CONSTRUCTOR
INSTITUTE

MSc Computer Science, Software, Engineering, and Leadership (CSSEL) handbook

About the document

Title: *MSc Computer Science, Software, Engineering, and Leadership (CSSEL) handbook*

Identification: Constructor Institute Report CI-2023-016-CSL

Status:

☑ Public

☐Available to Constructor Institute members only (on Intranet)

☐ Restricted availability, subject to approval

Related documents (if applicable):

Date	Authors	Comments about this version
2023-07-10	Manuel Oriol	● Initial version ● Approved by President 03.08.2023 ● Approved by Academic Senate 09.08.2023

Table of Contents

1. Program overview

1.1 Concept

Computer science is one of the most impactful and lively research disciplines as digitalization has become the backbone of industry and society, as well as academia. Much of this progress is being driven by artificial intelligence including machine learning and cyber-physical systems, but there are also new challenges, like dealing with malicious uses and threads, which highlights the need for cybersecurity. Software, software engineering, and more generally, digital companies play a key role in this domain. Leading companies have a critical need for a new breed of digital experts. The complexity of modern software and of digitization processes demand a new generation of experts with technical management and leadership skills. Technological and economic disruption is often driven by small start-ups that require not only technical skills in developing software, but management and entrepreneurial skills to be successful in the market.

The Master of Science in Computer Science, Software Engineering, and Leadership at Constructor Institute Schaffhausen is a consecutive master's program that combines a research-oriented technological education with essential management and leadership skills to educate future leaders in technological research and industry. To prepare students for this leadership role, Constructor Institute offers training in software engineering, in terms of both development and management, as well as technical courses in three core subject areas that are currently of the utmost importance:

- Software engineering
- Cybersecurity
- Artificial Intelligence

These academic offerings mirror the research activities at Constructor Institute Schaffhausen and our partner institution Constructor University Bremen and those of the involved faculty. This guarantees excellent teaching competence and hands-on experience from individuals at the cutting edge of research and industry.

As a consecutive master's program, the MSc in Computer Science, Software Engineering, and Leadership (CSSEL) is targeted at graduates of undergraduate programs related to the field of computer science. Core knowledge in the field is a mandatory requirement to enter the MSc CSSEL program. Upon graduating from the MSc CSSEL, students will have obtained a portfolio of skills in highly relevant areas of computer science, namely s software engineering, Artificial Intelligence (AI), and c cybersecurity. Students will develop their creative and constructive abilities to produce, develop, and evaluate solutions to technical challenges. They will acquire knowledge in a selected subject area and will learn the skills necessary to approach, develop, and document small independent projects dealing with the latest in research, (industrial) applications, and the launching and management of start-ups.

Students at Constructor Institute Schaffhausen have access to the educational offerings of Constructor University. Many courses offered on software engineering at Constructor Institute Schaffhausen are taught by experts in the field. Some courses are delivered remotely at our partner institution. This cooperation facilitates quick access to real world applications and the IT job market via Constructor's excellent international network.

To strengthen the education the Institute provides, the program uses contemporary blended e-learning techniques. In addition, flipped classroom teaching will enable, whenever possible, a student-centric and hands-on experience. Team-based work on software projects benefits from this agile and developmental approach. Together with state-of-the-art equipment in software and hardware, it allows for seamless collaboration among students and instructors of different institutions, and naturally adapts to conditions that may change quickly—like they did with the onset of the Covid-19 pandemic in 2020.

By completing the master's program, students acquire core leadership skills, a solid technological backbone developed along three complementary areas, and additional management skills that will serve them well as they enter the job market. They will acquire the essential soft skills to become active leaders in a global and multi-ethnic society thanks to the international environment that characterizes Constructor Institute Schaffhausen and Constructor University Bremen. The CSSEL degree equally prepares graduates to enter the research field and pursue a PhD program or 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 in this transition, and leading companies have a critical need for a new breed of digital experts with deep technological knowledge and technical management and leadership skills.

The Computer Science, Software Engineering, and Leadership 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 successfully and responsibly engineer software. The program seeks to expand the student's competencies and capabilities in the subject areas of software engineering, cybersecurity, and Artificial Intelligence (AI), which all play dominant roles in both industry and research. To make the most of the Institute's excellent faculty and the student's time, each student selects one of these three areas as their specialized field of study. The curriculum further complements this engineering-based education by teaching modules in cross-disciplinary leadership and management, thus better preparing our graduates to become tomorrow's digital leaders.

Students are introduced to practical and research-oriented work through a Capstone project, which is an elective research project supported by frequent individual feedback sessions and personal guidance from a faculty member that culminates in the presentation of a thesis. This educational model accelerates the student's career development and helps them become valuable assets in industry and research within a short period of time.

Constructor Institute Schaffhausen 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 English-speaking work and study environments. 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.

Graduates of Computer Science, Software Engineering, and Leadership will have obtained the following competences and skills:

- Subject-matter competence in a computer science specialization
  Graduates have an 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
  Graduates gain a broad and deep knowledge in formal, algorithmic, and applied competencies in computer science. This enables them to develop independent ideas as digital experts. In response to the massive industry demand, graduates also acquire broad knowledge in software engineering, which enables them to solve practical and scientific problems in the field.

- Learning, transfer, and research skills
  Graduates learn new methodologies which enable them to solve problems in new and unfamiliar situations. They integrate learned skills in complex and multidisciplinary contexts, which has become increasingly necessary in both industry and research. Graduates are able to design 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, Constructor Institute focuses on teaching its students a broad and integrated knowledge and understanding of the fundamentals from management and leadership. Their knowledge corresponds to the standard literature in the field, which allows them to solve related problems in the field of computer science and software engineering.

- Teamwork and communication skills
  Graduates gain proficiency in the specialized exchange of ideas in a group setting with the goal of transferring these skills for the collaborative development of digital software or hardware systems. This is reinforced by the reflective practice of communication and collaboration in both academic and non-academic settings.

- Personal and professional competence
  Graduates will be able to develop a professional profile both in and out of academia and make, justify, and reflect on decisions based on theoretical and professional knowledge. They can critically examine their own behavior and assess social consequences.

1.2.2 Intended learning outcomes

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

- critically assess and creatively apply technological innovations in the fields of computer science and software engineering;
- critically assess and apply software engineering methodologies considering real life situations, organizations, and industries;

- use, adapt, und improve modern artificial intelligence techniques related to data, planning, and application;
- design, implement, and exploit methods in cryptography and security related fields;
- apply cross-disciplinary management methodologies to solve academic and professional problems;
- critically assess and integrate a consistent tool set of leadership abilities into a professional work environment;
- plan, conduct, and document small research projects in the fields of computer science and software engineering;
- independently research, document, and present a scientific topic using the appropriate language skills;
- use scientific methods as appropriate in the field of computer science, software engineering and leadership such as defining research questions, justifying methods, collecting, assessing and interpreting relevant information, and drawing scientifically-founded conclusions that consider social, scientific, and ethical impacts;
- develop and advance solutions to problems and arguments and defend these in discussions with specialists and non-specialists;
- engage ethically with academic, professional, and lay communities to actively contribute to a sustainable future, reflecting and respecting different views;
- take responsibility for their own learning, personal and professional development, and role in society, incorporating critical feedback and self-analysis;
- apply their knowledge and understanding in a professional context;
- take on responsibility as part of a diverse team;
- adhere to and defend ethical, scientific, and professional standards.

1.3 Target audience

The program is designed for students from 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 as well as understanding management and leadership topics related to technology. The program is specifically designed for candidates who are dedicated to and interested in gaining theoretical, application-oriented, and management and leadership knowledge in the fields of software engineering, cybersecurity, and artificial intelligence.

Prior to admission, applicants must have already completed their first degree in Computer Science or a scientific subject and possess strong development skills.

The program prepares students for key roles in the IT industry and for entering research in related subject fields. Part of what makes this program unique are the additional educational offerings in management and leadership courses. This coursework allows them to develop their own start-ups while on campus. The program's educational approach supports intellectual exchange and discussion within the student community, which helps students interact, appreciate different teaching and learning formats, accept challenges, and develop professionally during their time at the Institute.

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 for everyone, from technology companies to society at large.

Computer science researchers contribute new insights in a wide spectrum of disciplines. IT practitioners work in literally every area of industry, business, government, finances, energy, education, healthcare, aerospace, and many more. This work can take the form of a core IT task, such as being an administrator responsible for a system, or it can take the form of applied work done in collaboration with other experts. IT professionals 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 of the jobs they perform.

Computer Science, Software Engineering, and Leadership graduates are desperately needed all over the planet, so employers will seek out CSSEL graduates, allowing them to select from a choice of highly-paid offers.

Constructor's Alumni Association helps students establish a long-lasting and global network they can use to explore career opportunities in start-ups, industry, and academia. In addition, the broad business network of the Constructor Group provides excellent access to leading companies in many advanced technological fields.

1.5 Admission requirements

The Computer Science, Software Engineering, and Leadership graduate program requires students to have completed an undergraduate program in computer science, software engineering, information technology or another scientific discipline with a strong focus on programming skills. Students not fulfilling these requirements may still be conditionally admitted with the requirement to re-take relevant undergraduate courses. In order to receive this special dispensation, applicants need to prove a strong interest in the study program in a letter.

An applicant's social commitment as well as extracurricular and voluntary activities during undergraduate studies (e.g. university service, clubs, varsity, social work, etc.) will be considered. Work experience is not a prerequisite.

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

Study at Constructor Institute Schaffhausen 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 study.

Applicants need to submit the following documents to be considered for admission:

- Motivation Letter
- Curriculum Vitae (CV)
- University transcript in English or German
- Bachelor's degree certificate or equivalent in English or German (may be handed in later)

- An English language proficiency test (minimum score of 90 TOEFL, 6.5 IELTS, or 110 Duolingo). Native speakers and applicants who completed their undergraduate studies in English may be exempted from this requirement.
- Copy of passport
- Optional letter of recommendation.

2. Curriculum

2.1 The curriculum at a glance

In its classic form, the curriculum of the Computer Science, Software Engineering, and Leadership master's 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, culminating in a master's thesis that can cover academic research, industrial applications, or development towards a start-up.

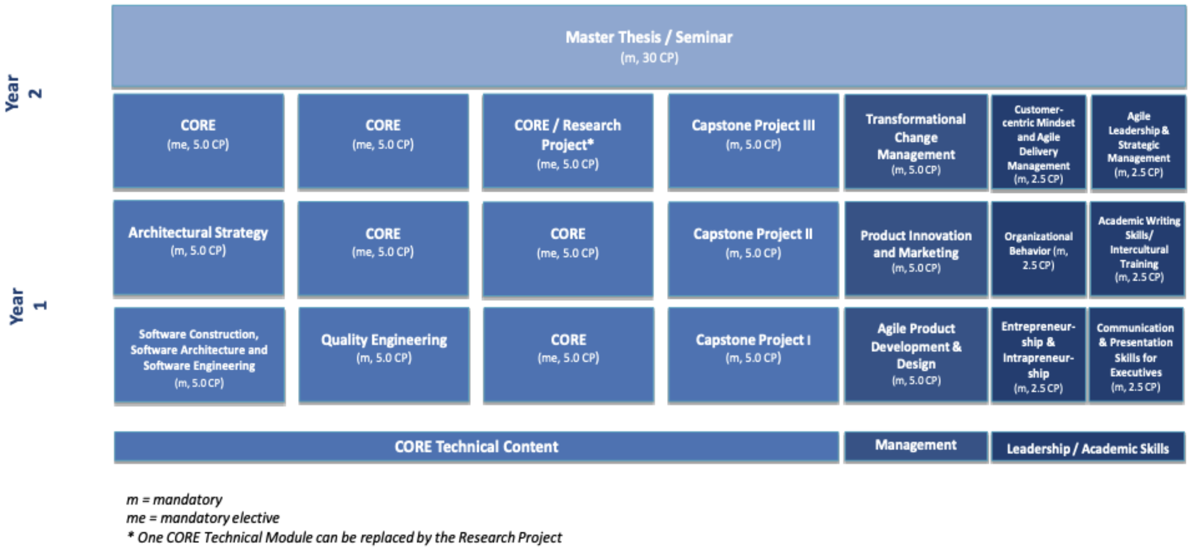
The modules are grouped into several domains, as outlined in the Schematic Study Plan (see Figure 1).

