Lecture	Written examination	Examination Period			
MCSSE-CYB-02	**Main content: System Security**				**me**	**2**	**5**
MCSSE-CYB-02	System Security	Lecture	Written examination	Examination Period			
MCSSE-CYB-03	**Main content: Network Security**				**me**	**3**	**5**
MCSSE-CYB-03	Network Security	Lecture	Written examination	Examination Period			
MDSSB-SOCB-01	**Further Core Module: Cybercriminology**				**me**	**3**	**5**
MDSSB-SOCB-01	Cybercriminology	Seminar	Term Paper	Examination Period			
Artificial Intelligence							
Each student must choose at least 5 ECTS from this area. In order to specialize at least 20 ECTS must be chosen including all main content modules							
MCSSE-AI-01	**Main content: Deep Learning**				**me**	**1 or 3**	**5**
MCSSE-AI-01	Deep Learning	Lecture	Written examination	Examination Period			
MCSSE-AI-02	**Main content: Intelligent Autonomous Systems**				**me**	**1 or 3**	**5**
MCSSE-AI-02	Intelligent Autonomous Systems	Lecture	Written examination	Examination Period			
MCSSE-AI-03	**Main content: Symbolic Artificial Intelligence**				**me**	**2**	**5**
MCSSE-AI-03	Symbolic Artificial Intelligence	Lecture	Written examination	Examination Period			
MDSSB-MET-02	**Further Core Module: Text Analysis and Natural Language Processing**				**me**	**2**	**5**
MDSSB-MET-02	Text Analysis and Natural Language Processing	Seminar/Lab	Project Report	Examination Period			
MDE-CO-02	**Further Core Module: Data Analytics**				**me**	**1**	**5**
MCDE-CO-02	Data Analytics	Lecture	Project Report	Examination Period			
MDE-CO-04	**Further Core Module: Machine Learning**				**me**	**2**	**5**
MDE-CO-04	Machine Learning	Lecture	Written examination	Examination Period			
Breakthrough modules							
MDE-QT-02	**Quantum Informatics**				**me**	**1/3**	**5**
MDE-QT-02-A	Quantum Informatics	Lecture	Written examination	Examination Period			2.5
MDE-QT-02-B	Quantum Informatics - Lab	Lab	Portfolio Assessment	During the semester			2.5
	Research Project						**5**
MCSSE-RP-01	**Module: Research Project**				**me**	**3**	**5**
MCSSE-RP-01	Research Project	Project	Project Report	Examination period			

Figure 2:Study and Examination Plan

2.4 Technical CORE Modules

The main subject areas of the CORE modules are:

- Software Engineering,
- Cybersecurity, and
- Artificial Intelligence.

Additionally, there is an area with offerings that are assumed to become breakthrough disciplines in the field.

All students take 15 ECTS of lecture modules from the Software Engineering subject area which reflects the orientation of the study program**.** It is also mandatory to take at least one *main content* module (5 ECTS) from Cybersecurity and Artificial Intelligence each.

Students select one of the three specialization areas in which they have to at least earn 20 ECTS.

2.4.1 Software Engineering Modules

The software engineering area exposes a broad range of methodological and systematic approaches for developing software and related applications in a professional environment. All three main content modules are mandatory. At least one further core module can be taken to make this area the specialization of a student.

To pursue a CSSE master, the following Software Engineering Core modules (15 CP) need to be taken as mandatory modules (m):

- CORE Module: Software Construction, Architecture and Engineering (m, 5 CP)
- CORE Module: Quality Engineering (m, 5 CP)
- CORE Module: Artificial Intelligence in Software Engineering (m, 5 CP)

Students choose another mandatory main content module (5 CP) from the specialization areas Cybersecurity and Artificial Intelligence.

Students following the Software Engineering Track need to select 5 CP from the mandatory elective (me) modules:

- CORE Module: Advances in Software Engineering (me, 5 CP)
- CORE Module: Advanced Databases (me, 5 CP)
- CORE Module: Advanced AI-Native Software Engineering with Constructor Fabric (me, 5 CP)

2.4.2 Cybersecurity Modules

In the Cybersecurity specialization, Cryptography is the entry module into the field. This content is complemented by extended modules on security methods, tools and technologies both on system and on network level.

Students following the Cybersecurity Track need to select 15 CP from the mandatory elective (me) modules:

- CORE Module: Cryptography (me, 5 CP)
- CORE Module: System Security (me, 5 CP)
- CORE Module: Network Security (me, 5 CP)
- CORE Module: Cybercriminology (me, 5 CP)

2.4.3 Artificial Intelligence Modules

The Artificial Intelligence specialization covers a spectrum of the field ranging from methods in machine learning over (symbolic) artificial intelligence techniques up to applications in cyberphysical systems. Students specializing in this area that have not been exposed to the field, so far, are suggested to take at least the courses on Data Analytics, Machine Learning, and Deep Learning. Students that have been exposed to the field before can immediately start into the main content modules via Deep Learning, Symbolic Artificial Intelligence and Intelligent Autonomous Systems.

Students following the Artificial Intelligence Track need to select 15 CP from the mandatory elective (me) modules:

- CORE Module: Deep Learning (me, 5 CP)
- CORE Module: Intelligent Autonomous Systems (me, 5 CP)
- CORE Module: Symbolic Artificial Intelligence (me, 5 CP)
- CORE Module: Text Analysis and Natural Language Processing (me, 5 CP)
- CORE Module: Data Analytics (me, 5 CP)
- CORE Module: Machine Learning (me, 5 CP)

2.4.4 Breakthrough Area Modules

Digital Leadership requires a long-term perspective. In this elective area, students are exposed to potential future breakthrough applications in the field. This area is expanded as more such applications are identified.

As part of the Core Technical content, this module can also be selected as a mandatory elective module.

- CORE Module: Quantum Informatics (me, 5 CP)

2.5 Management Modules

To equip students with market-relevant management skills they take modules in the fields of product development, marketing and change management. All modules are mandatory for the program.

To pursue a CSSE master, the following Management modules (15 CP) need to be taken as mandatory modules (m):

- Management Module: Agile Product Development & Design (m, 5 CP)
- Management Module: Product Innovation & Marketing (m, 5 CP)
- Management Module: Transformational Change Management (m, 5 CP)

2.6 Leadership / Academic Skills Modules

Success in industry and research is further strengthened with a set of Leadership and Academic Skills Modules. All modules below have to be taken in order to graduate.

To pursue a CSSE master, the following Leadership/ Academic Skills modules (15 CP) need to be taken as mandatory modules (m):

- Leadership / Academic Skills Module: Entrepreneurship & Intrapreneurship (m, 2.5 CP)
- Leadership / Academic Skills Module: Communication & Presentation Skills for Executives (m, 2.5 CP)
- Leadership / Academic Skills Module: Organizational Behavior (m, 2.5 CP)
- Leadership / Academic Skills Module: Academic Writing Skills / Intercultural Training (m, 2.5 CP)
- Leadership / Academic Skills Module: Agile Leadership and Strategic Management (m, 2.5 CP)
- Leadership / Academic Skills Module: Customer-centric Mindset and Agile Delivery Management (m, 2.5 CP)

2.7 Project, Capstone Project & Master Thesis

Students select a startup, an industrial, or a research orientation.

To explore the full development process of a software application with relation to the areas of specialization of the program, students may take the three modules of the Capstone Project. If selected, it is highly recommended to take the three modules in their numerical order, to gain full experience of the project.

Thus, students following the startup orientation complete these modules:

- Capstone Module: Capstone Project I (me, 5 CP)
- Capstone Module: Capstone Project II (me, 5 CP)
- Capstone Module: Capstone Project III (me, 5 CP)

Students following the industry orientation complete these modules:

- 10 CP Internship
- 5 CP additional technical core module

Students following the research orientation complete these modules:

- 5 CP Research Lab I
- 5 CP Research Lab II
- 5 CP Research Project

Students not pursuing the research orientation but with an interest in academic research can replace in their third semester one Technical CORE Module by the Research Project, which is carried out in one of the research areas of the Faculty.

The master studies are concluded by a Master Thesis (30 CP), which extends over the fourth and final semester.
Detailed module descriptions in their latest version are available in the catalogue on CampusNet.

3 Computer Science and Software Engineering Graduate Program Regulations

3.1 Scope of these Regulations

The regulations in this handbook are valid for all students who entered the Computer Science and Software Engineering graduate program at Constructor University in Fall 2026. In case of conflict between the regulations in this handbook and the general Policies for Master Studies, the latter apply (see https://constructor.university/student-life/student-services/university-policies/academic-policies).

In exceptional cases, certain necessary deviations from the regulations of this study handbook might occur during the course of study (e.g., change of the semester sequence, assessment type, or the teaching mode of courses).

Updates to Study Program Handbooks are based on the policies approved by the Academic Senate on substantial and nonsubstantial changes to study programs. Students are integrated in the decision-making process through their respective committee representatives. All students affected by the changes will be properly informed.

In general, Constructor University therefore reserves the right to change or modify the regulations of the program handbook also after its publication at any time and in its sole discretion.

3.2 Examination Concept

According to the Policies for Bachelor and Master studies, modules generally carry at least five ECTS. Each program ensures appropriate examination frequency and organization, justified in an examination concept and regularly reviewed with student involvement.

Constructor University's examination concept follows the principle of Constructive Alignment (Biggs 1996), ensuring that learning outcomes, activities, and assessments are consistently aligned: students learn what is intended, and assessments both measure and shape learning. Where one assessment cannot cover all Intended Learning Outcomes (ILOs) complementary forms could be used (e.g., written exams plus lab reports). Module descriptions map ILOs to assessments.

In specific contexts, such as asynchronous online modules or courses emphasizing student engagement, Module Achievements or other types of formative assessments may support competence-oriented assessment.

Student feedback, embedded in the Quality Assurance System (QAS), systematically monitors workload, competence orientation, and alignment of ILOs and assessments. Student surveys and feedback are regulated in the Policy for student surveys and evaluations.

3.3 Degree

Upon successful completion of the study program, students are awarded a Master of Science (MSc) degree in Computer Science and Software Engineering.

3.4 Graduation Requirements

To graduate, students take out of these modules a total of 120 ECTS with

- 45 CP in technical core modules (further detailed below),
- 15 CP in mandatory management modules,
- 15 CP in mandatory leadership / academic skills modules,
- 15 CP in startup / industrial / research orientation modules, and
- 30 CP in the master thesis module.

All students select one specialization area (Software Engineering, Artificial Intelligence, or Cybersecurity) in which they must earn at least 20 CP in technical core modules. Furthermore, all students must complete the following mandatory technical core modules:

- 5 CP Software Construction, Architecture and Engineering
- 5 CP Quality Engineering
- 5 CP Artificial Intelligence in Software Engineering

All students must earn at least 5 CP from the following Artificial Intelligence modules. Students specializing in Artificial Intelligence must take all three modules as part of their 20 CP requirement.

- 5 CP Symbolic Artificial Intelligence
- 5 CP Deep Learning
- 5 CP Intelligent Autonomous Systems

All students must earn at least 5 CP from the following Cybersecurity modules. Students specializing in Cybersecurity must take all three modules as part of their 20 CP requirement.

- 5 CP Cryptography
- 5 CP Network Security
- 5 CP System Security

Students select a startup, an industrial, or a research orientation. Students following the startup orientation complete these modules:

- 5 CP Capstone Project I
- 5 CP Capstone Project II
- 5 CP Capstone Project III

Students following the industry orientation complete these modules:

- 10 CP Internship
- 5 CP additional technical core module

Students following the research orientation complete these modules:

- 5 CP Research Lab I
- 5 CP Research Lab II

- 5 CP Research Project

3.5 Other Program-specific Policies & Practices

Close contact and cooperation between program representatives and students is crucial. Therefore, regular meetings are held to continuously evaluate the program, its modules and workshops, supervision, and opportunities. In doing so, the study program chair and involved faculty gain important insights into students' experiences, demands, and overall impressions of the program. On the module component level, students are asked to perform module component evaluations to ensure that the modules are high-quality and that lecturers can make any necessary changes.

The study program chair makes intensive use of this feedback as well as feedback from industry partners to improve the learning environment, the program's offering, and its progress. The current program was shaped through input from previous experiences and discussions with several stakeholders, including students and industry practitioners.

In exceptional cases, certain necessary deviations from the regulations of this study handbook might occur during the course of study (e.g., change of the semester sequence, assessment type, or the teaching mode of courses). Constructor University reserves therefore the right to modify the regulations of the program handbook.

