Bring ideas to life
VIA University College

Date: August 2021

Rev. August 2022
Technical correction:
Oct. 2022 (appendix 2 SDJ3)

Curriculum

Programme section

Bachelor of Engineering in Software Technology Engineering

Applicable to students enrolled in August 2021 and later.

However, students enrolled in summer 2021 and winter 2022 will follow the study plan in Appendix 1.

(For students enrolled before summer 2021, please refer to the 2017 curriculum).

CONTENTS

Introduction

In accordance with the Executive Order on Bachelor of Engineering, the purpose of Bachelor of Engineering is to qualify the students to carry out the following professional functions nationally and internationally:

- Translate technical research results as well as scientific and technical knowledge into practical use in development tasks and in solving technical problems
- Critically acquire new knowledge within relevant engineering areas
- Independently solve common engineering tasks
- Plan, implement and manage technical and technological facilities, including being able to involve societal, economic, environmental and occupational health and safety consequences in the solution of technical problems
- Participate in collaborative and managerial functions and contexts at a qualified level with people who have different educational, linguistic and cultural backgrounds

In addition, the education must qualify students to participate in further studies.

VIA's engineering programmes work on the basis of a common graduate profile. The graduate profile is a common profile for all VIA Engineers. The graduate profile is to be combined with the identity of the specific engineering programme.

At VIA Engineering, we are practice-oriented, project-oriented and world-focused. This is actualised in the form of qualified new graduates obtained through targeted teaching, relevant research and development, as well as collaboration and ongoing dialogue with the business community. The programmes must qualify graduates to handle practical and development-oriented business functions.

Programmes in English as well as admission of international students are hallmarks of our engineering programmes. This profile creates a unique opportunity to educate students who can act in a Danish context in an increasingly global market. Our lecturers have vast and solid practical experience and know how to anchor theory in practice through lab work, company visits and projects for and in collaboration with companies.

1 Identity of the programme

The Software Technology Engineering at VIA is a study programme at the bachelor level. It is an applied engineering degree giving students skills and competences to be employed as software engineers after graduation. Software Technology Engineering graduates are qualified to:

- Apply research, theory, tools and methods from software engineering and natural science to conceive, design and implement solutions to practical engineering problems.
- Critically acquire new knowledge within the field of software engineering
- Consider the social, financial, and environmental consequences of the suggested solutions
- Work independently as well as in teams with members from different educational and cultural backgrounds

The goals of the programme are achieved primarily by:

- Project work being an essential aspect of the teaching, where the academic elements of the programme are integrated via problem solving into a whole, with a focus on application-oriented and practical engineering work. In project work, emphasis is also placed on the students developing academic, professional, methodological, communicative and personal skills.
- Collaborating with research environments and businesses in connection with the implementation of the teaching.
- Offering an international study environment, where parts of the study can be completed abroad, and where several courses are taught in English for Danish and foreign students alike.
- Actively using the student's engineering internship to bring about the exchange of knowledge and experiences between VIA and the profession.
- Achieving application- and practice-oriented skills primarily by utilising VIA's facilities within laboratories, manufacturing workshops and libraries, as well as completing internships and workshops.
- Priority being given to interdisciplinary focus areas within Digitalisation, Sustainability and Innovation and Entrepreneurship in the programme across the various semesters.

2 Graduate profile for VIA Engineers

Purpose
The newly graduated VIA engineer works problem-oriented, project- and team-based and contributes to advising, developing, inventing and quality-assuring products and solutions. The VIA engineer creates innovative, digital, sustainable and workable solutions to and for current and future societal and engineering challenges worldwide.

Skills
VIA Engineering educate holistic-thinking engineers who, through societal insight and personal development, can exploit the full potential of technology. Therefore, the skills of the VIA engineer range from highly specialised engineering skills to personal skills and the skills of the outside world.

Professional engineering skills

- Masters and applies – with critical reflection – highly specialised engineering knowledge.
- Works challenge-driven, innovative and problem-oriented when developing engineering results.
- Integrates engineering and scientific knowledge, skills and methods in solving engineering challenges.
- Designs, plans, simulates, manages, implements and evaluates engineering solutions and products using digital and technological tools.
- Implements and operates solutions that match engineering needs within the industry.

Organisational skills

- Organises and manages projects and processes based on both risk assessment and market and business understanding.
- Collaborates inter-professionally with a global view and respect for the organisation, culture and methods of businesses and stakeholders.
- Involves knowledge of sustainability and circular economy in the development and implementation of new solutions.