In order to graduate, students take out of these modules a total of 120 ECTS with:

- CORE Modules: 45 ECTS,
- Management Modules: 15 ECTS,
- Leadership / Academic Skills Modules: 15 ECTS,
- Capstone Project: 15 ECTS,
- Master Thesis module: 30 ECTS.

If of interest, students can replace 5 ECTS of Technical CORE Modules with a Research Project module.

2.2 Schematic study scheme



Year 2	Master Thesis / Seminar (m, 30 CP)						
	CORE (me, 5.0 CP)	CORE (me, 5.0 CP)	CORE / Research Project* (me, 5.0 CP)	Capstone Project III (m, 5.0 CP)	Transformational Change Management (m, 5.0 CP)	Customer-centric Mindset and Agile Delivery Management (m, 2.5 CP)	Agile Leadership & Strategic Management (m, 2.5 CP)
	Architectural Strategy (m, 5.0 CP)	CORE (me, 5.0 CP)	CORE (me, 5.0 CP)	Capstone Project II (m, 5.0 CP)	Product Innovation and Marketing (m, 5.0 CP)	Organizational Behavior (m, 2.5 CP)	Academic Writing Skills/ Intercultural Training (m, 2.5 CP)
Year 1	Software Construction, Software Architecture and Software Engineering (m, 5.0 CP)	Quality Engineering (m, 5.0 CP)	CORE (me, 5.0 CP)	Capstone Project I (m, 5.0 CP)	Agile Product Development & Design (m, 5.0 CP)	Entrepreneurship & Intrapreneurship (m, 2.5 CP)	Communication & Presentation Skills for Executives (m, 2.5 CP)
	CORE Technical Content				Management	Leadership / Academic Skills	

m = mandatory
me = mandatory elective
** One CORE Technical Module can be replaced by the Research Project*

Figure 1: Overview of the Master of Science in Computer Science, Software Engineering and Leadership.

2.3 Other options for the curriculum

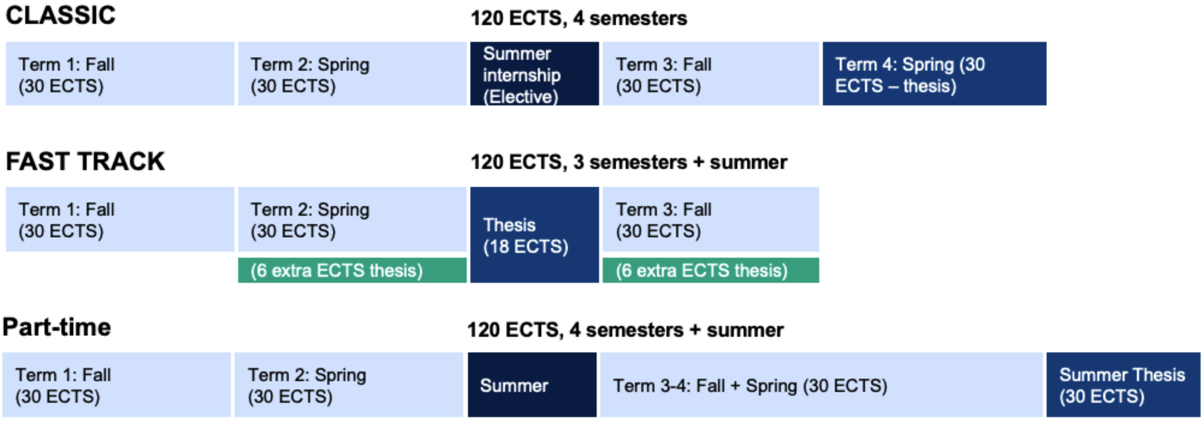
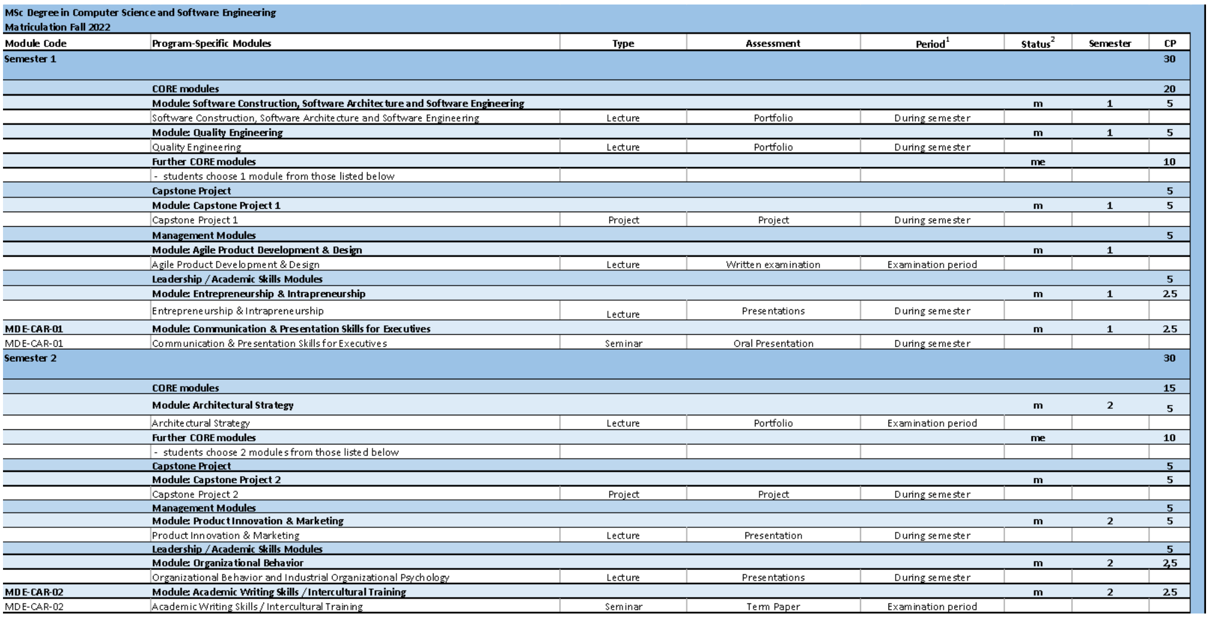


Figure 2: Overview of the Master of Science in Computer Science, Software Engineering and Leadership, Classic, Fast track and part-time

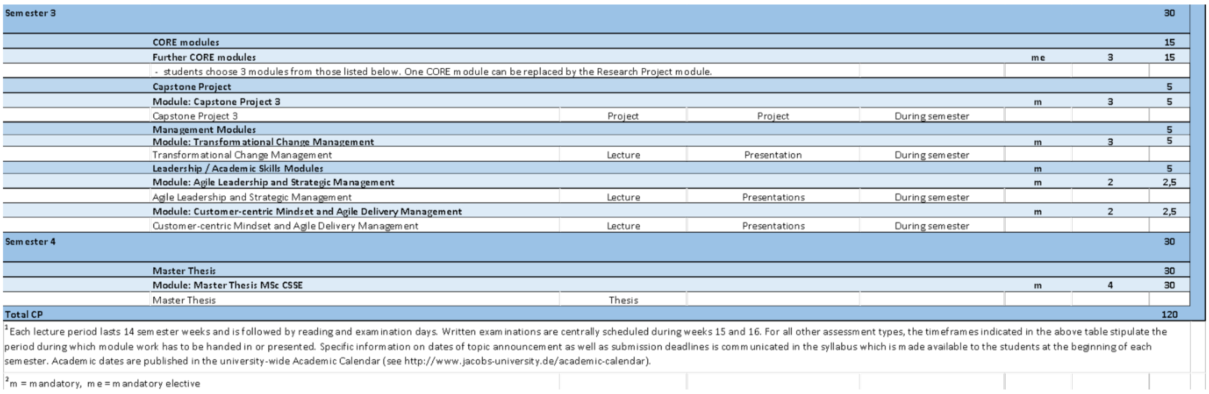
In addition to the classic program, it is also possible to choose to follow the program in a fast-track or part-time mode. These options must be discussed with the program director to see which modules should be attended and whether the student satisfies the prerequisites.