4 Module Descriptions

4.1 Software Engineering Modules

4.1.1 Software Construction, Architecture and Engineering

Module Name	Software Construction, Architecture and Engineering
Module Code	MCSSE-SE-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Dr. Joaquín Aguado

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory

Student Workload	
Lecture	70
Independent Study	55
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Software Construction, Architecture and Engineering	MCSSE-SE-01-A	Lecture	5

Module Description

This course covers software construction, engineering, and architecture, combining practical techniques for developing software with the study of architectural decisions that influence system quality. Students examine software construction practices, including design patterns, programming contracts, agile methods, and requirements engineering, together with methods for designing, analysing and documenting evolving software architectures.

The module considers how architectural decisions affect maintainability, scalability, and reliability, emphasising practices that produce software which can be changed reliably and cost-effectively. Students are also introduced to modern development tools and workflows that support construction and architectural practices.

Students successfully completing this course will have acquired an understanding of modern techniques for software construction and architecture and the ability to apply these to complex software systems.

Recommended Knowledge

Basic programming experience and familiarity with software development concepts.

Usability and Relationship to other Modules

More advanced software engineering techniques are covered in Advances in Software Engineering (MCSSE-SE-04-A), to which the present course is a prerequisite.

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Explain the principles of software construction, engineering practices, and architectural design, including the role of contracts for ensuring correctness and maintainability.
2	Apply	Apply design patterns, programming contracts and agile methods to construct software modules and components that are maintainable and reliable.
3	Analyze	Analyze architectural decisions and their impact on system quality attributes such as maintainability, scalability, and reliability.
4	Evaluate	Critically assess software construction and architectural strategies to identify strengths, weaknesses and areas of improvement in complex systems.
5	Apply/Understand	Apply and evaluate practices that facilitate software change, ensuring systems remain maintainable and adaptable over time.
6	Create/Synthesize	Design and construct software systems that integrate engineering practices and architectural principles to produce maintainable, reliable, and adaptable solutions.
7	Reflect/Evaluate	Reflect on the effectiveness of construction and architectural choices and propose improvements based on best practices and modern development workflows.

Indicative Literature

- Bass, L., Clements, P. and Kazman, R. (2013) Software Architecture in Practice (3rd Edition). Addison-Wesley.
- Hunt, A. & Thomas, D. (2019) The Pragmatic Programmer: Your Journey to Mastery (20th Anniversary Edition). Pearson.
- Meyer, B (2009) Touch of Class: Learning to Program Well Using Objects and Contracts.
- Neal Ford, Mark Richards, Pramod Sadalage and Zhamak Dehghani (2022). Software Architecture: The Hard Parts: Modern Trade-Off Analyses for Distributed Architectures. O'Reilly Media.
- Pfleeger, S. and Atlee, J.M. (2010). Software Engineering: Theory and Practice (4th Edition).
- Richards, M. & Ford, N. (2025) Fundamentals of Software Architecture: A Modern Engineering Approach. O'Reilly Media.
- Taylor, R.N., Medvidovic, N. and Dashofy, E.M. (2009) Software Architecture: Foundations, Theory, and Practice. Wiley.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Software Construction, Architecture and Engineering	Written Examination	120 Minutes	100	45%	1-7

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.1.2 Quality Engineering

Module Name	Quality Engineering
Module Code	MCSSE-SE-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Mauro Pezzé

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	1	Mandatory

Student Workload	
Lecture	35
Independent Study	90
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Quality Engineering	MCSSE-SE-02	Lecture	5

Module Description

Software quality is an essential part of the software development and cannot be guaranteed a-priori, but most be verified both during and after the development. This course introduces the main testing and analysis techniques that can be used to identify failures and verify the quality of software systems. The course introduces the general testing and analysis principles and the basic techniques, shows how to apply them to solve relevant quality problems, illustrates complementarities and differences among the different techniques, and presents the organization of a coherent quality process. The course provides the elements needed to understand principles, techniques and process that comprise the basic background of test designer, quality manager and project manager. At the end of the course, the students will be able to define and implement quality plans for complex software systems. The student will have the basic knowledge of a project and a quality manager.

Students will know in the first session which assignments will be part of the portfolio examination.

Recommended Knowledge

-Programming skills in an imperative language at CS bachelor level

- Algorithms and data structure at CS bachelor level

- Basic skills in software testing: structural testing, Junit

- Basic knowledge of software engineering and IDEs at CS bachelor level

- Discrete math at CS bachelor level

Intended Learning Outcomes

No	Competence	ILO
1	Manage	Manage a software quality process.
2	Select	Select and implement a suitable set of testing and analysis activities to certify the quality of software systems.
3	Understand	Understand the core principles of software testing and program analysis.
4	Master	Master the basic techniques underlying software testing and program analysis.
5	Choose	Choose the suitable approaches to address the different testing and analysis programs.
6	Design	Design and monitor a suitable quality process.

Indicative Literature

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Quality Engineering	Portfolio Assessment	(Individual Assignments, Group Assignments)	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum pass 45%- portfolio.

Module Achievement

None.

4.1.3 Advances in Software Engineering

Module Name	Advances in Software Engineering
Module Code	MCSSE-SE-04
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Bertrand Meyer

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Lecture	17.5
Laboratory	17.5
Independent Study	90
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Advances in Software Engineering	MCSSE-SE-04-A	Lecture	2.5
Advances in Software Engineering - Lab	MCSSE-SE-04-B	Laboratory	2.5

Module Description

The course covers topics of modern software engineering beyond the basic concepts covered in the first- semester SCAE course (Software Construction, Architecture and Engineering). After taking it, the students will master important techniques for high-quality software development and management, particularly in three areas: requirements engineering; formal methods and software verification; project management and agile methods.

The second one of these areas, foral specification and verification, occupies about half of the course and covers various verification techniques such as axiomatic (Floyd-Hoare-Dijkstra) semantics, model checking, abstract interpretation and symbolic execution. Students should be aware that this part of the course relies on mathematical techniques (mostly, elementary logic) and that the project will require performing a formal verification (computer-supported mathematical proof) of a simple program.

While the course emphasizes non-programming aspects and is applicable to software development using any programming language, exercises that need a specification, design or programming notation generally use Eiffel. The lectures and exercises on axiomatic semantics rely on the AutoProof software verification system under development at the Chair of Software Engineering at Constructor Institute.

Recommended Knowledge

- Familiarity with basics of software engineering and software architecture

- Programming experience

Usability and Relationship to other Modules

Software Construction, Architecture and Engineering (MCSSE-SE-01) is a prerequisite. Exceptions are subject to approval by the instructor.

Intended Learning Outcomes

No	Competence	ILO
1	Apply	Apply techniques of formal software verification, particularly axiomatic semantics, to proving program correctness.
2	Use	Use a program-proving framework.
3	Perform	Perform effective requirements.
4	Apply	Apply requirements techniques such as use cases and object-oriented requirements.
5	Use	Use agile development techniques to manage a project.
6	Make	Make the difference between productive and harmful agile ideas.
7	Combine	Combine agile methods with process models such as CMMI.

Indicative Literature

- Bertrand Meyer, Agile! The Good, the Hype and the Ugly, Springer. 2014.
- Bertrand Meyer, Handbook of Requirements Engineering and Business Analysis, Springer, 2022.
- Course notes by the instructor
- Flemming Nielson, Hanne Riis Nielson, Chris Hankin: Principles of Program Analysis, Springer, most recent edition .
- Patrick Cousot, Abstract Interpretation, MIT Press, 2021.
- Rustan Leino, Program Proofs, MIT Press, 2023.

Entry Requirements

Prerequisites	MCSSE-SE-01 Software Construction, Architecture and Engineering
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Advances in Software Engineering	Written Examination	90 minutes	50	45%	1-7
Advances in Software Engineering - Lab	Project Assessment		50	45%	1-7

Module Completion Requirements

- **Module Component Examination:** minimum for pass- 45%.

Module Achievement

None.

4.1.4 Advanced AI-Native Software Engineering with Constructor Fabric

Module Name	**Advanced AI-Native Software Engineering with Constructor Fabric**
Module Code	MCSSE-SE-07
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science and Software Engineering)
Module Coordinator	Prof. Giancarlo Succi

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc **Computer Science and Software Engineering**	1	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Term paper and Presentation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Advanced AI-Native Software Engineering with Constructor Fabric	MCSSE-SE-07-A	Seminar	5

Module Description

This course studies the engineering of software in an AI-native development environment, using Constructor Fabric as a concrete case. The primary focus is Constructor Studio, which supports the full software lifecycle — planning, design, agentic code generation, continuous integration and delivery, and product analytics — organized around an explicit world model of the system that AI agents plan and build against. The course examines what disciplined engineering requires when much of the implementation is produced by AI agents: structured specifications and contracts, traceability from requirements to code and tests, architectural decision records, non-functional requirement gates, and reproducible verification of behavior, security, and performance. It treats the toolchain as a stable abstraction layer that insulates teams from the rapid churn of underlying AI models and tools and weighs the trade-offs of this approach. Students then critically assess two complementary components: Constructor Gears, reusable platform and business modules that encapsulate external dependencies above the runtime, and Constructor Insight, which provides engineering-productivity measurement, delivery metrics, and production observability as evidence for decisions. Students complete an individual or small-team project in which they propose, implement in bounded scope, and evaluate an AI-assisted software change, supported by

appropriate engineering evidence and describe the results of such project in a term paper and then discuss them orally in a 10 minute presentation.

Usability and Relationship to other Modules

The course is suitable for students who already have a solid foundation in software engineering and distributed systems.

Intended Learning Outcomes

No	Competence	ILO
1	Critique	Critique AI-native development approaches with reference to specification quality, traceability, verification rigour, security boundaries, and exposure to model and tooling change.
2	Model	Model a software subsystem as an explicit specification — domain concepts, contracts, acceptance criteria, and cross-cutting policies — suitable for driving and constraining AI agents.
3	Design	Design an AI-assisted development workflow that preserves architectural and interface constraints and makes appropriate use of reusable platform modules.
4	Evaluate	Evaluate AI-generated software using appropriate evidence, including automated tests, security and performance measurements, delivery metrics, and traceability artefacts.
5	Implement	Implement a bounded proof of concept using AI agents that conform to stated architectural, interface, and quality constraints.
6	Defend	Defend a technical recommendation in writing and orally, explaining trade-offs, the role and limits of AI assistance, uncertainties, and the supporting evidence.

Indicative Literature

- Bass, Clements, and Kazman: Software Architecture in Practice.
- Constructor Studio documentation, course-provided architectural decision records, and selected readings on AI-assisted software engineering.

Entry Requirements

Prerequisites	**None**
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs

Advanced AI-Native Software Engineering with Constructor Fabric	Project		100	45%	1-6

Module Completion Requirements

- **Module Examination:** Minimum for pass- 45%

Module Achievement

None.

4.1.5 Advanced Databases

Module Name	Advanced Databases
Module Code	MCSSE-SE-05
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Peter Baumann

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective
DE-MSc Data Engineering	2	Mandatory Elective

Student Workload	
Lecture	40
Laboratory	40
Independent Study	45
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Advanced Databases	MCSSE-SE-05-A	Lecture	2.5
Advanced Databases Lab	MCSSE-SE-05-B	Laboratory	2.5

Module Description

This course deepens knowledge and skills in managing and serving Big Data with emphasis on flexibility and scalability. As a result of this course, students will know the state of the art in data management for particularly large and complex data, including in cloud-based data setups. Based on the Data Engineering Core lecture Data Management the course starts with a reinspection of classical SQL, preparing an overview of SQL query processing.

Based on this understanding opportunities of optimization and parallelization are discussed. Subsequently, novel developments in Big Data services are discussed. NoSQL approaches with their new data models are inspected, such as documents, graphs and arrays. This is contrasted with NewSQL and their novel techniques for competitive performance. Dedicated architectures are discussed, such as MapReduce.

This leads to general scalability considerations, with an emphasis on large-scale parallel and distributed processing. Throughout the course practical considerations play an important role, including practitioner hints on database modeling, tuning, and security. Practical guided hands-on exercises complement this.

The Advanced Databases module consists of a lecture and a project part.

In the lecture, concepts and skills are taught. The lecture part has no assignments - practicing database skills is covered in the project - but only a single exam at the end which contains problem solving as well as knowledge recap tasks.

In the accompanying project, students build teams on their own choice. Each team selects a project topic which they subsequently implement over the semester in weekly assignments, common to all teams. The project module has weekly result submissions which get graded; the overall module achievement consists of the simple average over all weekly achievements. A single assessment type cannot sufficiently test all intended learning outcomes. The programming evaluates practical skills, whereas the written examination assesses understanding of theoretical knowledge, core principles, and analytical reasoning.