Personal skills

- Works consistently with a curious and innovative mindset and seeks out, critically acquires and brings new knowledge into play throughout life.
- Communicates effectively and collaborates professionally with colleagues and people of different educational and cultural backgrounds.

3 Teaching and working methods

The engineering programme’s priority focus areas within Digitalisation, Sustainability and Innovation and Entrepreneurship are integrated into relevant courses, so that together they constitute learning streams for all three areas.

Active and practice-oriented learning is supported by:

- Dialogue-based teaching with a high degree of active participation from students.
- Lectures in subjects where there is a large proportion of knowledge transfer. Lectures are usually combined with practice sessions with a student instructor or a lecturer.
- Project work and problem-oriented learning (PBL) are an essential part of the teaching, as the academic elements of the education programme are integrated into application-oriented engineering projects with emphasis on methodological problem solving.
- Projects being carried out in groups within the programme and in an interdisciplinary collaboration with other engineering programmes.
- Collaborating with research environments and businesses in connection with the implementation of the teaching.
- Offering an international study environment, where parts of the study can be completed abroad and where several courses are held in English for Danish and foreign students alike.
- The student’s engineering internship being actively used to bring about the exchange of knowledge and experiences between VIA and the profession.

Application- and practice-oriented skills are primarily achieved by utilising VIA’s facilities within laboratories, manufacturing workshops and library.

Teaching can be physical, online or located at another campus.

4 Structure and content

The programme is organised as an ordinary full-time higher education programme.
The structure and progression are stated in the overview on page 7.

The official duration of the programme is 3½ years, divided into 7 semesters of 30 ECTS, corresponding to 210 ECTS points in total.

The scope of each course or project is documented in the form of ECTS points (European Credit Transfer System). 1 ECTS point corresponds to a workload of approx. 27.5 hours for a student, an academic year of 60 ects thus corresponds to 1,650 hours of work for the student.

New students are admitted in August every year.
Up to and including 2022, new students were also admitted in February.

The study includes:

- Compulsory courses and projects
- Elective courses
- Internship
- Bachelor project
- Workshops

A semester consists of 3-4 courses, which are delimited courses. A course's scope can range from 5 to 10 ECTS points, and a project's scope from 10 to 15 ECTS points.

The purpose, scope, learning objectives and exams of courses are described in this curriculum. For a detailed and complete description of the individual courses, please refer to the course descriptions in force at any given time, which are available on VIA's website and on VIA's Studynet.

The programme is structured as illustrated below:

Semester ***Theme***	**Course** **5 ECTS**	**Course** **5 ECTS**	**Course** **5 ECTS**	**Course/ Project** **5 ECTS**	**Project** **5 ECTS**	**Project** **5 ECTS**
7 semester ***Electives***	**Elective Course**	**Elective Course**	**Elective Course**	**BPR2** Bachelor Project		
6 semester ***Innovation and Electives***	**Elective Course**	**Elective Course**	**Elective Course**	**BPR1** Bachelor Project Preparation	**SEP6** Semester Project (Innovation)	
5 semester ***Internship***	**INP1** Internship					
4 semester ***Internet-of-Things***	**ADS1** Algorithms and Data Structures	**ESW1** Embedded Software	**WEB2** Web Development 2	**DOC1** DevOps and Cloud	**SEP4** Semester Project	
3 semester ***Heterogeneous Systems***	**SDJ3** Software Development with UML and Java 3	**CAO1** Computer Architecture and Organisation	**DNP1** .NET Programming	**NES1** Networking and Security	**SEP3** Semester Project	
2 semester ***Client/Server Systems***	**SDJ2** Software Development with UML and Java 2		**SWE1** Software Engineering	**DBS1** Database Systems	**SEP2** Semester Project	
1 semester ***Single User Systems***	**SDJ1** Software Development with UML and Java 1		**WEB1** Web Development 1	**DMA1** Discrete Mathematics and Algorithms	**SEP1** Semester Project	

Depending on the choice of electives, students may specialise in one of three areas:

- Internet of Things
- Interactive Media
- Data Engineering

5 Compulsory elements of the programme, 1st – 4th semester

All courses on the first 4 semesters are compulsory, and they all include a semester project. The overall goal of the semester projects is to connect the courses and for students to apply the skills acquired during the semester. Project methods, teamwork, communication, and documentation skills are taught in the context of the semester projects.