2.4 Study and examination plan



MSc Degree in Computer Science and Software Engineering
Matriculation Fall 2022

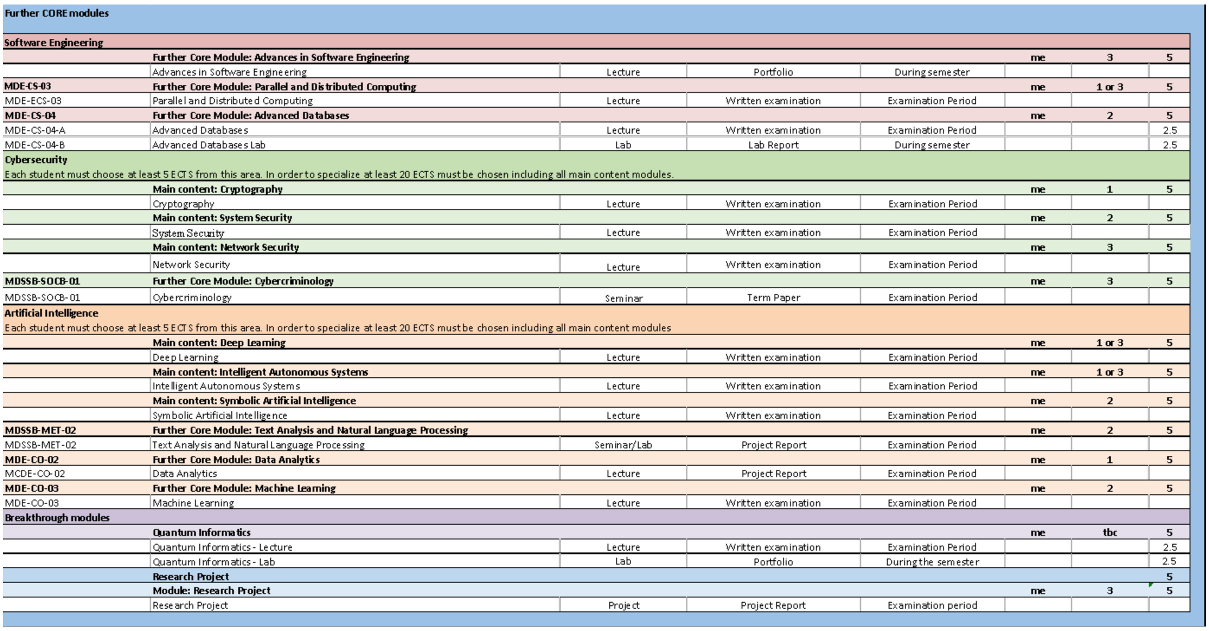
Module Code	Program-Specific Modules	Type	Assessment	Period[1]	Status[2]	Semester	CP
Semester 1							**30**
	CORE modules						**20**
	Module: Software Construction, Software Architecture and Software Engineering				**m**	**1**	**5**
	Software Construction, Software Architecture and Software Engineering	Lecture	Portfolio	During semester			
	Module: Quality Engineering				**m**	**1**	**5**
	Quality Engineering	Lecture	Portfolio	During semester			
	Further CORE modules				**me**		**10**
	- students choose 1 module from those listed below						
	Capstone Project						**5**
	Module: Capstone Project 1				**m**	**1**	**5**
	Capstone Project 1	Project	Project	During semester			
	Management Modules						**5**
	Module: Agile Product Development & Design				**m**	**1**	
	Agile Product Development & Design	Lecture	Written examination	Examination period			
	Leadership / Academic Skills Modules						**5**
	Module: Entrepreneurship & Intrapreneurship				**m**	**1**	**2.5**
	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**
	Module: Architectural Strategy				**m**	**2**	**5**
	Architectural Strategy	Lecture	Portfolio	Examination period			
	Further CORE modules				**me**		**10**
	- students choose 2 modules from those listed below						
	Capstone Project						**5**
	Module: Capstone Project 2				**m**		**5**
	Capstone Project 2	Project	Project	During semester			
	Management Modules						**5**
	Module: Product Innovation & Marketing				**m**	**2**	**5**
	Product Innovation & Marketing	Lecture	Presentation	During semester			
	Leadership / Academic Skills Modules						**5**
	Module: Organizational Behavior				**m**	**2**	**2,5**
	Organizational Behavior and Industrial Organizational Psychology	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.						
	Capstone Project						5
	Module: Capstone Project 3				m	3	5
	Capstone Project 3	Project	Project	During semester			
	Management Modules						5
	Module: Transformational Change Management				m	3	5
	Transformational Change Management	Lecture	Presentation	During semester			
	Leadership / Academic Skills Modules				m		5
	Module: Agile Leadership and Strategic Management				m	2	2,5
	Agile Leadership and Strategic Management	Lecture	Presentations	During semester			
	Module: Customer-centric Mindset and Agile Delivery Management				m	2	2,5
	Customer-centric Mindset and Agile Delivery Management	Lecture	Presentations	During semester			
Semester 4							30
	Master Thesis						30
	Module: Master Thesis MSc CSSE				m	4	30
	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 period during which module work has to be handed in or presented. Specific information on dates of topic announcement as well as submission deadlines is communicated in the syllabus which is made available to the students at the beginning of each semester. Academic dates are published in the university-wide Academic Calendar (see http://www.jacobs-university.de/academic-calendar).

[2] m = mandatory, me = mandatory elective



Further CORE modules							
Software Engineering							
	Further Core Module: Advances in Software Engineering				me	3	5
	Advances in Software Engineering	Lecture	Portfolio	During semester			
MDE-CS-03	**Further Core Module: Parallel and Distributed Computing**				me	1 or 3	5
MDE-ECS-03	Parallel and Distributed Computing	Lecture	Written examination	Examination Period			
MDE-CS-04	**Further Core Module: Advanced Databases**				me	2	5
MDE-CS-04-A	Advanced Databases	Lecture	Written examination	Examination Period			2.5
MDE-CS-04-B	Advanced Databases Lab	Lab	Lab Report	During semester			2.5
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.							
	Main content: Cryptography				me	1	5
	Cryptography	Lecture	Written examination	Examination Period			
	Main content: System Security				me	2	5
	System Security	Lecture	Written examination	Examination Period			
	Main content: Network Security				me	3	5
	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							
	Main content: Deep Learning				me	1 or 3	5
	Deep Learning	Lecture	Written examination	Examination Period			
	Main content: Intelligent Autonomous Systems				me	1 or 3	5
	Intelligent Autonomous Systems	Lecture	Written examination	Examination Period			
	Main content: Symbolic Artificial Intelligence				me	2	5
	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-03	**Further Core Module: Machine Learning**				me	2	5
MDE-CO-03	Machine Learning	Lecture	Written examination	Examination Period			
Breakthrough modules							
	Quantum Informatics				me	tbc	5
	Quantum Informatics - Lecture	Lecture	Written examination	Examination Period			2.5
	Quantum Informatics - Lab	Lab	Portfolio	During the semester			2.5
	Research Project						5
	Module: Research Project				me	3	5
	Research Project	Project	Project Report	Examination period			

2.5 Technical CORE modules

The main subject areas of the CORE modules are:

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

All students must take 15 ECTS of lecture modules in software engineering. It is also mandatory for all students to take at least one main content module (5 ECTS) each in cybersecurity and Artificial Intelligence.

Students select one of the three areas of specialization in which they have to take 20 ECTS in lecture modules out of main and suggested cross-subject content (further outlined below) and broaden their computer science, software engineering, and leadership knowledge with further free electives in Technical CORE Modules across all subject areas and suggested content.

Students not fulfilling the main admission criterion of at least 60 ECTS of computer science- related topics can still be conditionally admitted on a case-by-case basis . Part of the condition for admission can be the requirement to take further relevant courses offered in the computer science related undergraduate programs at Constructor University Bremen or at Constructor Institute Schaffhausen. These will likely be courses from the CHOICE or CORE area from these programs or mathematics courses.

2.5.1 Software engineering modules

Coursework in software engineering exposes students to a broad range of methodological and systematic approaches to 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 student's area of specialization. The majority of the modules in this area are taught in person at Schaffhausen.

Software Engineering Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Main Content (15 CP mandatory)						
Software Construction, Software Architecture and Software Engineering	XXX	1	Yes	B. Meyer	5	Schaffhausen
Quality Engineering	XXX	1	Yes	N.N.	5	Schaffhausen
Architectural Strategy	XXX	2	Yes	N.N.	5	Schaffhausen
Further CORE Module						
Advances in Software Engineering	XXX	3	No	B. Meyer	5	Schaffhausen
Parallel and Distributed Computing	MDE-CS-02	1 or 3	No	P. Zaspel	5	Bremen
Advanced Databases	MDE-CS-04	2	No	P. Baumann	5	Bremen

2.5.2 Cybersecurity modules

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

Cybersecurity Modules

Module Title	Module No.	Semester	Mandatory	Coordinator	CP	Location
Main Content (5 CP mandatory)						
Cryptography	XXX	1	No	J. Schönwälder	5	Bremen
System Security	XXX	2	No	J. Schönwälder	5	Bremen
Network Security	XXX	3	No	J. Schönwälder	5	Bremen
Further CORE Module						
Cybercriminology	MDSSB-SOCB-01	3	No	H. Brockmann	5	Bremen

2.5.3 Artificial Intelligence (AI) modules

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

Artificial Intelligence Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Main Content (5 CP mandatory)						
Deep Learning	XXX	1 or 3	No	Tbc	5	Bremen
Intelligent Autonomous Systems	XXX	1 or 3	No	A. Birk / F. Maurelli	5	Bremen
Symbolic Artificial Intelligence	XXX	2	No	A. Birk / F. Maurelli	5	Bremen
Further CORE Module						
Text Analysis and Natural Language Processing	MDSSB-MET-02	2	No	H. Brockmann / J. Lorenz / A. Wilhelm	5	Bremen
Data Analytics	MDE-CO-02	1	No	A. Wilhelm	5	Bremen

Machine Learning	MDE-CO-04	2	No	S. Kettemann	5	Bremen

2.5.4 Breakthrough area modules

Digital leadership requires a long-term perspective. In this elective area, students are exposed to breakthrough areas of the field. This area is expanded as more applications are identified.

Breakthrough Area Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Quantum Informatics	XXX	Tbc	No	W. Tittel	5	Schaffhausen

2.6 Management modules

Students take modules in the fields of product development, marketing, and change management to teach them the market-relevant management skills they will need in the future. All modules are mandatory for the program.

Management Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Agile Product Development and Design	XXX	1	Yes	T. Halaszovich	5	Bremen
Product Innovation and Marketing	XXX	2	Yes	T. Halaszovich	5	Bremen
Transformational Change Management	XXX	3	Yes	T. Halaszovich	5	Bremen

2.7 Leadership / academic skills modules

Future success in industry and in research is strengthened with a set of Leadership and Academic Skills Modules. All modules below must be taken in order for a student to graduate.

Leadership and Academic Skills Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Entrepreneurship and Intrapreneurship	XXX	1	Yes	T. Halaszovich	2.5	Bremen
Communication and Presentation Skills for Executives	MDE-CAR-01	1	Yes	S. Kettemann	2.5	Bremen

Organizational Behavior	XXX	2	Yes	C. Stamov Roßnagel	2.5	Bremen
Academic Writing Skills / Intercultural Training	MDE-CAR-02	2	Yes	S. Kettemann	2.5	Bremen
Agile Leadership and Strategic Management	XXX	3	Yes	T. Halaszovich	2.5	Bremen
Customer-centric Mindset and Agile Delivery Management	XXX	3	Yes	T. Halaszovich	2.5	Bremen

2.8 Project, Capstone project and master’s thesis

To explore the full development process of a software application with relation to the areas of specialization of the program, all students take the three modules of the Capstone Project. It is highly recommended to take the three modules in their numerical order. Students with a strong focus on academic research can replace Technical CORE Module with the Research Project in their third semester, which is carried out in one of the research areas of the Faculty.

The master’s studies conclude with a 6-month Master’s Thesis, which extends over the fourth and final semester.

Capstone Project, Research Project and Thesis Modules						
Module Title	**Module No.**	**Semester**	**Mandatory**	**Coordinator**	**CP**	**Location**
Capstone Project 1	XXX	1	Yes	M. Oriol	5	Schaffhausen
Capstone Project 2	XXX	2	Yes	M. Oriol	5	Schaffhausen
Capstone Project 3	XXX	3	Yes	M. Oriol	5	Schaffhausen
Research Project	XXX	3	No	B. Meyer	5	Schaffhausen
Master Thesis	XXX	4	Yes	B. Meyer	30	Schaffhausen

3. CSSEL graduate program regulations

3.1 Scope of these regulations

The regulations in this handbook are valid for all students who entered the Computer Science, Software Engineering, and Leadership graduate program at Constructor Institute in Fall 2023 or later. In case of conflict between the regulations in this handbook and the general Policies for Master's Studies, the latter apply.

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

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

3.2 Degree

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

3.3 Graduation requirements

To graduate, students need to obtain 120 credit points. In addition, the following graduation requirements apply.

3.4 Other program-specific policies and practices

Close contact and cooperation between program representatives and students are 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 impression 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 so 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 study (e.g., change of the semester sequence, assessment type, or the teaching mode of courses). Constructor Institute Schaffhausen reserves the right to modify the regulations of the program handbook.