Altogether, this allows teaching and assessing both conceptual and practical work in a tightly intertwined manner.

Usability and Relationship to other Modules

For DE: pre-requisite Introduction to Data Management with Python.

Recommended Knowledge

- Good Mandatory knowledge of SQL

- Working knowledge of fundamental data structures, such as trees

- Working knowledge of computer architectures

- Good command of at least one programming language, as several languages will be used in the lab

Intended Learning Outcomes

No	Competence	ILO
1	Summarize	Summarize the state of the art in data management for particularly large and complex data.
2	Establish	Establish criteria for selecting adequate scalable data management technology based on various criteria.
3	Establish	Establish a state-of-the-art database schema for a given application scenario.
4	Tune	Tune a relational database for best performance on some given query workload.
5	Consider	Consider adequately security aspects in databases.
6	Develop	Develop applications using Web and database technology.

Indicative Literature

- McLellan (2013): Big Data: An Overview https://www.zdnet.com/article/big-data-an-overview/.
- S. Akter & S. Fosso Wamba, Big data analytics in e-commerce: A systematic review and agenda for future research, 2016. Electronic Markets, 26 173-194.

- Z. Lv, H. Song, P. Basanta-Val, A. Steed and M. Jo. "Next-Generation Big Data Analytics: State of the Art, Challenges, and Future Research Topics," in IEEE Transactions on Industrial Informatics, vol. 13, no. 4, pp. 1891-1899, Aug. 2017.

Entry Requirements

Prerequisites	MDE-CO-07 Data Management and Databases
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Advanced Databases	Written Examination	120 Minutes	50	45%	1-5
Advanced Databases Lab	Laboratory Report		50	45%	3-6

Module Completion Requirements

- **Module Component Examination:** minimum pass 45% of each assessment.

Module Achievement

None.

4.1.6 Artificial Intelligence in Software Engineering

Module Name	Artificial Intelligence in Software Engineering
Module Code	MCSSE-SE-06
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Dr. Joaquín Aguado

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory

Student Workload	
Lecture	70
Independent Study	55
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Artificial Intelligence in Software Engineering	MCSSE-SE-06-A	Lecture	5

Module Description

This course promotes critical thinking in evaluating and applying AI solutions to Software Engineering. It studies the integration of generative AI code assistants (based on large language models) into the software development lifecycle. The focus is on understanding how AI can support development that results in maintainable, testable and adaptable code, while recognizing the limitations, risks, and non-deterministic behavior of AI systems

Practical applications may involve working with open-source projects, legacy code, AI-assisted code editors or technologies such as the Model Context Protocol (MCP), providing project context to AI tools. Students may also use AI tools to refactor, document and analyze existing code bases.

A fundamental aspect of the course is the design and execution of test cases to validate AI-generated code and verify that it meets maintainability standards. Students will develop skills to critically assess AI-generated code, compare tools

By the end of the course, students should be able to:

- Evaluate and compare AI-assisted software development tools in terms of effectiveness and maintainability.

- Apply AI tools to real-world software projects, ensuring maintainability and test coverage.

- Design and implement test cases to validate AI-generated code.

- Integrate AI-assisted development practices into software engineering workflows while maintaining code quality

Recommended Knowledge

Students should have some experience in software development, including programming, design patterns and testing, as well as a basic understanding of AI and machine-learning concepts

Intended Learning Outcomes

No	Competence	ILO
1	Analyze	Critically evaluate AI-assisted software development tools and techniques, identifying their strengths, limitations, risks, and suitability for different stages of the software development lifecycle.
2	Evaluate	Compare and assess AI-generated code against software engineering standards, including maintainability, readability, and testability.
3	Create/Synthesize	Design software solutions by integrating AI-assisted coding tools and human creativity, applying architectural and design patterns effectively in practical workflows.
4	Apply	Implement test cases and validation strategies for AI-generated code, including AI-assisted testing and code analysis, ensuring reliability, maintainability, and alignment with software engineering best practices in the presence of potentially non-deterministic outputs.
5	Create/Synthesize	Develop AI-assisted software development workflows, including the integration of AI tools and emerging interfaces such as the Model Context Protocol (MCP), that combine human problem-solving and creativity with AI capabilities to produce high-quality code.
6	Reflect/Evaluate	Critically reflect on the outcomes of AI-assisted development, proposing improvements to workflows and tool usage based on evidence and practical experience.

Indicative Literature

- Huyen, C. (2025) AI Engineering: Building Applications with Foundation Models. O'Reilly Media.
- Lakshmanan, V. and Hapke, H. (2025) Generative AI Design Patterns: Solutions to Common Challenges When Building GenAI Agents and Applications. O'Reilly Media.
- Nguyen-Duc, A., Abrahamsson, P. and Khomh, F. (eds.) (2024) Generative AI for Effective Software Development. Springer.
- Pereira, S. (2025) Generative AI for Software Development: Building Software Faster and More Effectively. O'Reilly Media.

Entry Requirements

Prerequisites	MCSSE-SE-01 Software Construction, Architecture and Engineering MCSSE-SE-02 Quality Engineering
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Artificial Intelligence in Software Engineering	Oral Examination	30 minutes	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.2 Cybersecurity Modules

4.2.1 Cryptography

Module Name	Cryptography
Module Code	MCSSE-CYB-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Exam Preparation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Cryptography	MCSSE-CYB-01	Lecture	5

Module Description

Information security requires techniques to protect information and to secure communication. Cryptography studies the design of cryptographic algorithms that can ensure confidentiality, integrity, and authenticity of data and messages exchanged in a secure communication protocol or when data is stored. This module focuses on the mathematical and algorithmic foundations of cryptography, and it covers the application of basic primitives to solve common information security challenges. Students familiar with the foundations of cryptographic algorithms will be able to judge the applicability and limitations of different cryptographic algorithms.

Recommended Knowledge

Students are expected to have a solid mathematical foundation. Students should review basic concepts of number theory, probability theory, and complexity theory in preparation for this module.

Usability and Relationship to other Modules

- The module serves as the foundational module in the cyber security specialization in CSSE. Other modules related to cyber security build on this module.

- This module belongs to the Software Engineering Track in the MSc AST

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand the mathematical problems on which cryptographic algorithms are built.
2	Describe	Describe pseudo random number generators and pseudo random functions.
3	Evaluate	Evaluate the strengths, weaknesses, and the applicability of cryptographic algorithms.
4	Select	Select from a set of symmetric block cipher, message integrity, and authenticated encryption algorithms.
5	Contrast	Contrast different asymmetric ciphers (finite field based, elliptic curve based, lattice based, hash based).
6	Explain	Explain the notion of quantum resistant cryptographic algorithms.
7	Analyze	Analyze the properties of cryptographic protocols such as key exchange mechanisms.
8	Apply	Apply techniques to analyze cryptographic protocols and their implementations.
9	Explain	Explain homomorphic encryption schemes and differential privacy.

Indicative Literature

- Bruce Schneier: Applied Cryptography, 20th Anniversary Edition, Wiley, 2015.
- Dan Boneh, Victor Shoup: A Graduate Course in Applied Cryptography, version 0.5, online, 2020.
- Simon Singh: The Code Book: Science of Secrecy from Ancient Egypt to Quantum Cryptography, Anchor Books, 2000.
- Wm. Arthur Conklin, Gregory White: Principles of Computer Security, 5th Edition, McGraw-Hill, 2018.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Cryptography	Written Examination	120 Minutes	100	45%	1-9

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.2.2 System Security

Module Name	System Security
Module Code	MCSSE-CYB-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	2	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Exam Preparation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
System Security	MCSSE-CYB-02	Lecture	5

Module Description

This module focuses on system level security aspects of computing systems. The module starts with investigating attacks on the microarchitecture of computing systems, such as attacks to gain information from side channels targeting caches. It then introduces trusted execution environments that use hardware isolation mechanisms to provide protected storage for keys and to bootstrap the integrity of bootloaders and the loaded operating systems. Students learn about the different levels of isolation that can be achieved using various types of hypervisors or sandboxing mechanisms. Techniques that can be used to protect a system against misbehaving code and malware are introduced. Students will gain knowledge how protected data storage components can be provided at the system level and how systems can offer support for collections of (distributed) authentication mechanisms. Finally, the module will discusses how authorization mechanisms are realized in the different system software components and how they can be used to define effective security policies.

Recommended Knowledge

Students are expected to be familiar with program execution at the system and machine level. Students should have a good understanding of computer architecture and operating systems at the level of typical undergraduate modules covering these topics. Students who have not taken an undergraduate course on computer architecture or operating systems yet may consider taking a

remedial course or an online course to obtain a fundamental understanding how computer systems function.

Usability and Relationship to other Modules

- The module serves as a mandatory elective module in the cyber security specialization. Parts of the module require an understanding of cryptographic algorithms.

- This module belongs to the Software Engineering Track in the MSc AST

Intended Learning Outcomes

No	Competence	ILO
1	Describe	Describe microarchitectural attacks and computer components and suitable counter measures.
2	Illustrate	Illustrate trusted execution environments and how they can be used to bootstrap security.
3	Compare	Compare the isolation achieved by hypervisors and operating system mechanisms.
4	Assess	Assess application layer isolation and sandboxing mechanisms.
5	Explain	Explain how systems can identify misbehaving code and protect themselves against malware.
6	Outline	Outline how protected data storage can be implemented.
7	Recommend	Recommend authentication methods suitable for different kinds of applications.
8	Compose	Compose authorization mechanisms to define effective security policies.

Indicative Literature

- Swarup Bhunia: Hardware Security: A Hands-on Learning Approach, Morgan Kaufmann, 2018.
- William Stallings, Lawrie Brown: Computer Security: Principles and Practice, 4th edition, Pearson, 2018.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs

System Security	Written Examination	120 Minutes	100	45%	1-8

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.2.3 Network Security

Module Name	Network Security
Module Code	MCSSE-CYB-03
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	3	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Exam Preparation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Network Security	MCSSE-CYB-03	Lecture	5

Module Description

Computer networks such as the Internet connect millions of computing systems, enable a fast exchange of information, and provide the technological basis on which large parts of the modern online economy are built. Computer networks, however, also expose an infrastructure that can be used by criminals or nation states to attack computing systems, to control the flow of messages, or to distribute malicious programs to potentially large numbers of targeted systems. This module educates students about how computer networks can be used to obtain information about remote systems, to manipulate the flow of data traffic, to disrupt access to remote services, or to control malicious software using botnets and distributed command and control channels. The module also covers technologies that help to protect the integrity of computer networks and that provide generic security services that can be used by applications requiring secure communication.

Recommended Knowledge

Students are expected to have a general understanding of computer networks, as provided by typical undergraduate modules on computer networks. Students who have not taken an undergraduate course on computer networks yet may consider taking a remedial course or an online course to obtain a fundamental understanding how computer networks function.

Usability and Relationship to other Modules

- The module serves as a mandatory elective module in the cyber security specialization. It builds on the cryptography module, which provides the necessary knowledge of cryptographic primitives that are used to protect data exchanged over computer networks and to authenticate communicating peers.

- This module belongs to the Software Engineering Track in the MSc AST

Intended Learning Outcomes

No	Competence	ILO
1	Describe	Describe techniques to obtain information about networked computing systems.
2	Contrast	Contrast mechanisms in the different network protocol layers for traffic manipulation and redirection.
3	Explain	Explain how distributed denial of service attacks are executed and how botnets are constructed.
4	Evaluate	Evaluate security mechanisms such as firewalls and anomaly / intrusion detection systems.
5	Analyze	Analyze generic security protocols such as IPsec, TLS, SSH and how they have evolved.
6	Compare	Compare protocols aiming to secure the network infrastructure (name resolution, routing).
7	Evaluate	Evaluate the security properties of modern software-defined network architectures.
8	Design	Design scalable solutions for protecting communication in distributed applications.

Indicative Literature

- Chris McNab, Network Security Assessment, O'Reilly, 2017.
- James Forshaw: Attacking Network Protocols, A Hacker's Guide to Capture, Analysis, and Exploitation, no starch press, 2017.
- William Stallings: Cryptography and Network Security: Principles and Practice, 7th edition, Pearsons, 2018.