Each semester has a theme. The themes of the first four semesters are:

- 1st semester: Single User Systems
- 2nd semester: Client/Server Systems
- 3rd semester: Heterogeneous Systems
- 4th semester: Internet of Things

5.1 1st semester: Single User Systems

The goal of the 1st semester is to give the student knowledge and practical skills in object-oriented programming and systems development. The student will also learn about responsive web design, discrete mathematics and algorithms. The semester is organised as a number of introductory courses and a semester project in which the students will design, implement and document a single-user software system.

The purpose of the courses, ECTS and assessment:

Software Development with UML and Java 1 (SDJ1) – 10 ECTS	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
The main purpose of the course is to provide students with the qualifications needed to understand the core object-oriented concepts and to implement smaller programs in Java from UML class diagrams.	Attendance (≥ 75%) Participation in oral mid-term test	Oral exam (30 minutes) External assessment
Web Development 1 (WEB1) – 5 ECTS		
The purpose of this course is to introduce a set of theories and tools in order for students to obtain a proficient level of knowledge and gain a practical set of skills for designing and developing responsive web sites for both desktops and mobile devices using basic web programming. WEB1 was previously labelled RWD1, Responsive Web Design, running up to and including spring semester 2022.	Attendance (≥ 75%) 3 approved assignments	Written exam (2 hours) External assessment
Discrete Mathematics and Algorithms (DMA1) – 5 ECTS		
The aim of the course is to train students in the mathematical concepts and process of algorithmic thinking, allowing them to construct simpler, more efficient solutions to real-world computational problems, building on the principles of mathematics.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (3 hours) Internal assessment

Semester Project (SEP1) – 10 ECTS		
Develop and document a single user system and a basic introduction to team-based project work.	Reports handed in before deadline	Oral exam Group presentation 5 minutes per student followed by a joint exam with a joint discussion and individual question rounds for approx. 15 minutes per student including voting. Internal assessment

The learning objectives of the courses (knowledge, skills and competencies) and further information on the form and conditions of the assessment can be found in Appendix 1.

ECTS credits: 30

5.2 2nd semester: Client/Server Systems

During the 2nd semester, the student will learn about software engineering methods, database development and system development methods, and the student will accumulate the programming skills necessary to build client/server systems. The semester is organised as a number of compulsory courses, along with a semester project in which the students will develop a client/server system.

The purpose of the courses, ECTS and assessment:

Software Development with UML and Java 2 (SDJ2) – 10 ECTS	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
The purpose is to qualify the student to understand and master the concepts and techniques of object-oriented system development and programming, including Client/Server programming.	Attendance (≥ 75%) Mandatory course activities completed	Oral exam External assessment
Database Systems (DBS1) – 5 ECTS		
The course has two purposes. Firstly, students are to learn methods for designing, implementing and operating single-user relational databases. Secondly, students are to learn the main principles, architecture and technologies of a typical relational database management system (RDBMS).	Attendance (≥ 75%) Mandatory course activities completed	Written exam (4 hours) External assessment
Software Engineering (SWE1) – 5 ECTS		
The purpose is to qualify the student to apply software engineering concepts used to develop object-oriented software. Structure the software development process by applying SCRUM and Unified Process to conduct Analyse, Design and Test-descriptions to exemplify a final solution from a real-life problem. This involves requirement capturing (Use Cases and non-functional requirements), analysis, domain models, interaction diagrams, design classes, design patterns and test-descriptions etc.	Attendance (≥ 75%) Mandatory course activities completed	Oral exam Internal assessment

Semester Project (SEP2) – 10 ECTS		
The purpose is to develop and document a client/server system and demonstrate an acquisition of process skills and interpersonal competences.	Attendance (≥ 75%) Project report handed in before deadline	Oral exam Group presentation (5 minutes per student) Group exam (20 minutes per student) Internal assessment

The learning objectives of the courses (knowledge, skills and competencies) and further information on the form and conditions of the assessment can be found in Appendix 1.

ECTS credits: 30

5.3 3rd semester: Heterogeneous Systems

The aim of the 3rd semester is to design and implement heterogeneous software solutions including the use of the programming languages Assembler, C, Java and C#. The semester is built around a larger semester project in which students will integrate several programming languages.