4. Module descriptions

4.1 Core modules

4.1.1 Software Engineering modules

4.1.1.1 Software Construction, Software Architecture, and Software Engineering

<table>
<tr><td colspan="3"><b><i>Module Name</i></b>
Software Construction, Software Architecture, and Software Engineering</td><td><b><i>Module Code</i></b>
tbd</td><td><b><i>Level (type)</i></b>
Year 1</td><td><b><i>CP</i></b>
5</td></tr>
<tr><td colspan="6"><b><i>Module Components</i></b></td></tr>
<tr><td><i>Number</i></td><td colspan="3"><i>Name</i></td><td><i>Type</i></td><td><i>CP</i></td></tr>
<tr><td>tbd</td><td colspan="3">Software Construction, Software Architecture, and Software Engineering</td><td>Lecture</td><td>5</td></tr>
<tr><td><b><i>Module Coordinator</i></b>
Prof. Dr. Bertrand Meyer</td><td colspan="3"><b><i>Program Affiliation</i></b>
● MSc Computer Science, Software Engineering and Leadership</td><td colspan="2"><b><i>Mandatory Status</i></b>
Mandatory for CSSEL</td></tr>
<tr><td rowspan="2"><b><i>Entry Requirements</i></b>
<i>Pre-requisites</i>
☒ none</td><td rowspan="2"><i>Co-requisites</i>
☒ none</td><td rowspan="2"><i>Knowledge, Abilities, or Skills</i>
● Some programming experience</td><td><b><i>Frequency</i></b>
Annually (Fall)</td><td colspan="2"><b><i>Forms of Learning and Teaching</i></b>
● Lectures (35 hours)
● Private study (90 hours)</td></tr>
<tr><td><b><i>Duration</i></b>
1 semester</td><td colspan="2"><b><i>Workload</i></b>
125 hours</td></tr>
<tr><td colspan="6"><b><i>Recommendations for Preparation</i></b></td></tr>
<tr><td colspan="6"><b><i>Content and Educational Aims</i></b>
Software engineering is the body of concepts and techniques that make it possible to construct industrial software systems of high quality. The size, complexity and ambition of systems being developed today requires a systematic approach based on best practices learned over the past decades. Software engineering includes many aspects, both technical (requirements, design, programming, testing, and other validation techniques, maintenance) and managerial (project management, metrics, empirical studies, agile methods, lifecycle models, quality assurance). After taking the course, students will understand the issues and challenges of successful software</td></tr>
</table>

system construction and will be ready to apply them to build high-quality software, including in management roles.

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

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Use object-oriented techniques to produce high-quality programs.
2. Take advantage of mechanisms of inheritance, genericity and information hiding.
3. Take advantage of Design by Contract techniques to guarantee the reliability of their programs.
4. Apply fundamental design patterns (Observer, Visitor and others).
5. Apply basic techniques of modern software engineering such as configuration management.
6. Apply basic agile development techniques.

Indicative Literature

Pfleeger, S. and Atlee, J.M. (2010). Software Engineering: Theory and Practice (4th Edition)
Ghezzi, C., Jazayeri, M. and Mandrioli, D (2003). Fundamentals of software engineering (2th Edition), ISBN 978-0-13-305699-0

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment: Portfolio (Quizzes, Programming Assignments) Weight: 100 %

Scope: All intended learning outcomes of the module.

Bonus achievement: Additional bonus homework as a voluntary task can improve the grade but is not required to reach the best grade in the module (1.0).

4.1.1.2 Quality Engineering

Module Name Quality Engineering			**Module Code** tbd	**Level (type)** Year 1	**CP** 5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Quality Engineering			Lecture	5
Module Coordinator N.N.	**Program Affiliation** ● MSc Computer Science, Software Engineering and Leadership			**Mandatory Status** Mandatory for CSSEL	
Entry Requirements			**Frequency** Annually (Fall)	**Forms of Learning and Teaching** ● Lectures (35 hours) ● Private study (90 hours)	
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*			
☒ none	☒ none	● Programming skills in an imperative language at CS bachelor's level ● Algorithms and data structure at CS bachelor's level ● Basic skills in software testing: structural testing, Junit ● Basic knowledge of software engineering and IDEs at CS bachelor's level	**Duration** 1 semester	**Workload** 125 hours	

• Discrete math at CS bachelor's level		

Recommendations for Preparation

Content and Educational Aims

Software quality can be defined as the degree of satisfaction with what is produced; it represents an essential part of the software development process 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 general testing and analysis principles and 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Manage a software quality process.
2. Select and implement a suitable set of testing and analysis activities to certify the quality of software systems.
3. Understand the core principles of software testing and program analysis.
4. Master the basic techniques underlying software testing and program analysis.
5. Choose the suitable approaches to address the different testing and analysis programs.
6. Design and monitor a suitable quality process.

Indicative Literature

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment: Portfolio (Individual Assignments, Group Assignments,) Weight: 100 %

Scope: All intended learning outcomes of the module.

Bonus achievement: Additional bonus homework as a voluntary task can improve the grade but is not required to reach the best grade in the module (1.0).

4.1.1.3 Architectural Strategy

Module Name Architectural Strategy			***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Architectural Strategy			Lecture	5
Module Coordinator N.N.	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements			***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ● Lectures (35 hours) ● Private study (70 hours)	
Pre-requisites ☒ none	*Co-requisites* ☒ none	*Knowledge, Abilities, or Skills* ●			
			Duration	***Workload*** 125 hours	
Recommendations for Preparation					

Content and Educational Aims

The course “Architectural Strategy” focuses on Software Architectures, the key element for systematically developing large and complex software systems. During the course, we study how to design, recover, analyze, and document software architectures and understand how the main design decisions comprising them influence the quality attributes of the resulting systems.

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

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Understand methods for designing large software systems.
2. Design complex and large software systems using components and connectors.
3. Use UML as modelling language to represent the main concepts of software systems.
4. Document their main design decisions and motivate them in terms of quality attributes.

Indicative Literature

R.N. Taylor, N. Medvidovic, E.M. Dashofy, Software Architecture: Foundations, Theory, and Practice, Wiley, January (2009)

Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice. Addison Wesley 2013

C. Pautasso, Software Architecture, 2020 (Visual Lecture Notes)

Usability and Relationship to other Modules

•

Examination Type: Module Examination

Assessment: Portfolio (Individual Assignments, Group Assignments,) Weight: 100 %

Scope: All intended learning outcomes of the module.

Bonus achievement: Additional bonus homework as a voluntary task can improve the grade but is not required to reach the best grade in the module (1.0).

4.1.1.4 Advances in Software Engineering

<table>
<tr><td colspan="3"><b><i>Module Name</i></b>
Advances in Software Engineering</td><td><b><i>Module Code</i></b>
tbd</td><td><b><i>Level (type)</i></b>
Year 2</td><td><b><i>CP</i></b>
5</td></tr>
<tr><td colspan="6"><b><i>Module Components</i></b></td></tr>
<tr><td><i>Number</i></td><td colspan="3"><i>Name</i></td><td><i>Type</i></td><td><i>CP</i></td></tr>
<tr><td>tbd</td><td colspan="3">Advances in Software Engineering</td><td>Lecture</td><td>2.5</td></tr>
<tr><td>tbd</td><td colspan="3">Advances in Software Engineering – Lab</td><td>Lab</td><td>2.5</td></tr>
<tr><td><b><i>Module Coordinator</i></b>
Prof. Dr. Bertrand Meyer</td><td colspan="3"><b><i>Program Affiliation</i></b>
● MSc Computer Science, Software Engineering and Leadership</td><td colspan="2"><b><i>Mandatory Status</i></b>
Mandatory elective for CSSEL</td></tr>
<tr><td colspan="3"><b><i>Entry Requirements</i></b></td><td rowspan="2"><b><i>Frequency</i></b>
Annually
(Spring)</td><td colspan="2" rowspan="2"><b><i>Forms of Learning and Teaching</i></b>
● Lectures (17.5 hours)
● Lab (17.5 hours)
● Private Study (90 hours)</td></tr>
<tr><td><i>Pre-requisites</i></td><td><i>Co-requisites</i></td><td><i>Knowledge, Abilities, or Skills</i></td></tr>
<tr><td>☒ Software Construction, Software Architecture and Software Engineering</td><td>☒ None</td><td>● Familiarity with basics of software engineering and software architecture
● Programming experience</td><td><b><i>Duration</i></b>
1 semester</td><td colspan="2"><b><i>Workload</i></b>
125 hours</td></tr>
<tr><td colspan="6"><b><i>Recommendations for Preparation</i></b></td></tr>
<tr><td colspan="6"><b><i>Content and Educational Aims</i></b></td></tr>
</table>

The course covers topics of modern software engineering beyond the basic concepts covered in the first semester SCAE course (Software Construction, Architecture, and Engineering). 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.

Intended Learning Outcomes

1. Apply techniques of formal software verification, particularly axiomatic semantics, to proving program correctness.
2. Use a program-proving framework.
3. Perform effective requirements.
4. Apply requirements techniques such as use cases and object-oriented requirements.
5. Use agile development techniques to manage a project.
6. Make the difference between productive and harmful agile ideas.
7. Combine agile methods with process models such as CMMI.

Indicative Literature

Bertrand Meyer, Handbook of Requirements Engineering and Business Analysis, Springer, 2022

Flemming Nielson, Hanne Riis Nielson, Chris Hankin: Principles of Program Analysis, Springer, most recent edition

Bertrand Meyer, Agile! The Good, the Hype and the Ugly, Springer. 2014

Usability and Relationship to other Modules

Examination Type: Module Component Examinations (tbc)

Module Component 1: Lecture

Assessment Type: Written examination Duration/length: 90 min

Weight: 50%

Scope: All intended learning outcomes of this module.

Module Component 2: Lab

Assessment Type: Requirements Project

Weight: 50 %

Scope: All intended learning outcomes of this module. Completion: To pass this module, the examination of each module component must be passed with at least 45%

4.1.1.5 Parallel and Distributed Computing

<table>
<tr><td colspan="3"><b><i>Module Name</i></b>
Parallel and Distributed Computing</td><td><b><i>Module Code</i></b>
MDE-CS-02</td><td><b><i>Level (type)</i></b>
Year 2</td><td><b><i>CP</i></b>
5</td></tr>
<tr><td colspan="6"><b><i>Module Components</i></b></td></tr>
<tr><td><i>Number</i></td><td colspan="3"><i>Name</i></td><td><i>Type</i></td><td><i>CP</i></td></tr>
<tr><td>MDE-CS-02</td><td colspan="3">Parallel and Distributed Computing</td><td>Lecture</td><td>5</td></tr>
<tr><td><b><i>Module Coordinator</i></b>
tbd</td><td colspan="3"><b><i>Program Affiliation</i></b>
▪ MSc Computer Science, Software Engineering and Leadership</td><td colspan="2"><b><i>Mandatory Status</i></b>
Mandatory elective for CSSEL</td></tr>
<tr><td rowspan="2"><b><i>Entry Requirements</i></b>
<i>Pre-requisites</i>
☒ None</td><td rowspan="2"><i>Co-requisites</i>
☒ None</td><td rowspan="2"><i>Knowledge, Abilities, or Skills</i>
▪ Basic knowledge in C/C++
▪ Mandatory proficiency in Python</td><td><b><i>Frequency</i></b>
Annually
(Fall)</td><td colspan="2"><b><i>Forms of Learning and Teaching</i></b>
▪ Lecture (35 hours)
▪ Private study (90 hours)</td></tr>
<tr><td><b><i>Duration</i></b>
1 semester</td><td colspan="2"><b><i>Workload</i></b>
125 hours</td></tr>
<tr><td colspan="6"><b><i>Recommendations for Preparation</i></b>
If no knowledge in C/C++ is present, interested students are encouraged get a basic understanding of C/C++ (via online material) in order to better understand some of the concepts discussed.</td></tr>
<tr><td colspan="6"><b><i>Content and Educational Aims</i></b>
In recent years, the development of parallel and cloud computing has opened the door for Big Data analysis and processing. This module aims to provide an overview and introduction to the vast field of parallel and cloud computing. In traditional parallel computing, we aim to develop notions for different parallelization models (shared-memory, distributed-memory, SIMD, SIMT), get to know appropriate programming methodologies for high performance data analysis (OpenMP / MPI), and aim to understand performance and scalability in this field (weak vs. strong scaling, Amdahl’s law).This fundamental knowledge will then be carried over to recent developments in cloud computing, where distributed processing frameworks (Spark / Hadoop MapReduce / Dask), based on appropriated deployment infrastructures, will become de facto standards for Big Data processing and analysis. We will approach these technologies from a practical point of view and aim to develop the</td></tr>
</table>

necessary knowledge to carry out scalable machine learning and data processing on Big Data.