Entry Requirements

Prerequisites	MCSSE-CYB-01 Cryptography
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Network Security	Written Examination	120 Minutes	100	45%	1-8

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.2.4 Cybercriminology

Module Name	Cybercriminology
Module Code	MDSSB-SOCB-01
Module ECTS	5
Program Owner	DSSB-MSc (Data Science for Society and Business)
Module Coordinator	Prof. Dr. Hilke Brockmann

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective
DSSB-MSc Data Science for Society and Business	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective
DSSB-MSc Data Science for Society and Business	3	Mandatory Elective

Student Workload	
Seminar	35
Independent Study	90
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Cybercriminology	MDSSB-SOCB-01	Seminar	5

Module Description

New technologies also provide new spaces and tools for deviant behavior. Cybercriminology addresses crimes committed on or facilitated by the Internet. These encompass crimes against computers—from hacking and malware attacks to cyberwarfare, crimes against intellectual, virtual, and analog properties, crimes against persons like cyberbullying and cyberstalking, and crimes involving illicit content from hate speech, to adult and child pornography

In this module, we will learn about these cybercriminal offenses and their prevalence, along with discussing prominent court cases. We get insights into the socio-demographic and psychological profiles of cybercrime offenders and victims. We interrogate national and international cybercrime jurisdiction, policing structures, and policing techniques. At the end of the module, students will be able to engage with cybercrime experts to design and undertake policing cybercrime studies, and draft political and technical solutions to fight cybercrimes.

Recommended Knowledge

- Python or R

- Watch the ted-talk: https://www.youtube.com/watch?v=c_2Ja-OTmGc

Intended Learning Outcomes

No	Competence	ILO
1	Know	Know and understand the core concepts of cybercriminology, policing structures and techniques, and national as well as international cybercrime jurisdiction.
2	Demonstrate	Demonstrate the ability to critically, autonomously, and creatively identify and formulate cybercrime related problems.
3	Demonstrate	Demonstrate methodological knowledge in studying and critically analyzing cybercrime research questions.
4	Find	Find best solutions to secure private persons, business organizations, and entire societies from cybercrime offenses.
5	Demonstrate	Demonstrate insights into the possibilities and limitations of cybercrime research and their role in the society.
6	Formulate	Formulate policy recommendations to secure firms, organizations, and private persons from cybercrimes.

Indicative Literature

- Jaishankar (Ed) (2011) Cyber Criminology. Exploring Internet Crimes and Criminal Behavior. Coba Raton: Taylor & Francis. Maimon, Louderback (2019) Cyber-Dependent Crimes: An Interdisciplinary Review. Annual Review of Criminology 2, 191-216.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Cybercriminology	Term Paper	3000 - 4000 words	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3 Artificial Intelligence Modules

4.3.1 Deep Learning

Module Name	Deep Learning
Module Code	MCSSE-AI-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Dr. Dmitry Kropotov

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective
DE-MSc Data Engineering	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective
DE-MSc Data Engineering	3	Mandatory Elective
PHDS-BSc Physics and Data Science	5	Mandatory Elective
SDT-BSc Software, Data and Technology	5	Mandatory Elective

Student Workload	
Exam Preparation	20
Independent Study	70
Lecture	35
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Deep Learning	MCSSE-AI-01	Lecture	5

Module Description

In machine learning we aim at extracting meaningful representations, patterns and regularities from high-dimensional data. In recent years, researchers from various disciplines have developed “deep” hierarchical models, i.e. models that consist of multiple layers of nonlinear processing. An important property of these models is that they can “learn” by reusing and combining intermediate concepts, so that these models can be used successfully in a variety of domains, including information retrieval,

natural language processing, and visual object detection. After a brief introduction into core knowledge related to training, model evaluation and multilayer perceptrons, this module focuses on exposing students to deep learning techniques including convolutional and recurrent neural networks, autoencoders, generative adversarial networks and reinforcement learning. The central aim is hence to enable students to critically assess and apply modern methods in machine learning.

Recommended Knowledge

- This module is recommended for students that have been exposed to core knowledge in machine learning / statistical learning on undergraduate level. Students without this background knowledge can still join since required core knowledge is re-introduced. Preparation via auxiliary literature or online courses will facilitate the start into the course.

- Strong knowledge and abilities in mathematics (linear algebra, calculus).

Usability and Relationship to other Modules

While the graduate level modules "Data Analytics" and "Machine Learning" provide an applied introduction to the field and are therefore recommended for students with a focus on Software Engineering or Cybersecurity, this module complements the undergraduate module "Machine Learning" or can be used independently as a strong introduction to the field of Deep Learning.

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand core techniques to train neural networks.
2	Select	Select from modern neural network architectures the most appropriate method (e.g. convolutional and recurrent neural networks) based on given input data.
3	Contrast	Contrast different recent unsupervised learning methods including autoencoders and generative adversarial networks.
4	Describe	Describe techniques in reinforcement learning.

Indicative Literature

- Aurélien Géron: Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, 2nd Edition, O'Reilly, 2019.
- Charu C. Aggarwal: Neural Networks and Deep Learning – A Textbook, Springer, 2018.
- Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, 2006.
- Ian Goodfellow, Yoshua Bengio, Aaron Courville: Deep Learning, MIT Press, 2016.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Deep Learning	Written Examination	120 Minutes	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3.2 Intelligent Autonomous Systems

Module Name	Intelligent Autonomous Systems
Module Code	MCSSE-AI-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Andreas Birk

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Exam Preparation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Intelligent Autonomous Systems	MCSSE-AI-02	Lecture	5

Module Description

This module deals with the foundations of modern AI linking it to software development for applications in the real world. To this end, it provides an overview on intelligent autonomous systems (IAS), i.e., processes and machinery that can execute complex tasks in complex environments without permanent human supervision. Examples include driver assistance up to fully autonomous cars, intelligent mobile robots, or warehouse automation. The module includes hands-on elements to familiarize students with the programming and software architecture aspects for developing IAS using state-of-the-art tools, frameworks, and libraries. The module accordingly starts with an introduction to according software frameworks and packages. It then introduces fundamental concepts from different building blocks of IAS, namely (a) machine perception, e.g., object detection and recognition, (b) world modelling, e.g., Simultaneous Localization and Mapping (SLAM) and map semantics, (c) navigation, e.g., obstacle avoidance and path planning, and (d) manipulation, e.g., motion planning and grasping. Finally, the students learn to perform system integration, i.e., to combine software components of the different fundamental building blocks in an application-oriented scenario of modern AI.

Recommended Knowledge

Students are expected to be familiar with programming in C/C++. They should have a good mathematical foundation, especially with respect to Linear Algebra and the foundations of optimization.

Intended Learning Outcomes

No	Competence	ILO
1	Describe	Describe use-cases of AI in a system-oriented way.
2	Use	Use IAS software tools, frameworks, and libraries.
3	Assess	Assess which AI software components are needed to conduct a given complex task in an intelligent autonomous way by a machine.
4	Explain	Explain the fundamental concepts and algorithms of core building blocks, namely machine perception, world modelling, navigation, and manipulation.
5	Recommend	Recommend software architectures for system-oriented AI applications.
6	Integrate	Integrate IAS software components in an application scenario.

Indicative Literature

- Robin R. Murphy: Introduction to AI Robotics, Bradford Books, 2019.
- Steven L. Brunton, J. Nathan Kutz: Data-Driven Science and Engineering, Cambridge University Press, 2019.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Intelligent Autonomous Systems	Written Examination	120 Minutes	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3.3 Symbolic Artificial Intelligence

Module Name	Symbolic Artificial Intelligence
Module Code	MCSSE-AI-03
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Francesco Maurelli Prof. Dr. Jakob Suchan

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	70
Exam Preparation	20
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Symbolic Artificial Intelligence	MCSSE-AI-03	Lecture	5

Module Description

This module deals with what is often called classical AI, i.e., especially formal methods based on symbolic representations. The module starts with an introduction to the history of AI research and the role of formal methods and symbolic representations. In doing so, its relation to other areas of AI, especially modern also known as nouvelle AI or Intelligent Autonomous Systems as well as Machine Learning including Artificial Neural Networks or sub-symbolic AI is explained. The presentation of specific methods starts with a discussion of problem-solving as search. It is followed by an introduction to knowledge representation, reasoning, and planning using classical Boolean and first order logic. The concepts and methods of Fuzzy Logic to deal with uncertain knowledge are then presented. Afterwards, probabilistic representations and reasoning methods are introduced. This is followed by a discussion of Multi-Agent-Systems (MAS) and related methods for, e.g., cooperation and coordination. Finally, it is shown how classical methods and representations are also increasingly used on a conceptual level within other AI areas, e.g., in form of explainable AI (exAI) to make the application-specific inner-workings and decision-making processes of (deep) neural networks more comprehensible for users to enable higher reliability and generality. Throughout the module, hands-on elements are used to make the students familiar with existing software approaches and libraries of classical AI plus their integration in general AI systems including hybrid approaches and the related software architectures.

Intended Learning Outcomes

No	Competence	ILO
1	Describe	Describe the different areas of AI and their conceptual relations to each other.
2	Explain	Explain the use of search algorithms for problem-solving.
3	Use	Use logic for representation, reasoning, and planning.
4	Implement	Implement and integrate fuzzy logic representation and reasoning.
5	Use	Use probabilistic knowledge representation, reasoning, and planning.
6	Explain	Explain core concepts and methods of Multi-Agent-Systems.
7	Assess	Assess which classical AI concepts and methods are useful and applicable components for a given application-oriented system.
8	Integrate	Integrate classical AI software components into hybrid AI systems.

Indicative Literature

- Peter Norvig, Stuart Russell: Artificial Intelligence, A Modern Approach, Pearson, 2021.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Symbolic Artificial Intelligence	Written Examination	120 Minutes	100	45%	1-8

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3.4 Text Analysis and Natural Language Processing

Module Name	Text Analysis and Natural Language Processing
Module Code	MDSSB-MET-02
Module ECTS	5
Program Owner	DSSB-MSc (Data Science for Society and Business)
Module Coordinator	Prof. Dr. Hilke Brockmann Prof. Dr. Jan Lorenz Prof. Dr. Adalbert F.X. Wilhelm

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective
DE-MSc Data Engineering	2	Mandatory Elective
DSSB-MSc Data Science for Society and Business	2	Mandatory
MBA-120-MA MBA 120	2	Mandatory Elective

Student Workload	
Seminar	17.5
Laboratory	17.5
Independent Study	90
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Text Analysis and Natural Language Processing	MDSSB-MET-02	Seminar	5

Module Description

This module will teach the fundamentals of text mining, natural language processing, and automated content analysis using R. Students will learn the entire text analysis pipeline, from basic web scraping techniques for collecting text data from social media, over text representations and ontologies, to text mining algorithms and efficient representation of analysis results. Students will be exposed to theoretical and methodological foundations of text mining, such as word frequencies, ontologies, bag-of-word, as well as the application of machine learning algorithms for text and sentiment analysis. The module will introduce exemplary studies on text and sentiment analysis and provide an opportunity for hands-on programming to realize different analyses. The module covers a spectrum of text mining methods, from basic lexicographic measures to more complex statistical learning algorithms such as sentiment analysis and topic modeling.

Recommended Knowledge

Programming skills in R or Python at an intermediate level

Usability and Relationship to other Modules

This module translates the insights from "Data Science Concepts" into text analysis. The module lays the basis for core and elective modules in semester 2 and 3, particularly for the "Digital Public Spheres," "Data Science Lab," "Data Analytics," and "Cybercriminology" modules.

Intended Learning Outcomes

No	Competence	ILO
1	Explain	Explain the concept of "text as data".
2	Use	Use basic methods for information extraction and text data retrieval.
3	Process	Process and prepare text data for statistical modeling and automated content analysis.
4	Perform	Perform different text analyses using text mining packages in R.
5	Interpret	Interpret diverse text analytical measures.
6	Undertake	Undertake a knowledgeable automated content analysis with text data.

Indicative Literature

- Silge, Robinson (2017) Text Mining with R: A Tidy Approach. Sebastopol, CA: O'Reilly.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Text Analysis and Natural Language Processing	Project Report	3000 words	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3.5 Data Analytics

Module Name	Data Analytics
Module Code	MDE-CO-02
Module ECTS	5
Program Owner	DE-MSc (Data Engineering)
Module Coordinator	Prof. Dr. Adalbert F.X. Wilhelm

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective
DE-MSc Data Engineering	1	Mandatory
DSSB-MSc Data Science for Society and Business	1	Mandatory Elective
MBA-120-MA MBA 120	1	Mandatory Elective
MBA-60-MA MBA 60	1	Mandatory Elective
MDDA-BSc Management, Decisions and Data Analytics	5	Mandatory Elective

Student Workload	
Independent Study	90
Lecture	17.5
Tutorial	17.5
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Data Analytics	MDE-CO-02	Lecture	5

Module Description

This module introduces concepts and methods of data analytics. The objective of the module is to present methods for gaining insight from data and drawing conclusions for analytical reasoning and decision-making. The module comprises a broad spectrum of methods for modelling and understanding complex datasets. Comprising both descriptive and predictive analytics, the standard portfolio of supervised and unsupervised learning techniques is introduced. Automatic analysis components, such as data transformation, aggregation, classification, clustering, and outlier detection, will be treated as an integral part of the analytics process.