The purpose of the courses, ECTS and assessment:

Software Development with UML and Java 3 (SDJ3) – 5 ECTS	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
The students will be introduced to basic theory of distributed systems and be able to design and implement a distributed system.	Attendance (≥ 75%) Mandatory course activities completed	Oral exam (20 minutes) External assessment
Computer Architecture and Organisation (CAO1) – 5 ECTS		
The main purpose of the course is to gain a basic understanding of the organisation and design of computers and how a computer works. Focus will be on the central processing unit (CPU) and the necessary logic involved in building a CPU.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (2 hours) External assessment
.NET Programming (DNP1) – 5 ECTS		
The purpose is to qualify the student to describe and implement the basic concepts of the C# programming language and the .NET developer platform with a focus on ASP.NET.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (3 hours) Internal assessment
Networking and Security (NES1) – 5 ECTS		
The main purpose of the course is to gain a basic understanding of computer networks, Internet protocols and security technology.	Attendance (≥ 75%)	Written exam (3 hours) Internal assessment
Semester Project (SEP3) – 10 ECTS		
The purpose is to develop and document a distributed, heterogeneous system and a network protocol, herein account for the security aspects of the system and the protocol.	Attendance (≥ 75%) Project report handed in before deadline	Oral exam Group exam (approx. 20 minutes) Individual exam (approx.15 minutes per student) Internal assessment

The learning objectives of the courses (knowledge, skills and competencies) and further information on the form and conditions of the assessment can be found in Appendix 1.

ECTS credits: 30

5.4 4th semester: Internet of Things

The 4th semester brings it all together and students will conceive, design and implement a software solution including hardware sensors, a web-based user interface and a persistent multiuser backend infrastructure. The solution must contain self-constructed electronics, and make use of the Java, C#, C and assembler programming languages.

The purpose of the courses, ECTS and assessment:

WEB Development 2 (WEB2) – 5 ECTS	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
The purpose of the course is to introduce the students to modern front-end web development, by going in depth with the JavaScript language & front-end JavaScript frameworks.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (3 hours) External assessment
Embedded Software (ESW1) – 5 ECTS		
The purpose is to qualify the student to apply basic concepts in real-time programming, and to implement real-time programs using the C-programming language on embedded micro-controllers and using interfaces (APIs) to a number of sensors and actuators.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (3 hours) Internal assessment
Algorithms and Data Structures (ADS1) – 5 ECTS		
The purpose is to qualify the student to design, implement and analyze different algorithms and to become acquainted with different advanced data structures	Attendance (≥ 75%) Mandatory course activities completed	Written exam (3 hours) Internal assessment
DevOps and Cloud (DOC1) – 5 ECTS		
The purpose of the course is to introduce students to a set of practices that help deliver software in a fast and reliable manner. By automating workflows and integrating the processes of software development and IT operations, the students will be able to create a consistent toolchain that supports planning, building and operating a software project.	Attendance (≥ 75%) Mandatory course activities completed	Written exam (1 hour) Internal assessment
Semester Project (SEP4) – 10 ECTS		
Conceive, design and implement a software solution including hardware sensors, a web-based user interface and a persistent multi user backend infrastructure. The integration and delivery of the software will be managed through the implementation of a deployment pipeline.	Project- and process report handed in before deadline	Group exam with individual assessment. Group presentation (~30 minutes) followed by a joint examination of 3-4 students from the group at a time (~15 minutes per student). Internal assessment

The learning objectives of the courses (knowledge, skills and competencies) and further information on the form and conditions of the assessment can be found in Appendix 1.

ECTS credits: 30

6 Internship, 5th semester

IT-INP1

The internship comprises a semester of 30 ECTS and is timewise placed in the 5th semester of the programme. As a general rule, the internship period is paid and settled in a private or public company in Denmark or abroad. Student must be on an internship for a minimum of 20 full weeks excluding holidays, etc.

The purpose of the internship is for the student to acquire insight into practical common engineering work corresponding to engineering assistant work, combined with the integrated application of the acquired concepts, methods and techniques of the discipline in the first four semesters.

The student will find an internship, which must be approved by VIA, who appoints a supervisor for the intern.

In collaboration with the company, the student prepares a plan for the internship with appertaining formulated assignments.

The basis for assessment of the internship is an ongoing report from the student to VIA, feedback from the internship company and a presentation where the supervisor can ask elaborating questions about the content of the internship.

If the engineering internship is interrupted before the end of the agreed internship period, the internship supervisor must, in consultation with the head of the education programme, assess whether the internship has been of sufficient length and content for there to be grounds for passing the internship present.

The internship is graded as passed/not passed. Internal assessment.

7 6th and 7th semester

The 6th and 7th semester consist of one mandatory course (BPR1), elective courses, and projects.