Intended Learning Outcomes

By the end of this module, students should be able to

1. Understand theory and fundamentals of parallelization models (shared-/distributed memory, SIMD, SIMT).
2. Explain and apply parallel programming methodologies (OpenMP / MPI).
3. Describe and analyze performance and scalability (weak vs. strong scaling, etc.).
4. Understand basic principles of distributed and cloud computing.
5. Use distributed processing frameworks (Spark / Hadoop MapReduce / Dask) for scalable distributed calculations.
6. Develop scalable machine learning and data processing on Big Data.

Indicative Literature

Zaccone, Python Parallel Programming Cookbook, O'Reilly.

J.C. Daniel, Data Science with Python and Dask, Manning Publications.

Z. Radtka, D. Miner, Hadoop with Python. Hadoop with Python, O'Reilly.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Written Examination Duration: 120 minutes

Weight: 100%

Scope: All intended learning outcomes of this module.

4.1.1.6 Advanced Databases

Module Name Advanced Databases			***Module Code*** MDE-CS-04	***Level (type)*** Year 1	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
MDE-CS-04-A	Advanced Databases			Lecture	2.5
MDE-CS-04-B	Advanced Databases Lab			Lab	2.5
Module Coordinator Prof. Dr. Peter Baumann	***Program Affiliation*** ▪ MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory Elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ● 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	***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ▪ Lecture (40 hours) ▪ Lab (40 hours) ▪ Private study (45 hours)	
			Duration 1 semester	***Workload*** 125 hours	

Recommendations for Preparation

Content and Educational Aims

This course deepens knowledge and skills in managing and serving Big Data with an emphasis on flexibility and scalability. After completing this course, students will know state of the art procedures in data management for particularly large and complex data, including in cloud-based data setups. Based on the Data Engineering Core lecture, the course starts with a reinspection of classical SQL, including an overview of SQL query processing. Based on this understanding, opportunities for 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 knowledge on database modelling, tuning, and security. Guided hands-on exercises complement this.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Summarize the state of the art in data management for particularly large and complex data.
2. Establish criteria for selecting adequate scalable data management technology based on various criteria.
3. Establish a state-of-the-art database schema for a given application scenario.
4. Tune a relational database for best performance on some given query workload.
5. Adequately consider security aspects in databases.
6. 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 and 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.

Usability and Relationship to other Modules

Examination Type: Module Component Examinations

Module Component 1: Lecture

Assessment Type: Written Exam Duration: 120 min

Weight: 67%

Scope: Intended learning outcomes (1,2,3,4,5).

Module Component 2: Lab

Assessment Type: Lab Report

Weight: 33%

Scope: Intended learning outcomes (3,4,5,6).

Completion: To pass this module, the examination of each module component must be passed with at least 45%.

4.1.2 Cybersecurity modules

4.1.2.1 Cryptography

Module Name Cryptography		***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components				
Number	*Name*		*Type*	*CP*
tbd	Cryptography		Lecture	5
Module Coordinator Prof. Dr. Jürgen Schönwälder	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership		***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ none	*Co-requisite* ☒ none *Knowledge, Abilities, or Skills*	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Lectures (35 hours) ● Private study (70 hours) ● Exam preparation (20 hours)	
		Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation Students are expected to have a solid mathematical foundation. Students should review basic concepts of number theory, probability theory, and complexity theory as preparation for this module.				

Content and Educational Aims

Information security requires techniques to protect information and to secure communication. Cryptography studies the design of cryptographic algorithms that can ensure the confidentiality, the integrity, and the authenticity of data and messages exchanged in a secure communication protocol. 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Understand the mathematical problems on which cryptographic algorithms are built.
2. Describe pseudo random number generators and pseudo random functions.
3. Evaluate the strengths, weaknesses, and the applicability of cryptographic algorithms.
4. Select from a set of symmetric block cipher, message integrity, and authenticated encryption algorithms.
5. Contrast different asymmetric ciphers (finite field based, elliptic curve based, lattice based, hash based).
6. Explain the notion of quantum resistant cryptographic algorithms.
7. Analyze the properties of cryptographic protocols such as key exchange mechanisms.
8. Apply techniques to analyze cryptographic protocols and their implementations.
9. Explain homomorphic encryption schemes and differential privacy.

Indicative Literature

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

Usability and Relationship to other Modules

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

Examination Type: Module Examination

Assessment: Written examination Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.2.2 System Security

Module Name System Security			***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	System Security			Lecture	5
Module Coordinator Prof. Dr. Jürgen Schönwälder	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ Cryptography	*Co-requisites* ☒ none	*Knowledge, Abilities, or Skills*	***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ● Lectures (35 hours) ● Private study (70 hours) ● Exam preparation (20 hours)	
			Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation Students are expected to be familiar with how programs are executed 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.					

Content and Educational Aims

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 discuss how authorization mechanisms are realized in the different system software components and how they can be used to define effective security policies.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Describe microarchitectural attacks and computer components and suitable counter measures.
2. Illustrate trusted execution environments and how they can be used to bootstrap security.
3. Compare the isolation achieved by hypervisors and operating system mechanisms.
4. Assess application layer isolation and sandboxing mechanisms.
5. Explain how systems can identify misbehaving code and protection themselves against malware.
6. Outline how protected data storage can be implemented.
7. Recommend authentication methods suitable for different kinds of applications.
8. Compose authorization mechanisms to define effective security policies.

Indicative Literature

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

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.

Examination Type: Module Examination

Assessment: Written examination

Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.2.3 Network Security

Module Name Network Security			***Module Code*** tbd	***Level (type)*** Year 2	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Network Security			Lecture	5
Module Coordinator Prof. Dr. Jürgen Schönwälder	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ Cryptography	*Co-requisites* ☒ none	*Knowledge, Abilities, or Skills*	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Lectures (35 hours) ● Private study (70 hours) ● Exam preparation (20 hours)	
			Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation 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.					

Content and Educational Aims

Computer networks like 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 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 on 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Describe techniques to obtain information about networked computing systems.
2. Contrast mechanisms in the different network protocol layers for traffic manipulation and redirection.
3. Explain how distributed denial of service attacks are executed and how botnets are constructed.
4. Evaluate security mechanisms such as firewalls and anomaly / intrusion detection systems.
5. Analyze generic security protocols such as IPsec, TLS, SSH and how they have evolved.
6. Compare protocols aiming to secure the network infrastructure (name resolution, routing).
7. Evaluate the security properties of modern software-defined network architectures.
8. Design scalable solutions for protecting communication in distributed applications.

Indicative Literature

- William Stallings: Cryptography and Network Security: Principles and Practice, 7th edition, Pearsons, 2018
- 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

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.

Examination Type: Module Examination

Assessment: Written examination

Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.2.4 Cybercriminology

Module Name Cybercriminology			***Module Code*** MDSSB-SOCB-01	***Level (type)*** Year 2	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
MDSSB-SOCB-01	Cybercriminology			Seminar	5
Module Coordinator Prof. Dr. Hilke Brockmann	***Program Affiliation*** ● MSc Data Science for Society and Business			***Mandatory Status*** Mandatory elective for DSSB and CSSEL	
Entry Requirements			***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Seminar (35 hours) ● Teamwork and Self-study (90 hours)	
Pre-requisites ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ● Python or R			
			Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation Watch the TED talk: https://www.youtube.com/watch?v=c_2Ja-OTmGc					
Content and Educational Aims 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.					

Intended Learning Outcomes

By the end of this module, students should be able to

1. Know and understand the core concepts of cybercriminology, policing structures and techniques, and national as well as international cybercrime jurisdiction.
2. Demonstrate the ability to critically, autonomously, and creatively identify and formulate cybercrime related problems.
3. Demonstrate methodological knowledge in studying and critically analyzing cybercrime research questions.
4. Find best solutions to secure private persons, business organizations, and entire societies from cybercrime offenses.
5. Demonstrate insights into the possibilities and limitations of cybercrime research and their role in the society.
6. 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 and Francis.
Maimon, Louderback (2019) Cyber-Dependent Crimes: An Interdisciplinary Review. *Annual Review of Criminology* 2, 191-216.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Term Paper Length: 3000 – 4000 words
Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.3 Artificial Intelligence (AI) modules

4.1.3.1 Deep Learning

Module Name Deep Learning		***Module Code*** tbd	***Level (type)*** Year 1 / 2	***CP*** 5
Module Components				
Number	*Name*		*Type*	*CP*

tbd	Deep Learning	Lecture	5
Module Coordinator N.N. / Prof. Dr. Peter Zaspel	***Program Affiliation*** ● MSc Computer Science and Software Engineering	***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ none	*Co-requisites* ☒ none *Knowledge, Abilities, or Skills* ● Strong knowledge and abilities in mathematics (linear algebra, calculus).	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Lectures (35 hours) ● Private study (70 hours) ● Exam preparation (20 hours)
		Duration 1 Semester	***Workload*** 125 hours
Recommendations for Preparation This module is recommended for students that have been exposed to core knowledge in machine learning / statistical learning on the 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 entry into the course.			

Content and Educational Aims

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, 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 the core knowledge related to training, model evaluation, and multilayer perceptrons, this module focuses on the exposing students to deep learning techniques including convolutional and recurrent neural networks, autoencoders, generative adversarial networks, and reinforcement learning. The central aim is to enable students to critically assess and apply modern methods in machine learning.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

2 Understand core techniques to train neural networks.
3 Select from modern neural network architectures the most appropriate method (e.g. convolutional and recurrent neural networks) based on given input data.
4 Contrast different recent unsupervised learning methods including autoencoders and generative adversarial networks.
5 Describe techniques in reinforcement learning.

Indicative Literature

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

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.