As a central part of this module, students are introduced to the major concepts of statistical learning such as cross-validation, feature selection, and model evaluation. The course takes an applied approach and combines the theoretical foundation of data analytics with practical exposure to the data analysis process.

Recommended Knowledge

- Read the Syllabus.

- Take the free online course: Introduction to Data Science at https://cognitiveclass.ai/courses/data-science-101/

Usability and Relationship to other Modules

In this module students will learn concepts and various techniques for data analysis. They will be rigorously applied in MDE-CS-03 as well as in the applied projects MDE-DIS-02 and MDE-DIS-03, and typically also in the master thesis.

Intended Learning Outcomes

No	Competence	ILO
1	Explain	Explain advanced data analytics techniques in theory and application.
2	Apply	Apply data analytics methods to real-life problems using appropriate tools.
3	Evaluate	Evaluate and compare different data analytics algorithms and approaches.
4	Apply	Apply statistical concepts to evaluate data analytics results.

Indicative Literature

- A. Telea, Data Visualization: Principles and Practice, Wellesley, Mass.: AK Peters, 1st edition, 2008.(DV).
- G. James, D.Witten, T. Hastie, Rob Tibshirani: Introduction to Statistical Learning with R by Springer, 2013 (ISLR).
- M. Ward, G. Grinstein, D. Keim, Interactive Data Visualization: Foundations, Techniques, and Applications. AK Peters, 1st edition, 2010. (IDV)

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Data Analytics	Project Report	20 Pages	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.3.6 Machine Learning

Module Name	Machine Learning
Module Code	MDE-CO-04
Module ECTS	5
Program Owner	DE-MSc (Data Engineering)
Module Coordinator	Prof. Dr. Markus Wenzel

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective
DE-MSc Data Engineering	2	Mandatory
DSSB-MSc Data Science for Society and Business	2	Mandatory Elective
MBA-120-MA MBA 120	2	Mandatory Elective

Student Workload	
Lecture	35
Independent Study	90
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Machine Learning	MDE-CO-04	Lecture	5

Module Description

Machine learning (ML) is a module that concerns algorithms that are fed with (large quantities of) real-world data, and which returns a compressed "model" of the data. An example is the "world model" of a robot: the input data are sensor data streams, from which the robot learns a model of its environment. Another example is a spoken language model: the input data are speech recordings, from which ML methods build a model of spoken English -- useful, for instance, in automated speech recognition systems. There are many formalisms in which such models can be cast, and an equally large diversity of learning algorithms. At the same time, there is a relatively small number of fundamental challenges that are common to all these formalisms and algorithms.

The module introduces such fundamental concepts and illustrates them with a choice of elementary model formalisms (linear classifiers and regressors, radial basis function networks, clustering, neural networks). Furthermore, the module also (re)introduces required mathematical material from probability theory and linear algebra. The main educational aims are twofold: to make students fully aware of the two main hurdles for obtaining good models from data: (i) the "curse of dimensionality" and (ii) the bias-variance dilemma and to provide standard tools to cope with these difficulties, namely

(i) dimension reduction by feature extraction, for example via PCA or clustering, and (ii) cross-validation and regularization.

Usability and Relationship to other Modules

This module is a natural companion to the "Principles of Statistical Modeling" (PSM) module MDE-CS-03.

The ML module focuses on practical ML skills, whereas PSM module on rigorous mathematical formalism and analysis.

For students not familiar with graph theory, it is recommended to take the first semester course MDE-CS-01 Network

Theory, which introduces concepts used in this Machine Learning module.

Recommended Knowledge

- Basic linear algebra, calculus and probability theory, as typically acquired in entry modules in BSc studies.

- Read the syllabus.

- Highly recommended: Mitchell, Tom M.: Machine Learning (McGraw-Hill, 1997) IRC: Q325.5.M58 1997. This standard, classical textbook gives a very accessible overview of ML.

Intended Learning Outcomes

No	Competence	ILO
1	Design	Design, implement and exploit elementary supervised ML methods for classification and regression with expert care given to dimension reduction preprocessing and regularization
2	Understand	Understand and practically use PCA and linear regression
3	Understand	Understand the core ideas behind feedforward neural networks and the backpropagation algorithm, as the basis for accessing "deep learning" methods

Indicative Literature

- T. M. Mitchel, Machine Learning, McGraw-Hill, 1997, IRC: Q325.5.M58.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Machine Learning	Project Report	10 pages	100	45%	1-3

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.4 Breakthrough Area Modules

4.4.1 Quantum Informatics

Module Name	Quantum Informatics
Module Code	MDE-QT-02
Module ECTS	5
Program Owner	DE-MSc (Data Engineering)
Module Coordinator	Prof. Dr. Wolfgang Tittel

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective
DE-MSc Data Engineering	2	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective
MMDA-BSc Mathematics, Modeling, and Data Analytics	6	Mandatory Elective
PHDS-BSc Physics and Data Science	6	Mandatory Elective

Student Workload	
Independent Study	90
Laboratory/Precepts	17.5
Lecture	17.5
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Quantum Informatics Lab	MDE-QT-02-B	Laboratory	2.5
Quantum Informatics	MDE-QT-02-A	Lecture	2.5

Module Description

The course introduces central topics in quantum communication and quantum computing such as:

- Quantum bits and their representation using state vectors and density matrices

- Quantum measurements, quantum gates and quantum circuit diagrams

- The no-cloning theorem and optical quantum cloning machines

- Entanglement, its roles as a fundamental property of nature and as a resource for quantum technology and the Bell inequality

- Quantum key distribution and the impact of eavesdropping

- Quantum teleportation, entanglement swapping and quantum repeaters

- Entanglement distillation and quantum error correction

- Simple quantum computing algorithms (Deutsch–Jozsa, Shor)

Recommended Knowledge

- Basic linear algebra, complex numbers

- Introductory texts on quantum mechanics, quantum information and quantum computing; review of vectors and matrices

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand the underlying concepts of quantum communication and computation.
2	Know	Know how to read and present related research papers and textbook material.
3	Solve	Solve simple problems based on quantum mechanical aspects such as superposition and entanglement.

Indicative Literature

- Extensive lecture notes and other material used during the lecture and the exercise sessions will be provided during the course.
- Michael A. Nielsen, Isaac L. Chuang: Quantum Computation and Quantum Information (10th Anniversary Edition), Cambridge University Press, 2010.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Quantum Informatics Lab	Portfolio Assessment	-Quizzes (10x), counting 50% towards the final grade	50	45%	

		-in-class presentation of research papers or parts of a textbook, counting 50% towards the final grade			
Quantum Informatics	Written Examination	120 Minutes	50	45%	All ILOs (focus on theory) .

Module Completion Requirements

- **Module Component Examination:** minimum for pass- 45%.

Module Achievement

None.

4.5 Management Modules

4.5.1 Agile Product Development & Design

Module Name	Agile Product Development & Design
Module Code	MCSSE-MGT-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory
AST-MSc Advanced Software Technology	3	Mandatory

Student Workload	
Lecture	80
Independent Study	45
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Agile Product Development & Design	MCSSE-MGT-01	Lecture	5

Module Description

This course is focused on key aspects of agile product and service development and design process.

State-of-the-art user centered design methods will be at the core of the course.

The overall goal of this module is to help managers without a business degree to learn, understand and practice agile customer- and data-driven innovation processes in the information age. This module helps students to understand today's real-life challenges in a complex world, with wicked problems and with multiple stakeholder interests, where unpredictable is common, and where managers need to focus on achieving goals rather than repetitive tasks.

Students learn to develop and present innovative user-centered and theory-oriented solutions for real-world challenges in an IT-driven world.

This course is strongly based on the agile paradigm of user-centeredness, user-centered design and the ideas of the Service Dominant Logic. Service-dominant (S-D) logic is a meta-theoretical framework for explaining value co-creation, through exchange, among configurations of actors.

Major challenges and concerns will be reflected:

- the role of the customer and data in a transformed business world

- new theories, concepts, and approaches (such as service dominant logic, customer integration, gamification, new service models)

- new methods and management techniques in (service) innovation (Design Thinking)

- new methods in handling business processes: (agile) business process management - BPM

- ethics and security issues.

The module will enable students to collaborate across disciplines with experts from various areas.

Intended Learning Outcomes

No	Competence	ILO
1	Develop	Develop practical knowledge and management skills, and mind sets to master the challenges from an agile business environment.
2	Understand	Understand (routine) business processes in various context and how to adapt business processes to an agile business environment (agile Business Process Management).
3	Summarize	Summarize and classify the new data- and customer-driven technologies in a business context.
4	Understand	Understand the ideas of the "service dominant logic" as a business opportunity, such as user-centricity, value in use, value in interaction, business service ecosystems.
5	Apply	Apply innovative creativity methods and processes for product and software development (Design Thinking).
6	Adapt	Adapt to a new working culture based on a user-centricity, empathy, and playful testing of new products and services.

Indicative Literature

- Brenner, W., Uebernickel, F., Abrell, T. (2016). Design Thinking as Mindset, Process, and Toolbox, in: Brenner, W., Uebernickel, F. (Eds.), Design Thinking for Innovation. Springer International Publishing, pp. 3–21. https://doi.org/10.1007/978-3-319-26100-3_1. Brown, T. (2008). Design Thinking. Harvard Business Review. 86, 84–92. Available at: https://hbr.org/2008/06/design-thinking.
- Daniel Paschek, D., Frank Rennung, F., Trusculescu, A., Draghici,A. (2016). Corporate Development with Agile Business Process Modeling as a Key Success Factor, Procedia Computer Science, Vol 100, Pages 1168-1175, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2016.09.273.
- Vargo, S.L., & Lusch, R. (2004). Evolving to a New Dominant Logic for Marketing. Journal of Marketing, Vol. 68(1), 1 – 17. Vargo SL, Akaka MA, Vaughan CM. (2017). Conceptualizing Value: A Service-ecosystem View. Journal of Creating Value. 3(2):117-124. https://doi.org/10.1177%2F2394964317732861. Lusch, R.F., Nambisan, S. (2015). Service Innovation: A Service-Dominant Logic Perspective. MIS Quarterly. Vol. 39 No.1 , pp. 155-175. https://doi.org/10.25300/MISQ/2015/39.1.07.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Agile Product Development & Design	Presentation	30 minutes	100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.5.2 Product Innovation & Marketing

Module Name	Product Innovation & Marketing
Module Code	MCSSE-MGT-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Dr. PingPing Meckel

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	2	Mandatory
CSSE-MSc Computer Science & Software Engineering	2	Mandatory

Student Workload	
Lecture	80
Independent Study	45
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Product Innovation & Marketing	MCSSE-MGT-02	Lecture	5

Module Description

This course focuses on key strategic aspects of the innovation and commercialization process. The course draws on insights from a variety of fields – in particular, product management, innovation, marketing, and strategic management – in order to (i) develop a holistic, state-of-the art understanding of this process, (ii) to nurture the underlying mindset that spans technology and market elements, and (iii) to provide students with concrete tools that help them in navigating the journey from product idea to market success. The course will take both the perspective of established companies as well as of new ventures.

Intended Learning Outcomes

No	**Competence**	**ILO**
1	Understand	Understand the innovation process, particularly in technology domains.
2	Understand	Understand the commercialization process, particularly in technology domains.
3	Analyze	Analyze how value can be created and appropriated through innovation.

4	Understand	Understand and apply tools, methods and concepts to manage the commercialization process.

Indicative Literature

- Kotler, P. et al. (2024). Principles of Marketing, Global Edition. 19th ed. Harlow: Pearson Education Limited.
- Schilling, M.A. (2019). Strategic Management of Technological Innovation. McGraw-Hill.
- Tidd, J. and Bessant, J. (2021). Managing Innovation: Integrating Technological, Market and Organizational Change. 7th ed. Hoboken: Wiley.