The software student can select one of the three specialisations, but the student can also complete their degree without a specialisation. A specialisation consists of three elective courses (15 ECTS credits), and a substantial part of the bachelor project must be within the specialisation area. Apart from those courses, a number of elective courses are offered in the Software Technology Engineering Programme. An overview of each elective course is shown below in section 8. To obtain a specialisation, the following elective courses should be selected:

Interactive Media	At least 15 ECTS should be selected from the “Interactive Media” courses in section 8.
Internet of Things	BEL1, HWP1, and at least one additional “Internet of Things” course from section 8.
Data Engineering	BUI1 and at least 10 additional ECTS should be selected from the “Data Engineering” courses in section 8.

It is also possible to choose one elective course (5 ECTS) offered by other programmes at VIA. Selecting a course from other programmes must be pre-approved by an Engineering study counsellor.

7.1 6th semester

A compulsory 10 ECTS semester project (SEP6) within the area of innovation and entrepreneurship is completed in project groups of students from different engineering programs at VIA. In addition to SEP6, the student must pass BPR1 and three electives at 6th semester.

Bachelor Project Preparation (BPR1) – 5 ECTS	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
The purpose of the course is to prepare the student for the Bachelor Project. In preparing the Bachelor Project students learn to recognise important sets of problems within the professional area, alternative solutions to them and the demands of companies and their environments. In the course, the students are taught how to apply scientific knowledge and work methods to their own field in new and changing situations. They also learn to communicate orally and in writing on questions related to the area of research, as well as methods for collecting data and testing their solutions. Each bachelor project group must consist of 2-3 students.	Mandatory course activities completed	Exam in the form of a mandatory written assignment (project description) Internal assessment
Electives		
For further details see section 8	For further details see section 8	For further details see section 8
Semester Project (SEP6) – 10 ECTS		
The purpose of SEP6 is to develop and document a cross-organisational innovation project in collaboration with a company or institution.	Mandatory course activities completed	Oral exam External assessment

7.2 7th semester

The programme is concluded with a bachelor project (BPR2) which constitutes 15 ECTS credits of the programme. The bachelor project is initiated on 6th semester (BPR1, 5 ECTS) where among other aspects the project description is completed.

The bachelor project must demonstrate individual self-critical reflection within the chosen area and must document the student's ability to apply engineering theories and methods. In addition, the bachelor project must reflect the student's ability to communicate professionally.

The requirement for attending the bachelor project (BPR2) assessment is that the student has passed all courses in the 1st - 7th semester (courses totalling 195 ECTS, including the 30 ECTS internship). The bachelor project is prepared in groups of two or three persons.

Electives	**Prerequisites for attending exam**	**Assessment** (all re-exams may be oral exam)
For further details see section 8	For further details see section 8	For further details see section 8
Bachelor Project (BPR2) – 15 ECTS		
The purpose of the Bachelor Project 2 is to evolve the student’s ability to solve a relevant software engineering problem and document	Project report handed in before deadline	Oral exam Group presentation (20 min)

the solution. In a group, students must be able to analyse, design, implement and test complex problems and be able to carry out well-documented and tested solutions. Each bachelor project group must consist of 2-3 students.		Individual assessment (20 minutes per student) External assessment As part of the examination, it is asked which parts of the report the student has primarily worked with.

8 Electives

On the Software Technology Engineering programme, the following electives are offered regardless of any specialisation:

Courses (5 ECTS)	Specialisation	Prerequisites for attending exam	Assessment (all re-exams may be oral exam)
Programming Concepts and Languages (PCL1)	None	Mandatory course activities completed	Written exam (3 hours) Internal assessment
Applied Linear Algebra (ALI1)	None	None	Written exam (3 hours) Internal assessment
Process Management for Software Engineering (PME1)	None	Mandatory course activities completed	Ongoing tests in the form of two individual written assignments, weighing 25% and 30%, exam in the form of an individual written assignment weighing 45%. Internal assessment
Compiler Construction (CMC1)	None	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
IT Security and Cryptography in Practice (SCP1)	None	Mandatory course activities completed	Midterm test on the basis of a written assignment (70%) Exam on the basis of written assignment (70%) Internal assessment
Single-page Web Applications (SWA1)	Interactive media	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Interaction Design (IDX1)	Interactive media	Mandatory course activities completed	Ongoing test in the form of one written assignment, weighing 50%. Oral exam (20 minutes) weighing 50% Internal assessment
Game Development (GMD1)	Interactive media	Mandatory course activities completed	Exam in the form of a written group assignment. Internal assessment
Digital Multi Media (DIM1)	Interactive media	Mandatory course activities completed	Exam in the form of a written group assignment. Internal assessment
XR Development (XRD1)	Interactive media	Mandatory course activities completed	Exam in the form of a written group assignment. Internal assessment