Examination Type: Module Examination

Assessment: Written Examination

Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.3.2 Intelligent Autonomous Systems

Module Name	***Module Code***	***Level (type)***	***CP***
Intelligent Autonomous Systems	tbd	Year 1 / 2	5

Module Components

Number	*Name*	*Type*	*CP*
tbd	Intelligent Autonomous Systems	Lecture	5

Module Coordinator	***Program Affiliation***	***Mandatory Status***
Prof. Dr. Andreas Birk, Prof. Dr. Francesco Maurelli	● MSc Computer Science, Software Engineering and Leadership	Mandatory elective for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	● Lectures (35 hours) ● Private study (70 hours) ● Exam preparation (20 hours)
☒ none	☒ none			
			Duration	***Workload***
			1 semester	125 hours

Recommendations for Preparation

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.

Content and Educational Aims

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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Describe use-cases of AI in a system-oriented way.
2. Use IAS software tools, frameworks, and libraries.
3. Assess which AI software components are needed to conduct a given complex task in an intelligent autonomous way by a machine.
4. Explain the fundamental concepts and algorithms of core building blocks, namely machine perception, world modelling, navigation, and manipulation.
5. Recommend software architectures for system-oriented AI applications.
6. Integrate IAS software components in an application scenario.

Indicative Literature

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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment: Written examination Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.3.5 Symbolic Artificial Intelligence

Module Name Symbolic Artificial Intelligence			***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Symbolic Artificial Intelligence			Lecture	5
Module Coordinator Prof. Dr. Andreas Birk, Prof. Dr. Francesco Maurelli	***Program Affiliation*** • MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ none	*Co-requisites* ☒ none	*Knowledge, Abilities, or Skills*	***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** • Lectures (35 hours) • Private study (70 hours) • Exam preparation (20 hours)	
			Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation					
Content and Educational Aims This module deals with what is often called classical AI, i.e., especially formal methods based on symbolic representations. The module begins 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 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 cooperation and coordination. Finally, it is shown how classical methods and representations are also increasingly used on a conceptual level within other AI areas 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

Upon completion of this module, students will be able to:

1. Describe the different areas of AI and their conceptual relations to each other.
2. Explain the use of search algorithms for problem-solving.
3. Use logic for representation, reasoning, and planning.
4. Implement and integrate fuzzy logic representation and reasoning.
5. Use probabilistic knowledge representation, reasoning, and planning.
6. Explain core concepts and methods of Multi-Agent-Systems.
7. Assess which classical AI concepts and methods are useful and applicable components for a given application-oriented system.
8. Integrate classical AI software components into hybrid AI systems.

Indicative Literature

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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment: Written examination Duration: 120 min

Weight: 100%

Scope: All intended learning outcomes of the module.

Text Analysis and Natural Language Processing

Module Name Text Analysis and Natural Language Processing	***Module Code*** MDSSB-MET-02	***Level (type)*** Year 1	***CP*** 5

Module Components

Number	*Name*	*Type*	*CP*
MDSSB-MET-02	Text Analysis and Natural Language Processing	Seminar/Lab	5

Module Coordinator Prof. Dr. Hilke Brockmann/ Dr. Jan Lorenz / Prof. Dr. Adalbert F.X. Wilhelm	***Program Affiliation*** ● MSc Data Science for Society and Business	***Mandatory Status*** Mandatory for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Spring)	● Seminar (17.5 hours) ● Lab sessions (17.5 hours) ● Private Study (90 hours)
☒ None	☒ None	☒ Programming skills in R or Python at an intermediate level (tbc)	***Duration*** 1 semester	***Workload*** 125 hours

Recommendations for Preparation

None.

Content and Educational Aims

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 the 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 modelling.

Intended Learning Outcomes

By the end of this module, students should be able to

1. Explain the concept of “text as data”.
2. Use basic methods for information extraction and text data retrieval.
3. Process and prepare text data for statistical modelling and automated content analysis.
4. Perform different text analyses using text mining packages in R.
5. Interpret diverse text analytical measures.
6. 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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Project Report Length: 3000 words
Weight: 100%

Scope: All intended learning outcomes of the module.

4.1.3.6 Data Analytics

Module Name	***Module Code***	***Level (type)***	***CP***
Data Analytics	MDE-CO-02	Year 1	5

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

Module Coordinator	***Program Affiliation***	***Mandatory Status***
Prof. Dr. Adalbert F.X. Wilhelm	▪ MSc Computer Science, Software Engineering and Leadership	Mandatory Elective for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	▪ Lecture (17.5 hours) ▪ Tutorials (17.5 hours) ▪ Private study (90 hours)
☒ None	☒ None	☒ None	***Duration***	***Workload***
			1 semester	125 hours

Recommendations for Preparation

Read the Syllabus.

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

Content and Educational Aims

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 a practical exposure to the data analysis process.

Intended Learning Outcomes

By the end of this module, students will be able to

1. Explain advanced data analytics techniques in theory and application.
2. Apply data analytics methods to real-life problems using appropriate tools.
3. Evaluate and compare different data analytics algorithms and approaches.
4. Apply statistical concepts to evaluate data analytics results.

Indicative Literature

G. James, D. Witten, T. Hastie, Rob Tibshirani: Introduction to Statistical Learning with R by Springer, 2013 (ISLR)

A. Telea, Data Visualization: Principles and Practice, Wellesley, Mass.: AK Peters, 1st edition, 2008.(DV)

M. Ward, G. Grinstein, D. Keim, Interactive Data Visualization: Foundations, Techniques, and Applications. AK Peters, 1st edition, 2010. (IDV)

Usability and Relationship to other Modules

This module together with the module “Machine Learning” are favorable companion modules for students with a focus on Software Engineering or Cybersecurity that still want to gain knowledge in these relevant areas. “Deep Learning” targets a deeper understanding of the related field.

Examination Type: Module Examination

Assessment Type: Project Report Length: 20 pages

Weight: 100%

Scope: All intended learning outcomes of this module.

4.1.3.7 Machine Learning

Module Name	***Module Code***	***Level (type)***	***CP***
Machine Learning	MDE-CO-04	Year 1	5

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

Module Coordinator	***Program Affiliation***	***Mandatory Status***
Prof. Dr. Stefan Kettemann	▪ MSc Computer Science, Software Engineering and Leadership	Mandatory for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ▪ Basic linear algebra, calculus and probability theory, as typically acquired in entry modules in BSc studies	Annually (Spring)	▪ Lectures (35 hours) ▪ Private Study, incl. exercises and exam preparation (90 hours)
			Duration 1 semester	***Workload*** 125 hours

Recommendations for Preparation

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.

Content and Educational Aims

Machine learning (ML) is a module that is concerned with algorithms that are fed (large quantities of) real-world data and which return a compressed "model" of the data. An example is a spoken language model where 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 these 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Design, implement and exploit elementary supervised ML methods for classification and regression with expert care given to dimension reduction preprocessing and regularization.
2. Understand and practically use PCA and linear regression.
3. 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.

Usability and Relationship to other Modules

This module together with the module “Data Analytics” are good companion modules for students with a focus on Software Engineering or Cybersecurity that still want to gain knowledge in these relevant areas. “Deep Learning” targets a deeper understanding of the related field.

Examination Type: Module Examination

Assessment Type: Written Exam Duration: 120 minutes

Weight: 100%

Scope: All intended learning outcomes of this module.

4.2 Breakthrough area modules

4.2.1 Quantum Informatics

Module Name

Quantum Informatics

Module Components

Number	*Name*
tbd	Quantum Informatics
tbd	Quantum Informatics Lab

Module Coordinators	***Program Affiliation***
Prof. Dr. Peter Schupp, Prof. Dr. Stefan Kettemann	● MSc Computer Science and Software Engineering

Entry Requirements

Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*
☒ none	☒ none	Basic linear algebra

Recommendations for Preparation

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

Content and Educational Aims

This module features a self-contained introduction to Quantum Informatics, one of the fastest growing emergent fields in science and technology, including essential elements from physics and mathematics. Topics include an overview of current quantum technology; pertinent aspects of quantum mechanics and information theory; qubits, quantum registers, quantum gates; no-cloning theorem, deferred and implicit quantum measurement; circuit model of quantum computing; quantum communication, cryptography and attacks; Grover, Shor, and further quantum algorithms; post-quantum cryptography; decoherence, quantum channels, quantum error correction; physical qubits; variational and adiabatic quantum computing, quantum annealing; quantum simulation; quantum programming and quantum SDKs.

The lectures are complemented by a lab where concepts are further deepened and practically applied. Part of the lab will be in precept-style with exercises; part will involve hands-on practical experience including mini projects.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Discuss the state of the art of quantum computing and quantum communication.
2. Apply the principles of quantum theory to analyze quantum circuits.
3. Develop quantum algorithms and quantum communication protocols.
4. Assess applications of quantum informatics.

Indicative Literature

Michael A. Nielsen, Isaac L. Chuang: Quantum Computation and Quantum Information (10th Anniversary Edition), Cambridge University Press, 2010

N. David Mermin: Quantum Computer Science: An Introduction, Cambridge University Press, 2007

Usability and Relationship to other Modules

Module Component Examinations

Module Component 1: Final Exam

Assessment Type: Written examination Duration/length: 120 min

Weight: 50%

Scope: all ILOs (focus on theory).

Module Component 2: Lab Assessment

Assessment Type: Portfolio (Graded Exercises, Project Work)

Weight: 50%

Scope: all ILOs (focus on practical application).

4.3 Management modules

4.3.1 Agile Product Development and Design

Module Name Agile Product Development and Design	***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components			
Number	*Name*	*Type*	*CP*
tbd	Agile Product Development and Design	Lecture	5
Module Coordinator Prof. Dr. Tilo Halaszovich	***Program Affiliation*** ▪ MSc Computer Science, Software Engineering and Leadership	***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements *Pre-requisites* ▪ none	*Co-requisites* ☒ None *Knowledge, Abilities, or Skills*	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ▪ Lecture (80 hours) ▪ Private study (45 hours)
		Duration 1 semester	***Workload*** 125 hours
Recommendations for Preparation N.A.			
Content and Educational Aims 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 the unpredictable is			

common, and where managers need to focus on achieving goals rather than completing 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

Upon completion of this module, students will be able to:

1. Develop practical knowledge and management skills, and mind sets to master the challenges from an agile business environment.
2. Understand (routine) business processes in various context and how to adapt business processes to an agile business environment (agile Business Process Management).
3. Summarize and classify the new data- and customer-driven technologies in a business context.
4. 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 innovative creativity methods and processes for product and software development (Design Thinking).
6. Adapt to a new working culture based on a user-centricity, empathy, and playful testing of new products and services.

Indicative Literature

Service Dominant Logic

Vargo, S.L., and 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

Business Process Management and agile Management

Daniel Paschek, D., Frank Rennung, F., Trusculescu, A., Draghici,A. (2016). Corporate Development with Agile Business Process Modelling 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.

Design Thinking

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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentation Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

4.3.2 Product Innovation and Marketing

Module Name Product Innovation and Marketing	***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 5
Module Components			
Number	*Name*	*Type*	*CP*
tbd	Product Innovation and Marketing	Lecture	5
Module Coordinator Prof. Dr. Tilo Halaszovich	***Program Affiliation*** ▪ MSc Computer Science, Software Engineering and Leadership	***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements *Pre-requisites* ▪ none *Co-requisites* ☒ None *Knowledge, Abilities, or Skills*	***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ▪ Lecture (80 hours) ▪ Private study (45 hours)	
	Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation N.A.			
Content and Educational Aims 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 those of new ventures.			
Intended Learning Outcomes			

Upon completion of this module, students will be able to:

1. Understand the innovation process, particularly in technology domains.
2. Understand the commercialization process, particularly in technology domains.
3. Analyze how value can be created and appropriated through innovation.
4. Understand and apply tools, methods and concepts to manage the commercialization process.