Entry Requirements

Prerequisites	MCSSE-LAS-01 Entrepreneurship and Intrapreneurship
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Product Innovation & Marketing	Presentation	30 minutes	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.5.3 Transformational Change Management

Module Name	Transformational Change Management
Module Code	MBA-LS-03
Module ECTS	5
Program Owner	MBA-120-MA (MBA 120)
Module Coordinator	Prof. Dr. Sohaib Hassan

Study Semester		
Program	**Semester**	**Status**
MBA-60-MA MBA 60	1	Mandatory
CSSE-MSc Computer Science & Software Engineering	3	Mandatory
MBA-120-MA MBA 120	3	Mandatory

Student Workload	
Lecture	80
Independent Study	45
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Transformational Change Management	MBA-LS-03-A	Lecture	5

Module Description

Change is part of every successful manager's and organization's life. Thus, learning to lead change and/or be part of a successful change effort, is essential for anyone who hopes to rise from being an individual contributor. Some change efforts have no impact whatsoever; the organization is neither better nor worse afterwards. This is a waste of human capital (and probably financial capital as well). Some change efforts work for a while, but then gravity takes over and the organization returns to where it was beforehand; again, a waste. And there are other change projects that get us to a new level, and we stay there, which is not bad; a vast improvement on the previous two situations. But what we all want, and what this course will focus on, is to change an organization in some way, and put it on a continuous upward trajectory.

That is transformation. To build this understanding, the courses deals with the following topics:

- Change management models

- Influencing styles and tactics

- Communicating well in a group

- Understanding your biases

- Seeing and understanding different leadership styles in company transformations

- Stakeholder management

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand, evaluate, and apply different leadership styles.
2	Understand	Understand and evaluate the change process in organizations.
3	Understand	Understand and apply communications and influencing.
4	Evaluate	Evaluate their role in a change situation.
5	Assess	Assess the stakeholders in any change context.
6	Lead	Lead or be part of an organizational change effort.

Indicative Literature

- Daniel Goleman, HBR, 2002, Leadership that gets results.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Transformational Change Management	Presentation	30 minutes	100	45%	All

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6 Leadership/ Academic Skills Modules

4.6.1 Entrepreneurship and Intrapreneurship

Module Name	Entrepreneurship and Intrapreneurship
Module Code	MCSSE-LAS-01
Module ECTS	2.5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Dr. PingPing Meckel

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory
CSSE-MSc Computer Science & Software Engineering	1	Mandatory

Student Workload	
Lecture	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Entrepreneurship and Intrapreneurship	MCSSE-LAS-01	Lecture	2.5

Module Description

The module introduces students to the themes which are relevant to clearly develop corporate innovation and entrepreneurship as an activity. It introduces entrepreneurial thinking styles that are important to develop radical forms of innovation in companies. This is about a way of thinking, reasoning and acting that is opportunity obsessed and holistic in approach. It is first and foremost a process that has an intention to create, enhance, realize, and renew value, not just for owners, but for all participants and stakeholders in either a new or existing organization. Today, entrepreneurship has evolved beyond the classic start-up notion to include companies and organizations of all types, old and new; small and large; fast and slow growing; private, not-for-profit, and public.

This focus on “entrepreneurship as a process” has become a fundamental part for three main reasons. The first is the growing recognition of the critical importance of entrepreneurial activities in the economy and the society at large. As such, having an insight in the specific challenges and solutions that characterize entrepreneurship has broader implications for any 21st century graduate. The second reason is that many graduates eventually find themselves occupying a position as entrepreneur, or are associated with one as their financier, partner, supplier or customer. This requires an action-oriented approach and approaching the phenomenon from multiple angles. Finally, given the specific challenges

entrepreneurs often face in terms of uncertainty and resource scarcity, solutions applied by expert entrepreneurs can be of value to any professional that finds him/herself in similar situations in organizations seeking growth, renewal or even survival.

The module focuses on the tasks and skills that entrepreneurs typically complete/use in their journey towards success. With this in mind, this module aims to provide students with insight into the approach entrepreneurs use to identify opportunities and build new ventures; the analytical skills that are needed to implement this approach; and the background knowledge and managerial skills that are needed for dealing with issues involved in starting, growing, and harnessing the value of new ventures. First and foremost, however, entrepreneurship is about action. Hence our approach is based on the primary objective of having students experience entrepreneurship.

This module is intentionally designed as a 2.5 CP module to enhance Employability by exposing students to a variety of professionally relevant topics rather than requiring in-depth specialization.

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand the essence of entrepreneurship.
2	Assess	Assess and develop a business case.
3	Analyze	Analyze and identify new venture opportunities in a more systematic way.
4	Understand	Understand the importance of a business model for new venture creation.
5	Evaluate	Evaluate the viability of a new venture idea.
6	Understand	Understand how to finance a new venture.
7	Create	Create and present a business case for a new venture.

Indicative Literature

- Greene, F. J. (2020). Entrepreneurship: Theory and Practice. London: Macmillan Education Ltd.
- Jones, O., Meckel, P., and Taylor, D. (2021). Creating Communities of Practice: Entrepreneurial Learning in a University-Based Incubator. Cham: Springer International Publishing AG. https://ebookcentral.proquest.com/lib/constructor-university/detail.action?docID=6467881.
- Rae, D. (2015). Opportunity-Centred Entrepreneurship. 2nd ed. London: Palgrave.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Entrepreneurship and Intrapreneurship	Presentation	30 minutes	100	45%	1-7

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6.2 Communication & Presentation Skills for Executives

Module Name	Communication & Presentation Skills for Executives
Module Code	MBA-NC-08
Module ECTS	2.5
Program Owner	MBA-120-MA (MBA 120)
Module Coordinator	Prof. Dr. Markus Wenzel

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory
DE-MSc Data Engineering	1	Mandatory
DSSB-MSc Data Science for Society and Business	1	Mandatory

Student Workload	
Seminar	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Communication & Presentation Skills for Executives	MBA-NC-01-A	Seminar	2.5

Module Description

An executive career in an international business environment requires excellent communication and presentation skills. Managers have to communicate effectively with a large variety of target audiences, often in different languages and with different cultural backgrounds. This is true for employees and/or direct reports, business partners as well as customers. The ability to present and communicate succinctly and confidently while being culturally aware and building rapport and trust with different audiences is crucial. In this interactive module, students are introduced to the basics of effective presentation and communication techniques. They learn how to present themselves, their business project, or academic work, with impact, tailoring both the content and their delivery style to different types of audiences.

This module is intentionally designed as a 2.5 CP module to enhance Employability by exposing students to a variety of professionally relevant topics rather than requiring in-depth specialization

Recommended Knowledge

- Analysis, Basic Calculus, and Linear Algebra

- Read the Syllabus

Intended Learning Outcomes

No	Competence	ILO
1	Act	Act as effective communicators – in both group and individual situations.
2	Understand	Understand interpersonal communication models and group dynamics in presentations.
3	Enjoy	Enjoy the process of presenting.
4	Understand	Understand the importance of building rapport and trust with audiences.
5	Use	Use presentation software (PowerPoint, Prezi) confidently and in a visually pleasant way.
6	Learn	Learn how to structure presentations in a coherent manner and develop captivating narratives.
7	Work	Work with different presentation formats (Ignite, Pecha Kucha, Pitching etc.).
8	Understand	Understand and apply the basics of logical reasoning in oratory (deductive/inductive).
9	Develop	Develop oratory and rhetorical skills drawing on Aristotle's teaching of logos, ethos and pathos.
10	Understand	Understand and apply the basics of interpersonal communication (Johari Window, 4-Ears model etc.).
11	Give	Give and receive constructive feedback.
12	Present	Present themselves in different business situations.
13	Collaborate	Collaborate effective in intercultural teams.

Indicative Literature

- This course utilizes lecture formats, case studies and interactive presentations, discussions, role play and peer-to-peer coaching. The course will also use internet resources, videos, and home assignments to illustrate and practice specific communication aspects.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Communication & Presentation Skills for Executives	Presentation	15 minutes	100	45%	1-13

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6.3 Organizational Behavior

Module Name	Organizational Behavior
Module Code	MBA-NC-09
Module ECTS	2.5
Program Owner	MBA-120-MA (MBA 120)
Module Coordinator	Prof. Dr. Christian Stamov Roßnagel

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory

Student Workload	
Lecture	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Organizational Behavior	MBA-NC-02-A	Lecture	2.5

Module Description

Geared towards improving an organization's effectiveness, organizational behavior (OB) focuses on the impact of people, groups, and organizational structures on work-related behavior within organizations. OB research findings help align personal and organizational needs in selecting, placing, and developing people in organizations. In the face of the current '3D' megatrends of digitalization, diversity, and demographic change, companies' demand for OB solutions is greater than ever. For a thorough understanding of the principles governing OB, you will build a generic model of the multilevel interactions between parameters on the individual, group, and organizational levels, and how those relate to individual and organizational productivity. From this comprehensive model, you will derive actionable guidelines for personnel selection, performance management, and leadership and apply them to addressing leadership and management challenges in selected business case examples. This module is intended to help you acquire the background to analyses and structure organizations in an evidence-based 21st -century manner.

The module assessment will consist of three presentations. Students will know in the first session which topics need to be covered in their presentations.

This module is intentionally designed as a 2.5 CP module to enhance Employability by exposing students to a variety of professionally relevant topics rather than requiring in-depth specialization.

Intended Learning Outcomes

No	Competence	ILO
1	Explain	Explain basic principles of individuals' and groups' behaviours in organisations.
2	Apply	Apply established theories to assessing and predicting behaviour.
3	Describe	Describe core techniques of influencing and modifying behaviour.
4	Discuss	Discuss critically selected approaches to effectively lead employees, teams, and groups.

Indicative Literature

- King, D., & Lawley, S. (2019). Organizational Behaviour (3rd ed.). Oxford University Press.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Organizational Behavior	Presentation	30 minutes	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6.4 Academic Writing Skills/Intercultural Training

Module Name	Academic Writing Skills/Intercultural Training
Module Code	MDE-CAR-02
Module ECTS	2.5
Program Owner	DE-MSc (Data Engineering)
Module Coordinator	Prof. Dr. Markus Wenzel

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory
DE-MSc Data Engineering	2	Mandatory

Student Workload	
Lecture	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Academic Writing Skills/Intercultural Training	MDE-CAR-02	Seminar	2.5

Module Description

The academically rigorous nature of graduate studies requires students to master academic writing skills and techniques. In this introductory course, students in DE master's program will learn the foundations of academic writing at a graduate level, with special focus on writing academic essays, identifying organizational patterns of academic texts, and formulating arguments to produce cohesive and coherent academic papers. Through the process of drafting, continuous feedback and editing, students will improve their writing skills. This course will also help students develop their research skills by highlighting techniques of finding and evaluating sources, and utilizing citation and referencing styles. As graduate students, adhering to The Code of Academic Integrity is a requirement. Hence, this course will incorporate a session on scholarly and intellectual standards set by Constructor University. The second part of this course is a training seminar. It will give answers to frequently asked questions by students on the topics of working and living in Germany. Here the students will find information on employment and how to get access to the German labor market. The seminar also provides an overview of labor conditions in Germany, the multifaceted forms of employment, business cultures and useful tips and information for the job entry in a German company.

This module is intentionally designed as a 2.5 CP module to enhance Employability by exposing students to a variety of professionally relevant topics rather than requiring in-depth specialization

Recommended Knowledge

- Read the Syllabus.

- Fraedrich, J. & Ferrell, O.C. (2014): Business Ethics: Ethical Decision Making & Cases. Cengage Learning.

Usability and Relationship to other Modules

For DE: Advanced Project 1, Advanced Project 2, Master thesis

Intended Learning Outcomes

No	Competence	ILO
1	Structure	Structure their ideas to write clear summaries, coherent paragraphs and cohesive literature reviews.
2	Write	Write different segments of an academic paper employing writing styles that display advanced grammar and precise and concise language use.
3	Find	Find and evaluate sources for research.
4	Use	Use citation and referencing styles applicable for their discipline.
5		Avoid unintentional plagiarism and adhere to the code of academic integrity.
6	Understand	Understand labor conditions in Germany.
7	Understand	Understand the typical business cultures in German companies.

Indicative Literature

- The literature is provided individually to each student by each instructor for the respective advanced project.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Academic Writing Skills/Intercultural Training	Term Paper	10 pages	100	45%	1-7

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6.5 Agile Leadership and Strategic Management

Module Name	Agile Leadership and Strategic Management
Module Code	MBA-LS-01
Module ECTS	2.5
Program Owner	MBA-120-MA (MBA 120)
Module Coordinator	Prof. Dr. Adalbert F.X. Wilhelm

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory
MBA-120-MA MBA 120	1	Mandatory
CSSE-MSc Computer Science & Software Engineering	3	Mandatory

Student Workload	
Lecture	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Agile Leadership and Strategic Management	MBA-LS-01-A	Lecture	2.5

Module Description

This module focuses on key strategic aspects of the leadership and strategy development processes, specifically strategic problems solving, alignment, engagement and coping with black swans and paradigm shifts. The module draws insights from a variety of fields such as business strategy, problem solving, strategic communication, strategic planning, and strategic resilience.