Indicative Literature

Gruber/Tal (2017). Where to Play: 3 Steps for Identifying your Most Valuable Market Opportunities, Financial Times/Pearson.

Mohr, J. et al. (2013). Marketing of high-technology products and innovations. Pearson Education.

Moore, G. A. (2014). Crossing the chasm. Harper Business.

Schilling, M.A. (2019). Strategic Management of Technological Innovation. McGraw-Hill.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentation Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

4.3.3 Transformational Change Management

Module Name	***Module Code***	***Level (type)***	***CP***
Transformational Change Management	tbd	Year 2	5

Module Components			
Number	*Name*	*Type*	*CP*
tbd	Transformational Change Management	Lecture	5

Module Coordinator	***Program Affiliation***	***Mandatory Status***
Prof. Dr. Tilo Halaszovich	▪ MSc Computer Science, Software Engineering and Leadership	Mandatory for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	▪ Lecture (80 hours) ▪ Private study (45 hours)
▪ none	☒ None		***Duration***	***Workload***
			1 semester	125 hours

Recommendations for Preparation
N.A.

Content and Educational Aims
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 before; again, a waste. 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 course 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

Upon completion of this module, students will be able to:

1. Understand, evaluate, and apply different leadership styles.
2. Understand and evaluate the change process in organizations.
3. Understand and apply communications and influencing.
4. Evaluate their role in a change situation.
5. Assess the stakeholders in any change context.
6. Lead or be part of an organizational change effort.

Indicative Literature

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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentation Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

4.4 Leadership / academic skills modules

4.4.1 Entrepreneurship and Intrapreneurship

Module Name Entrepreneurship and Intrapreneurship			***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 2.5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Entrepreneurship and Intrapreneurship			Lecture	2.5
Module Coordinator Prof. Dr. Tilo Halaszovich	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements *Pre-requisites* ▪ none	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills*	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ▪ Lecture (17.5 hours) ▪ Private study (45 hours)	
			Duration 1 semester	***Workload*** 62.5 hours	
Recommendations for Preparation N.A.					
Content and Educational Aims 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 model 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 society at large. As such, having an insight into the specific challenges and solutions that characterize entrepreneurship has broad 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 that approaches 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. 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, so our approach is based on the objective of having students experience entrepreneurship.

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

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Understand the essence of entrepreneurship.
2. Assess and develop a business case.
3. Analyse and identify new venture opportunities in a more systematic way.
4. Understand the importance of a business model for new venture creation.
5. Evaluate the viability of a new venture idea.
6. Understand how to finance a new venture.
7. Create and present a business case for a new venture.

Indicative Literature

Clarysse, B., Kiefer, S. The Smart Entrepreneur. Elliott and Thompson, 2011.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentations Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

4.4.2 Communication and Presentation Skills for Executives

Module Name Communication and Presentation Skills for Executives	***Module Code*** MDE-CAR-01	***Level (type)*** Year 1	***CP*** 2.5

Module Components			
Number	*Name*	*Type*	*CP*
MDE-CAR-01	Communication and Presentation Skills for Executives	Seminar	2.5

Module Coordinator Prof. Dr. Stefan Kettemann	***Program Affiliation*** ▪ MSc Computer Science, Software Engineering and Leadership	***Mandatory Status*** Mandatory for CSSEL

Entry Requirements *Pre-requisites* ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ▪ Analysis, Basic Calculus, and Linear Algebra	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ▪ Seminar (17.5 hours) ▪ Private study (45 hours)
			Duration 1 semester	***Workload*** 62.5 hours

Recommendations for Preparation Read the Syllabus

Content and Educational Aims An executive career in an international business environment requires excellent communication and presentation skills. Managers must communicate effectively to 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.

Intended Learning Outcomes

Upon completion of the module, students will be able to

1. Act as effective communicators – in both group and individual situations.
2. Understand interpersonal communication models and group dynamics in presentations.
3. Understand the importance of building rapport and trust with audiences.
4. Use presentation software (PowerPoint, Prezi) confidently and in a visually pleasant way.
5. Learn how to structure presentations in a coherent manner and develop captivating narratives.
6. Work with different presentation formats (Ignite, Pecha Kucha, Pitching etc.).
7. Understand and apply the basics of logical reasoning in oratory (deductive/inductive).
8. Develop oratory and rhetorical skills drawing on Aristotle’s teaching of logos, ethos and pathos.
9. Understand and apply the basics of interpersonal communication (Johari Window, 4-Ears model etc.).
10. Present themselves in different business situations.
11. Collaborate effective in intercultural teams.

Indicative Literature

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Oral Presentation Duration: 15 minutes

Weight: 100%

Scope: All intended learning outcomes of this module.

4.4.3 Organizational Behavior

Module Name			***Module Code***	***Level (type)***	***CP***
Organizational Behavior			tbd	Year 1	2.5
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Organizational Behavior			Lecture	2.5
Module Coordinator	***Program Affiliation***			***Mandatory Status***	
Prof. Dr. Christian Stamov Roßnagel	▪ MSc Computer Science, Software Engineering and Leadership			Mandatory for CSSEL	
Entry Requirements			***Frequency***	***Forms of Learning and Teaching***	
Pre-requisites ▪ none	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills*	Annually (Spring)	▪ Lecture (17.5 hours) ▪ Private study (45 hours)	
			Duration	***Workload***	
			1 semester	62.5 hours	
Recommendations for Preparation					
N.A.					

Content and Educational Aims

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.

Intended Learning Outcomes

Upon completion of this module, you will be able to:

1. Explain basic principles of individuals’ and groups’ behaviours in organisations.
2. Apply established theories to assessing and predicting behaviour.
3. Describe core techniques of influencing and modifying behaviour.
4. Critically discuss selected approaches to effectively lead employees, teams, and groups.

Indicative Literature

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

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentations Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

Completion: To pass this module, the examination of each module component must be passed with at least 45%.

4.4.4 Academic Writing Skills / Intercultural Training

Module Name Academic Writing Skills/Intercultural Training			***Module Code*** MDE-CAR-02	***Level (type)*** Year 1	***CP*** 2.5
Module Components					
Number	*Name*			*Type*	*CP*
MDE-CAR-02	Academic Writing Skills / Intercultural Training			Seminar	2.5
Module Coordinator Prof. Dr. Stefan Kettemann	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements *Pre-requisites* ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ☒ None	***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ● Lectures (17.5 hours) ● Private Study (45 hours)	
			Duration 1 semester	***Workload*** 62.5 hours	

Recommendations for Preparation

Read the Syllabus.

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

Content and Educational Aims

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 a 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 the scholarly and intellectual standards set by Constructor. 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 or Switzerland. Here the students will find information on employment and how to get access to the German and Swiss labor market. The seminar also provides an overview of labor conditions in Germany and Switzerland, the multifaceted forms of employment, business cultures, and useful tips and information for the job entry in a German or Swiss company.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Structure their ideas to write clear summaries, coherent paragraphs and cohesive literature reviews.
2. Write different segments of an academic paper employing writing styles that display advanced grammar and precise and concise language use.
3. Successfully find and evaluate sources for research.
4. Use citation and referencing styles applicable for their discipline.
5. Avoid unintentional plagiarism and adhere to the code of academic integrity.
6. Understand labor conditions in Germany and Switzerland.
7. Understand the typical business cultures in German or Swiss companies.

Indicative Literature

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Term Paper (Report) Length: 10 pages

Weight: 100%

Scope: All intended learning outcomes of this module.

Agile Leadership and Strategic Management

Module Name	***Module Code***	***Level (type)***	***CP***
Agile Leadership and Strategic Management	tbd	Year 2	2.5

Module Components

Number	*Name*	*Type*	*CP*
tbd	Agile Leadership and Strategic Management	Lecture	2.5

Module Coordinator Prof. Dr. Tilo Halaszovich	***Program Affiliation*** ▪ MSc Computer Science, Software Engineering and Leadership	***Mandatory Status*** Mandatory for CSSEL

Entry Requirements			***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ▪ Lecture (17.5 hours) ▪ Private study (45 hours)
Pre-requisites ▪ none	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills*	***Duration*** 1 semester	***Workload*** 62.5 hours

Recommendations for Preparation

N.A.

Content and Educational Aims

This module focuses on key strategic aspects of the leadership and strategy development processes, specifically strategic problems solving, alignment, engagement and copying with black swans and paradigm shifts. The module draws on 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. Understand and analyse business strategies.

2. Understand and analyse strategic statements and levels of ambition. 3. Understand opportunities and threats on the external environment. 4. Evaluate sources of competitive advantage as well as strategic strengths and weaknesses. 5. Analyse core challenges of agile leadership and strategy development. 6. Develop and communicate strategic initiatives. 7. Apply this knowledge to real-world strategic planning processes.
Indicative Literature Sola, D. and Couturier, J, 2013, How To Think Strategically, FT Publishing International.
Usability and Relationship to other Modules
Examination Type: Module Examination Assessment Type: Presentations Duration: 30 min Weight: 100% Scope: All intended learning outcomes.

4.4.5 Customer-Centric Mindset and Agile Delivery Management

Module Name		***Module Code***	***Level (type)***	***CP***
Customer-centric Mindset and Agile Delivery Management		tbd	Year 2	2.5
Module Components				
Number	*Name*		*Type*	*CP*
tbd	Customer-centric Mindset and Agile Delivery Management		Lecture	2.5
Module Coordinator	***Program Affiliation***		***Mandatory Status***	
Prof. Dr. Tilo Halaszovich	▪ MSc Computer Science, Software Engineering and Leadership		Mandatory for CSSEL	
Entry Requirements		***Frequency***	***Forms of Learning and Teaching***	
Pre-requisites ▪ none	*Co-requisites* ☒ None *Knowledge, Abilities, or Skills*	Annually (Fall)	▪ Lecture (17.5 hours) ▪ Private study (45 hours)	
		Duration	***Workload***	
		1 semester	62.5 hours	
Recommendations for Preparation				
N.A.				
Content and Educational Aims				
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 can utilize 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.

Intended Learning Outcomes

Upon completion of this module, students will be able to:

1. 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 various types of policies that can be employed in guiding market centric activities.
3. 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 different business models and understand how the marketing function can be employed to enhance them.

Indicative Literature

Chernev, A., 2018, Strategic Marketing Management.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Type: Presentations Duration: 30 min

Weight: 100%

Scope: All intended learning outcomes.

4.5 Research project, Capstone project, and Master’s thesis

4.5.1 Research Project

Module Name Research Project		***Module Code*** tbd	***Level (type)*** Year 2	***CP*** 5
Module Components				
Number	*Name*		*Type*	*CP*
tbd	Research Project		Project	5
Module Coordinator Prof. Dr. Bertrand Meyer	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership		***Mandatory Status*** Mandatory elective for CSSEL	
Entry Requirements *Pre-requisites* ☒ none	*Co-requisites* ☒ none *Knowledge, Abilities, or Skills*	***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Research group meetings (21 hours) ● Independent project work (104 hours)	
		Duration 1 semester	***Workload*** 125 hours	
Recommendations for Preparation				
Content and Educational Aims 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

Upon completion of this module, students will be able to:

4. Understand state-of-the-art research papers in a chosen field of specialization.
5. Plan a research project to reproduce research results or to extend ideas of recent research results.
6. Explain research questions and choose suitable methodologies to address them.
7. Document a research project in the style of a typical scientific paper.