To build a holistic understanding, the module deals with the following topics:

- The strategic process: from analysis, definition, planning and evaluation

- Hypothesis driven problem solving

- Pyramid principle strategic communication

- Antifragile strategies

The module assessment will consist of three presentations. Students will know in the first session which topics need to be covered in their presentations.

This module is intentionally designed as a 2.5 CP module to enhance Employability by exposing students to a variety of professionally relevant topics rather than requiring in-depth specialization

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understand and analyze business strategies.
2	Understand	Understand and analyze strategic statements and levels of ambition.
3	Understand	Understand opportunities and threats on the external environment.
4	Evaluate	Evaluate sources of competitive advantage as well as strategic strengths and weaknesses.
5	Analyze	Analyze core challenges of agile leadership and strategy development.
6	Develop	Develop and communicate strategic initiatives.
7	Apply	Apply this knowledge to real-world strategic planning processes.

Indicative Literature

- Sola, D. & Couturier, J, 2013, How To Think Strategically, FT Publishing International.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Agile Leadership and Strategic Management	Presentation	30 minutes	100	45%	1-7

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.6.6 Customer-centric Mindset and Agile Delivery Management

Module Name	Customer-centric Mindset and Agile Delivery Management
Module Code	MBA-LS-02
Module ECTS	2.5
Program Owner	MBA-120-MA (MBA 120)
Module Coordinator	Prof. Dr. Sohaib Hassan

Study Semester		
Program	**Semester**	**Status**
MBA-120-MA MBA 120	1	Mandatory
CSSE-MSc Computer Science & Software Engineering	3	Mandatory

Student Workload	
Lecture	17.5
Independent Study	45
Total Hours	**62.5**

Module Components	**Number**	**Type**	**CP**
Customer-centric Mindset and Agile Delivery Management	MBA-LS-02-A	Lecture	2.5

Module Description

Successful firms are forced to walk a tightwire between meeting market needs and creating organizational efficiencies. Just how they do this requires, organization, insights, management understanding and determination. The modern manufacturing or service firm is simultaneously engaged in three core processes. 1) The design and development of products and services (BUILD), 2) The efficient and effective delivery of those products and services to the market (DELIVER), and 3) The process of gaining customers that wish to purchase those products and services or enter into transactions with the firm (CAPTURE). How it organizes and the processes it adopts are key to a firm´s ability to optimize these often divergent but highly interdependent activities.

While these three processes are often at odds with each other, this module will inform, challenge, and enlighten the participants on a) The best practices in each of these areas, b) The ways to improve their understanding and implementation of course concepts, and c) The trends that they will invariably deal with in the near future. In this module, students touch upon the design of innovative R&D, operations, and marketing strategies that provide firms with a strategic and sustainable competitive advantage that is capable of utilizing global resources and capturing markets. These strategies will constantly be viewed in a competitive, resource constrained, and capital efficient marketplace.

The module assessment will consist of three presentations. Students will know in the first session which topics need to be covered in their presentations.

This 2.5 CP module supports Academic Excellence by providing essential contextual knowledge that underpins the program's qualification aims, while appropriately limiting academic depth relative to the core technical competencies of the field.

Intended Learning Outcomes

No	Competence	ILO
1	Analyze	Analyze critically the task of going to market under contemporary conditions and to examine the major functions that comprise the marketing servicing task.
2	Evaluate	Evaluate various types of policies that can be employed in guiding market centric activities.
3	Develop	Develop an awareness of the major types of market problems faced by organizations, with emphasis on sound analytical approaches to effective problem-solving decisions.
4	Analyze	Analyze different business models and understand how the marketing function can be employed to enhance them.

Indicative Literature

- Chernev, A., 2018, Strategic Marketing Management.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Customer-centric Mindset and Agile Delivery Management	Presentation	30 minutes	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7 Research Project, Capstone Project & Master Thesis

4.7.1 Research Project

Module Name	Research Project
Module Code	MCSSE-RP-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Research Group Meetings	21
Project (Independent Work)	104
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Research Project	MCSSE-RP-01	Project	5

Module Description

The competencies and knowledge earned in the first two semesters are deepened by developing a small research project. Students will be exposed to state-of-the-art research with the goal of reproducing results of recent research papers or extending ideas presented in recent research papers. Students will learn how to organize and execute a research project and how to present the results in the format of a typical research paper. Students are expected to participate in the meetings of the research group in which they are doing their research projects.

Intended Learning Outcomes

No	**Competence**	**ILO**
1	Understand	Understand state-of-the-art research papers in a chosen field of specialization.
2	Plan	Plan a research project to reproduce research results or to extend ideas of recent research results.
3	Explain	Explain research questions and choose suitable methodologies to address them.
4	Document	Document a research project in the style of a typical scientific paper.

Indicative Literature

- Recent publications provided by the research project supervisors.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Research Project	Project Report	5000 words	100	45%	1-4

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.2 Capstone Project I

Module Name	Capstone Project I
Module Code	MCSSE-CAP-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Manuel Oriol

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	1	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective

Student Workload	
Lecture	35
Tutorial	35
Project (Group-Based and Independent Work)	55
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Capstone Project 1	MCSSE-CAP-01	Project	5

Module Description

This series of Capstone modules gives the possibility of experiencing knowledge and expertise learned in the master by a posteriori analysis, transformational adaptation and coherent planning hands-on practice. The series spans over three modules during which students develop a complete product from scratch. The project starts with an ideation process, creation of clickable demos and initial requirements. It continues with the practical creation of a software architecture and development of the solution. It then finishes with application of artificial intelligence and cybersecurity. During the project, students are going through various steps during which they are encouraged to talk directly to potential real-world customers and users, thus gathering an understanding of what real users and customers for their project might want.

The project is organized in tribes (20-30 people) in charge of exactly one project. The tribes are then further split in agile teams working with the advice of the instructors and the assistants (impersonating the business owners and product owners). The teams can be geographically distributed and work with an up-to-date environment supported with open-source IDEs and engineering tools. Few lectures indicate the best practices to follow and the interim goals. Periodic meetings with the instructor and teaching assistants steer the process towards the overall goal.

This instance is the first semester of the Capstone project that focuses on ideation and requirements elicitation.

Recommended Knowledge

- Programming skills in an imperative language at CS bachelor level.

- Algorithms and data structure at CS bachelor level

- Train and advance programming, read about agile development, watch videos on ideation processes and read books on team and teamwork.

Usability and Relationship to other Modules

It is highly recommended to take the three Capstone Project modules in their numerical order to gain the full experience of the project.

Intended Learning Outcomes

No	Competence	ILO
1	Create	Create and propose mocks.
2	Perform	Perform requirements elicitation.
3	Prototype	Prototype
4	Approach	Approach customers and users.
5	Specify	Specify user stories.
6	Organize	Organize themselves through collaborative tools.
7	Understand	Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

- • Online resources on team dynamics: - https://www.challengeapplications.com/stages-of-team-development - https://agilescrumguide.com/blog/files/tag-5-stages-of-team-development.html
- Bertrand Meyer: Agile! The good, the Hype and the Ugly. Springer, 2024
- Patrick Lencioni: The Five Dysfunctions of a Team. Jossey-Bass, 2022.
- Timothy M. Franz: Group dynamics and Teams interventions. Wiley-Blackwell, 2012

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Capstone Project 1	Project Assessment		100	45%	1-7

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.3 Capstone Project II

Module Name	Capstone Project II
Module Code	MCSSE-CAP-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Manuel Oriol

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	2	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective

Student Workload	
Lecture	35
Tutorial	35
Project (Group-Based and Independent Work)	55
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Capstone Project II	MCSSE-CAP-02	Project	5

Module Description

This series of courses gives the possibility of experiencing knowledge and expertise learned in the master by a posteriori analysis, transformational adaptation and coherent planning hands-on practice. The course series spans over three courses during which students develop a complete product from scratch. The project starts with an ideation process, creation of clickable demos and initial requirements. It continues with the practical creation of a software architecture and development of the solution. It then finishes with application of artificial intelligence and cybersecurity. During the project students are going through various steps during which they are encouraged to talk directly to potential real-world customers and users, thus gathering an understanding of what real users and customers for their project might want.

The project is organized in tribes (20-30 people) in charge of exactly one project. The tribes are then further split in agile teams working with the advice of the instructors and the assistants (impersonating the business owners and product owners). The teams can be geographically distributed and work with an up-to-date environment supported with open source IDEs and engineering tools. Few lectures indicate the best practices to follow and the interim goals. Periodic meetings with the instructor and teaching assistants steer the process towards the overall goal.

This instance is the second semester of the capstone project that focuses on architecture and base implementation.

Recommended Knowledge

- Programming skills in an imperative language at CS bachelor level.

- Algorithms and data structure at CS bachelor level

- Train and advance programming, read about agile development, watch videos on ideation processes and read books on team and teamwork.

Usability and Relationship to other Modules

It is highly recommended to take the three Capstone Project modules in their numerical order to gain the full experience of the project.

Intended Learning Outcomes

No	Competence	ILO
1	Describe	Describe and defend a software architecture.
2	Code	Code in groups
3	Code	Code as a large team.
4	Integrate	Integrate independent works.
5	Use	Use a source code versioning system.
6	Specify	Specify user stories
7	Hold	Hold practical discussions with stakeholders.
8	Organize	Organize themselves through collaborative tools.
9	Understand	Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

- Bertrand Meyer: Agile! The good, the Hype and the Ugly. Springer, 2024
- Online resources on team dynamics: - https://www.challengeapplications.com/stages-of-team-development - https://agilescrumguide.com/blog/files/tag-5-stages-of-team-development.html
- Online resources on team dynamics: - https://www.challengeapplications.com/stages-of-team-development - https://agilescrumguide.com/blog/files/tag-5-stages-of-team-development.html
- Patrick Lencioni: The Five Dysfunctions of a Team. Jossey-Bass, 2022.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Capstone Project II	Project Assessment		100	45%	1-8

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.4 Capstone Project III

Module Name	Capstone Project III
Module Code	MCSSE-CAP-03
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Manuel Oriol

Study Semester		
Program	**Semester**	**Status**
AST-MSc Advanced Software Technology	3	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Lecture	35
Tutorial	35
Project (Group-Based and Independent Work)	55
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Capstone Project III	MCSSE-CAP-03	Project	5

Module Description

This series of courses gives the possibility of experiencing knowledge and expertise learned in the master by aposteriori analysis, transformational adaptation and coherent planning hands-on practice. The course series spans over three courses during which students develop a complete product from scratch. The project starts with an ideation process, creation of clickable demos and initial requirements. It continues with the practical creation of a software architecture and development of the solution. It then finishes with application of artificial intelligence and cybersecurity. During the project students are going through various steps during which they are encouraged to talk directly to potential real-world customers and users, thus gathering an understanding of what real users and customers for their project might want.

The project is organized in tribes (20-30 people) in charge of exactly one project. The tribes are then further split in agile teams working with the advice of the instructors and the assistants (impersonating the business owners and product owners). The teams can be geographically distributed and work with an up-to-date environment supported with open source IDEs and engineering tools. Few lectures indicate the best practices to follow and the interim goals. Periodic meetings with the instructor and teaching assistants steer the process towards the overall goal.

This instance is the third semester of the Capstone Project that focuses on integrating artificial intelligence, cybersecurity, and develops best practices.

Recommended Knowledge

- Programming skills in an imperative language at CS bachelor level

- Algorithms and data structure at CS bachelor level

- Train and advance programming, read about agile development, watch videos on ideation processes and read books on team and teamwork.

Usability and Relationship to other Modules

It is highly recommended to take the three Capstone Project modules in their numerical order to gain the full experience of the project.