Indicative Literature

- Recent publications provided by the research project supervisors.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment: Project report (5000 words) Weight: 100%

Scope: All intended learning outcomes of the module.

4.5.2 Capstone Project 1

Module Name	***Module Code***	***Level (type)***	***CP***
Capstone Project 1	tbd	Year 1	5

Module Components			
Number	*Name*	*Type*	*CP*
tbd	Capstone Project 1	Project	5

Module Coordinator	***Program Affiliation***	***Mandatory Status***
Prof. Dr. Manuel Oriol	● MSc Computer Science, Software Engineering and Leadership	Mandatory for CSSEL

Entry Requirements			***Frequency***	***Forms of Learning and Teaching***
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*	Annually (Fall)	● Project group meetings (42 hours) ● Group-based and independent project work (83 hours)
☒ None	☒ None	● Programming skills in an imperative language at CS bachelor level ● Algorithms and data structure at CS bachelor level	***Duration*** 1 semester	***Workload*** 125 hours

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

Content and Educational Aims
This series of Capstone modules gives the possibility of experiencing knowledge and expertise learned in the master's by a posteriori analysis, transformational adaptation, and coherent

planning with 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 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 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.

Intended Learning Outcomes

1. Create and propose mocks.
2. Perform requirements elicitation.
3. Prototype.
4. Approach customers and users.
5. Specify user stories.
6. Organize themselves through collaborative tools.
7. Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

Agile the good the hype and the ugly. Book by Bertrand Meyer

The Five Dysfunctions of a Team. Book by Patrick Lencioni

Group dynamics and Teams interventions. Book by Timothy M. Franz

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*

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.

Examination Type: Module Component Examinations

Assessment: Project		Weight: 100%

Scope: All intended learning outcomes of the module.

4.5.3 Capstone Project 2

Module Name Capstone Project 2			***Module Code*** tbd	***Level (type)*** Year 1	***CP*** 15
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Capstone Project 2			Project	15
Module Coordinator Prof. Dr. Manuel Oriol	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements			***Frequency*** Annually (Spring)	***Forms of Learning and Teaching*** ● Project group meetings (42 hours) ● Group-based and independent project work (83 hours)	
Pre-requisites ☒ None	*Co-requisites* ☒ None	*Knowledge, Abilities, or Skills* ● Programming skills in an imperative language at CS bachelor level ● Algorithms and data structure at CS bachelor level	***Duration*** 1 semester	***Workload*** 125 hours	
Recommendations for Preparation Train and advance programming, read about agile development, watch videos on ideation processes, and read books on team and teamwork.					
Content and Educational Aims This series of courses gives the possibility of experiencing knowledge and expertise learned in the master's 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 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.

Intended Learning Outcomes

1. Describe and defend a software architecture.
2. Code in groups.
3. Code as a large team.
4. Integrate independent works.
5. Use a source code versioning system.
6. Specify user stories.
7. Hold practical discussions with stakeholders.
8. Organize themselves through collaborative tools.
9. Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

Agile the good the hype and the ugly. Book by Bertrand Meyer

The Five Dysfunctions of a Team. Book by Patrick Lencioni

Group dynamics and Teams interventions. Book by Timothy M. Franz

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*	
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.	
Examination Type: Module Component Examinations Assessment: Project Scope: All intended learning outcomes of the module.	 Weight: 100%

4.5.4 Capstone Project 3

Module Name Capstone Project 3			***Module Code*** tbd	***Level (type)*** Year 1 and 2	***CP*** 15
Module Components					
Number	*Name*			*Type*	*CP*
tbd	Capstone Project			Project	15
Module Coordinator Prof. Dr. Manuel Oriol	***Program Affiliation*** ● MSc Computer Science, Software Engineering and Leadership			***Mandatory Status*** Mandatory for CSSEL	
Entry Requirements			***Frequency*** Annually (Fall)	***Forms of Learning and Teaching*** ● Project group meetings (42 hours) ● Group-based and independent project work (83 hours)	
Pre-requisites	*Co-requisites*	*Knowledge, Abilities, or Skills*			
☒ None	☒ None	● Programming skills in an imperative language at CS bachelor level ● Algorithms and data structure at CS bachelor level	***Duration*** 1 semester	***Workload*** 125 hours	
Recommendations for Preparation Train and advance programming, read about agile development, watch videos on ideation processes, and read books on team and teamwork.					
Content and Educational Aims This series of courses gives the possibility of experiencing knowledge and expertise learned in the master's 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 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 practices.

Intended Learning Outcomes

1. Know practical cybersecurity.
2. Hold practical discussions with stakeholders.
3. Practice of machine learning.
4. Work with continuous improvements tools.
5. Organize themselves through collaborative tools.
6. Understand team dynamics and resolve most interpersonal issues.

Indicative Literature

Agile the good the hype and the ugly. Book by Bertrand Meyer

The Five Dysfunctions of a Team. Book by Patrick Lencioni

Group dynamics and Teams interventions. Book by Timothy M. Franz

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*

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.	
Examination Type: Module Component Examinations Assessment: Project Scope: All intended learning outcomes of the module.	Weight: 100%

4.5.5 Master‘s Thesis

<table>
<tr><td colspan="3"><b><i>Module Name</i></b>
Master Thesis</td><td><b><i>Module Code</i></b>
tbd</td><td><b><i>Level (type)</i></b>
Year 2</td><td><b><i>CP</i></b>
30</td></tr>
<tr><td colspan="6"><b><i>Module Components</i></b></td></tr>
<tr><td><i>Number</i></td><td colspan="3"><i>Name</i></td><td><i>Type</i></td><td><i>CP</i></td></tr>
<tr><td>tbd</td><td colspan="3">Master Thesis</td><td>N.A.</td><td>30</td></tr>
<tr><td>tbd</td><td colspan="3">Colloquium</td><td></td><td></td></tr>
<tr><td><b><i>Module Coordinator</i></b>
Prof. Dr. Bertrand Meyer</td><td colspan="3"><b><i>Program Affiliation</i></b>
● MSc Computer Science, Software Engineering and Leadership</td><td colspan="2"><b><i>Mandatory Status</i></b>
Mandatory for CSSEL</td></tr>
<tr><td rowspan="2"><b><i>Entry Requirements</i></b>
<i>Pre-requisites</i>
● None</td><td rowspan="2"><i>Co-requisites</i>
☒ None</td><td rowspan="2"><i>Knowledge, Abilities, or Skills</i>
● Proficiency in the area of the chosen thesis topic.</td><td><b><i>Frequency</i></b>
Annually (Spring)</td><td colspan="2"><b><i>Forms of Learning and Teaching</i></b>
● Private Study (725 hours)
● Colloquium (25 hours)</td></tr>
<tr><td><b><i>Duration</i></b>
1 semester</td><td colspan="2"><b><i>Workload</i></b>
750 hours</td></tr>
<tr><td colspan="6"><b><i>Recommendations for Preparation</i></b>
Read the Syllabus.</td></tr>
</table>

Content and Educational Aims

The aim of this module is to train students to motivate, design, carry out, and document a 6-month 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. 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.

Intended Learning Outcomes

Discipline-Specific Skills (subject area depending on individual project):

1. Understanding, at a professional level, of a circumscribed segment of the project in its environment (research, industry, start-up).
2. Ability to apply specific and selected CSSEL techniques, as required for the project, at a professional level.
3. General professional skills.
4. Designing and carrying out the full cycle of a project by scientific means in a professional manner.
5. 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. Presentation of project results for specialists and non-specialists.

Indicative Literature

N.A.

Usability and Relationship to other Modules

Examination Type: Module Examination

Assessment Component 1: Thesis Length: 30 – 60 pages

Weight: 80%

Scope: All intended learning outcomes of this module.

Assessment Component 2: Oral Examination (Defense) Duration: 20 minutes

Weight: 20%

Scope: Mainly presentation of project results but the presentation touches all intended learning outcomes

Completion: This module is passed with an assessment-component weighted average grade of 45% or higher.

5. Appendix

5.1 Intended Learning Outcomes Assessment Matrix

Computer Science and Software Engineering (MSc.)					Software Construction, Software Architecture and Software Engineering	Quality Engineering	Architectural Strategy	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	Software Engineering Modules	Cybersecurity Modules	Artificial Intelligence Modules	Application Modules	Capstone Project 1	Capstone Project 2	Capstone Project 3	Masster Thesis
Semester					1	1	2	1	2	3	1	1	2	2	3	3	1-3	1-3	1-3	1-3	1	2	3	4
Mandatory/optional					m	m	m	m	m	m	m	m	m	m	m	m	me	me	me	me	m	m	m	m
Credits					5	5	5	5	5	5	2.5	2.5	2.5	2.5	2.5	2.5	0-15	5-20	5-20	0-5	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	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
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
Design, implement and exploit methods in cryptography and security related fields;	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
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
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
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
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	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	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	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	x
Apply their knowledge and understanding to a professional context;	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
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	x
Assessment Type																								
Oral examination																								x
Written examination																		x	x					
Project																								
Term paper														x				x						
Report																			x					
Poster presentation																								
Presentation								x	x	x	x	x	x		x	x								
Various					x	x	x										x			x	x	x	x	
Thesis																								x

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

Computer Science and Software Engineering (MSc.)					Advances in Software Engineering	Parallel and Distributed Computing	Advanced Databases	Cryptography	System Security	Network Security	Cybercriminology	Deep Learning	Intelligent Autonomous Systems	Artificial Intelligence	Text Analysis and Natural Language Processing	Data Analytics	Machine Learning	Quantum Informatics	Research Project
Semester					3	1/3	2	1	2	3	1/ 3	1/ 3	1/3	2	2	1	2	tba	3
Mandatory/ optional					me	me	me	me	me	me	me	me	me	me	me	me	me	me	me
Credits					5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	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
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
Design, implement and exploit methods in cryptography and security related fields;	x	x						x	x	x	x							x	x
Apply cross-disciplinary management methodologies to solve academic and professional problems;	x	x	x																x
Critically assess and integrate a consistent tool set of leadership abilities into a professional work environment;	x	x	x																x
Plan, conduct and document small research projects in the context of computer science and software engineering;	x	x	x																x
Independently research, document and present a scientific topic with appropriate language skills;	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
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
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
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
Apply their knowledge and understanding to a professional context;	x	x	x																x
Take on responsibility in a diverse team;		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
Assessment Type																			
Oral examination																			
Written examination						x	x	x	x	x		x	x	x			x		
Project																			
Term paper											x								
Report							x								x	x			x
Poster presentation																			
Presentation																			
Various					x														
Thesis																			

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

C›ONSTRUCTOR
INSTITUTE

Constructor Institute Schaffhausen
Rheinweg 9 · 8200 Schaffhausen · Switzerland
+ 41 52 630 28 38
institute.constructor.org