Intended Learning Outcomes

No	Competence	ILO
1	Know	Know practical cybersecurity.
2	Hold	Hold practical discussions with stakeholders.
3	Practice	Practice of machine learning.
4	Work	Work with continuous improvements tools.
5	Organize	Organize themselves through collaborative tools.
6	Understand	Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

- Bertrand Meyer: Agile! The good, the Hype and the Ugly. Springer, 2024
- Online resources on team dynamics: - https://www.challengeapplications.com/stages-of-team-development - https://agilescrumguide.com/blog/files/tag-5-stages-of-team-development.html.
- Patrick Lencioni: The Five Dysfunctions of a Team. Jossey-Bass, 2022.
- Timothy M. Franz: Group dynamics and Teams interventions. Wiley-Blackwell, 2012

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs

Capstone Project III	Project Assessment		100	45%	1-6

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.5 Research Lab I

Module Name	Research Lab I
Module Code	MCSSE-RL-01
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder (SPC)

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	1	Mandatory Elective

Student Workload	
Laboratory	30
Project (Independent Work)	95
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Research Lab I	MCSSE-RL-01	Laboratory	5

Module Description

The Research Lab I is part of the research orientation of the Computer Science and Software Engineering program. Students select a research field and a supervisor among all participating faculty based on their knowledge and interest. Enrollment into the module requires that students have found a supervisor. In their chosen field (e.g., software engineering, artificial intelligence, cybersecurity) students will receive intense training on state-of-the-art problems, experimental procedures, software systems, algorithms, etc. from the members of the research group selected. Students will receive a small research project to work on, typically in a group setting. Throughout the project, students discuss their results with their peers and supervisors. Students summarize their findings in a lab report written in the style of typical computer science conference papers.

Intended Learning Outcomes

No	**Competence**	**ILO**
1	Plan	Plan a research project by identifying research questions and suitable methodologies.
2	Establish	Establish and configure software components necessary to conduct research experiments.
3	Conduct	Conduct experiments to produce data suitable to evaluate research ideas against known state-of-the-art solutions.
4	Analyze	Analyze and interpret collected data.

5	Present	Present research results in a concise research report and evaluates the significance of the results produced.

Indicative Literature

- Justin Zobel, Writing for Computer Science, 3rd edition, Springer, 2014 .

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Research Lab I	Laboratory Report	10 pages	100	45%	1-5

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.6 Research Lab II

Module Name	Research Lab II
Module Code	MCSSE-RL-02
Module ECTS	5
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder (SPC)

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective

Student Workload	
Laboratory	30
Project (Independent Work)	95
Total Hours	**125**

Module Components	**Number**	**Type**	**CP**
Research Lab II	MCCSE-RL-02	Laboratory	5

Module Description

Research Lab II is part of the research orientation of the Computer Science and Software Engineering program. Students select a research field and a supervisor among all participating faculty based on their knowledge and interest. Enrollment into the module requires that students have found a supervisor. In their chosen field (e.g. software engineering, artificial intelligence, cybersecurity) students will receive intense training on state-of-the-art problems, experimental procedures, software systems, algorithms, etc. from the members of the research group selected. Students will receive a small research project to work on, typically in a group setting. Throughout the project, students discuss their results with their peers and supervisors. Students summarize their findings in a lab report written in the style of typical computer science conference papers.

Intended Learning Outcomes

No	**Competence**	**ILO**
1	Plan	Plan a research project by identifying research questions and suitable methodologies
2	Establish	Establish and configure software components necessary to conduct research experiments
3	Conduct	Conduct experiments to produce data suitable to evaluate research ideas against known state-of-the-art solutions
4	Analyze	Analyze and interpret collected data

5	Present	Present research results in a concise research report and evaluates the significance of the results produced.

Indicative Literature

- Justin Zobel, Writing for Computer Science, 3rd edition, Springer, 2014.

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Research Lab II	Laboratory Report	10 pages	100	45%	1-5

Module Completion Requirements

- **Module Examination:** minimum for pass- 45%.

Module Achievement

None.

4.7.7 Internship

Module Name	Internship
Module Code	MCSSE-INT-01
Module ECTS	10
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder (SPC)

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	2	Mandatory Elective
CSSE-MSc Computer Science & Software Engineering	3	Mandatory Elective

Student Workload	
Internship	231
Report Preparation	95
Total Hours	**326**

Module Components	**Number**	**Type**	**CP**
Internship	MCSSE-INT-01	Internship	10

Module Description

Students can undertake an internship in a company, government institution, or non-governmental organization to gain practical work experience and to start applying their knowledge into practice. A minimum of 231 working hours (i.e., 6 weeks of full-time occupation) is required for the successful completion of this module. To be professionally eligible, the content of the internship must be relevant to computer science and software engineering. The tasks to be executed during the internship should be appropriate for a master's level student. The module coordinator and Career Service Center support students in finding suitable positions. The module coordinator also decides on the professional eligibility of the internship. It is recommended to submit an internship work program prior to starting the internship.

The internship provides training and practical learning opportunities of software engineering skills in a professional setting. It assists the students' development of employer-valued skills, such as teamwork, communication, steadiness, and attention to detail. It exposes the students to the environment and performance expectations in the corporate world, may help prepare an application-oriented master thesis, and may make their entry into the professional job market easier.

Intended Learning Outcomes

No	**Competence**	**ILO**

1	Apply	Apply computer science and software engineering concepts and tools in real-world development projects.
2	Assess	Assess the quality of software development artefacts.
3	Design	Design and revise software components meeting software requirements and quality expectations.
4	Demonstrate	Demonstrate professional work attitude and business etiquettes.
5	Collaborate	Collaborate effectively in a professional environment.
6	Demonstrate	Demonstrate a solid work ethic and professional demeanor.
7	Demonstrate	Demonstrate commitment to ethical conduct and legal regulations.
8	Communicate	Communicate results to a non-expert audience.

Entry Requirements

Prerequisites	MCSSE-SE-02 Quality Engineering MBA-NC-08 Communication & Presentation Skills for Executives MCSSE-MGT-01 Agile Product Development & Design MCSSE-SE-01 Software Construction, Architecture and Engineering
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Internship	Internship Report or Business Plan and Reflection	2000 words; Completion: Pass/Fail	100	Pass/Fail	1-8

Module Completion Requirements

- **Module Examination:** pass/fail grading.

Module Achievement

None.

4.7.8 Master Thesis CSSE

Module Name	Master Thesis CSSE
Module Code	MCSSE-THE-01
Module ECTS	30
Program Owner	CSSE-MSc (Computer Science & Software Engineering)
Module Coordinator	Prof. Dr. Jürgen Schönwälder

Study Semester		
Program	**Semester**	**Status**
CSSE-MSc Computer Science & Software Engineering	4	Mandatory

Student Workload	
Colloquium	25
Independent Study	725
Total Hours	**750**

Module Components	**Number**	**Type**	**CP**
Master Thesis	MCSSE-THE-01	Thesis	30
Master Thesis	MCSSE-THE-01		

Module Description

The aim of this module is to train students to motivate, design, carry out and document a thesis project. The thesis topic is determined in mutual agreement with the module instructor. Among others, it may arise

- from research in the instructor's research area (research thesis),

- from a collaboration with a company (industry thesis), or

- from a student-driven product development idea for a start-up (start-up thesis)

In all cases, the instructor needs to agree to supervise the thesis.

The thesis work comprises the full cycle of a scientific project, starting from the identification of an open research question or focus of the work with a survey on the state of the art in research / industry / business, over the formulation of a concrete objective to the design, implementation and evaluation of an object of interest by scientific measures and with respect to the state of the art. All results are documented in the thesis report. document all of this in a thesis report. Depending on the type of thesis (research / industry / start-up), additional components, like a research / business plan, might be a necessary part of the thesis. Irrespective of the thesis type, it is a mandatory part of each thesis to

develop a digital system as known from the various branches of Computer Science and Software Engineering.

All above outlined work should be done with as much self-guidance as can be reasonably expected. The instructor will likely give substantial guidance for the first steps, whereas the other aspects will be addressed with larger degrees of self-guidance. The project consists of the thesis report (target size: 30–60 pages), and an oral presentation at the end of the course.

Recommended Knowledge

- Proficiency in the area of the chosen thesis topic.

- Read the Syllabus.

Intended Learning Outcomes

No	Competence	ILO
1	Understand	Understanding, at a professional level, of a circumscribed segment of the project in its environment (research, industry, startup).
2	Apply	Apply specific and selected CSSE techniques, as required for the project, at a professional level.
3	Apply	Apply general professional skills
4	Design	Designing and carrying out the full cycle of a project by scientific means in a professional manner.
5	Write	Writing a thesis such that it could be submitted to a scientific publication venue, as a project report to a funding agency / industrial client, or as a proposal for start-up funding.
6	Present	Presentation of project results for specialists and non-specialists.

Indicative Literature

Entry Requirements

Prerequisites	None
Co-requisites	None
Additional Remarks	None

Assessment and Completion

Module Component	Examination Type	Duration or Length	Weight (%)	Minimum for Pass	ILOs
Master Thesis	Thesis	30-60 pages	90	Ø 45% of component 1 and 2	1-6

Master Thesis	Oral Examination	30 minutes	10		1-6

Module Completion Requirements

- **Module Examination:** weighted average grading of both assessments.

Module Achievement

None.

5 Appendix

5.1 Intended Learning Outcomes Assessment Matrix

Computer Science and Software Engineering (MSc.)					Software Construction, Architecture and Engineering	Quality Engineering	Artificial Intelligence in Software Engineering	Management: Agile Product Development & Design	Management: Product Innovation & Marketing	Management: Transformational Change Management	Leadership: Entrepreneurship & Intrapreneurship	Communication & Presentation Skills for Executives	Leadership: Organizational Behavior and Industrial Organizational Psychology	Academic Writing Skills / Intercultural Training	Leadership: Agile Leadership and Strategic Management	Leadership: Customer-centric Mindset and Agile Delivery Management	Capstone Project 1	Capstone Project 2	Capstone Project 3	Research Lab I	Research Lab II	Internship	Master Thesis
Semester					1	1	2	1	2	3	1	1	2	2	3	3	1	2	3	1	1	3/4	4
Mandatory/ optional					m	m	m	m	m	m	m	m	m	m	m	m	me	me	me	me	me	me	m
Credits					5	5	5	5	5	5	2.5	2.5	2.5	2.5	2.5	2.5	5	5	5	5	5	5	30
	Competencies*																						
Program Learning Outcomes	**A**	**E**	**P**	**S**																			
Critically assess and creatively apply technological possibilities and innovations in the fields of computer science and software engineering;	x	x	x		x	x	x	x	x	x	x					x	x	x	x	x	x	x	x
Critically assess and apply software engineering methodologies considering real life situations, organizations and industries;	x	x			x	x	x	x	x	x	x					x	x	x	x	x	x	x	x
Use, adapt und improve modern artificial intelligence techniques related to data, planning and applications;	x	x						x	x	x	x					x	x	x	x	x	x	x	x
Design, implement and exploit methods in cryptography and security related fields;	x	x							x	x							x	x	x	x	x	x	x
Apply cross-disciplinary management methodologies to solve academic and professional problems;	x	x	x					x	x	x			x		x	x	x	x	x			x	x
Critically assess and integrate a consistent tool set of leadership abilities into a professional work environment;	x	x	x					x		x	x	x	x	x	x	x	x	x	x			x	x
Plan, conduct and document small research projects in the context of computer science and software engineering;	x	x	x					x	x	x	x	x	x		x	x	x	x	x	x	x		x
Independently research, document and present a scientific topic with appropriate language skills;	x	x	x	x								x	x	x	x	x	x	x	x	x	x		x
Use scientific methods as appropriate in the field of Computer Science and Software Engineering such as defining research questions, justifying methods, collecting, assessing and interpreting relevant information, and drawing scientifically-founded conclusions that consider social, scientific and ethical insights;	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		x
Develop and advance solutions to problems and arguments in their subject area and defend these in discussions with specialists and non-specialists;		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Engage ethically with academic, professional and wider communities and to actively contribute to a sustainable future, reflecting and respecting different views;		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Take responsibility for their own learning, personal and professional development and role in society, evaluating critical feedback and self-analysis;		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Apply their knowledge and understanding to a professional context;	x	x	x					x	x	x	x		x		x	x	x	x	x			x	x
Take on responsibility in a diverse team;		x	x	x				x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Adhere to and defend ethical, scientific and professional standards.		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Assessment Type																							
Written examination					x																		
Term paper														x									
Essay																							
Project report																						x	
Poster presentation																							
Laboratory Report																				x	x		
Program code																							
Oral examination																							x
Presentation								x	x	x	x	x	x		x	x							
Practical Assessments																							
Project Assessments																	x	x	x				
Portfolio Assessments					x	x																	
Master Thesis																							x
Module achievements																							

*Competencies: A-scientific/academic proficiency; E-competence for qualified employment; P-development of personality; S-competence for engagement in society

Figure 3: Intended Learning Outcomes Assessment Matrix