Basic Electronics (BEL1)	Internet of Things	Mandatory course activities completed	Ongoing tests in the form of three written assignments that weigh equally and make up 50% of the assessment Oral exam (20 minutes), weighing 50% Internal assessment
Digital Signal Processing (DSP1)	Internet of Things	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Embedded Operating Systems (EOS1)	Internet of Things	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Real-Time Programming (RTP1)	Internet of Things	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Internet-of-Things WAN’s (LWA1)	Internet of Things	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Hardware Oriented Programming (HWP1)	Internet of Things	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
Data Analytics Infrastructure (DAI1)	Data Engineering	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
ERP Systems SAP ABAP/4 Programming (ERP1)	Data Engineering	Mandatory course activities completed	Ongoing tests in the form of course work, weighing 50% Oral exam (20 minutes), weighing 50% Internal assessment
Machine Learning (MAL1)	Data Engineering	Mandatory course activities completed	Oral exam Internal assessment
Stochastic Processes and Modelling (SMP1)	Data Engineering	None	Written exam (3 hours) Internal assessment
Business Intelligence (BUI1)	Data Engineering	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment
No-SQL versus relational databases (NSQ1)	Data Engineering	Mandatory course activities completed	Oral exam (20 minutes) Internal assessment

The learning objectives of the courses (knowledge, skills and competencies) and further information on the form and conditions of the assessment can be found in Appendix 1.

9 Workshops

The program covers two workshops, related to SDJ1 and SDJ2, respectively:

WS1:
Initially, a test is completed which result determines whether the student must take part of WS1 or not (the test will take place within the first five weeks of 1st semester).

WS2:
Students who achieve 10 or 12 at the SDJ1-exam do not have to follow WS2. All other students must follow WS2.

In order to pass WS1 and WS2 a minimum of 75% attendance is required.

10 Bachelor Project

IT-BPR1
IT-BPR2

The programme concludes with a bachelor project (BPR2), which accounts for 15 of the programme’s total 210 ECTS and concludes with an oral exam. The bachelor project commences in the 6th semester (BPR1) with a choice of subject and preparation of a project description.

The bachelor project must demonstrate independent critical reflection within the chosen topic and must document the student’s ability to apply engineering theories and methods. The bachelor project must also reflect the student’s ability to express themselves in an academic and structured manner within their subject.

A condition for being able to commence the bachelor project is the student being assessed as being likely ready for the exam, as BPR2 must be the last exam of the study.

As a rule, the bachelor project is prepared in groups of 2-3 people.

The bachelor project includes an independent experimental, empirical and/or theoretical treatment of a practical problem in connection with the central topics of the software technology engineering programme.

The project must be documented in the form of a report containing a project basis, solution description, calculations, drawings, etc. If the report is a group assignment, it must be clear who wrote which sections of the report.

The students are examined in the project by oral exam/group exam with individual assessment in accordance with the programme’s overall goals as described in Section 1 of the Curriculum. The basis for exam is the bachelor project. It is a prerequisite for participation in the exam that the bachelor project is submitted within the stipulated deadline and meets the described criteria for the project.

Exam can take place at the earliest when all the other exams of the programme, including internships, have been passed. The exam is assessed according to the 7-point scale and with the participation of an external assessment.

11 Title and issue of degree

Graduates who have completed the programme of study according to this curriculum + joint regulations, are entitled to use the Title Bachelor of Engineering in Software Technology Engineering.

It is also possible to obtain the following special designations:

- Interactive Media
- Internet of Things
- Data Engineering

For completed programmes, VIA University College issues a diploma stating the title, programme and, if applicable, special designation. Furthermore, information is provided on the scope of the sub-elements in ECTS, the result of the assessments achieved as well as the topics for the interdisciplinary project and the graduation project. Similarly, the graduate’s basis of admission to the programme is also stated.

In the event of the programme being interrupted, proof of having passed study units will be issued.

12 Appendix 1: Study plan – intake summer 2021 and winter 2022

Semester	**Course**	**Course**	**Course**	**Course/Project**	**Project**	**Project**
Theme	**5 ECTS**	**5 ECTS**	**5 ECTS**	**5 ECTS**	**5 ECTS**	**5 ECTS**
7 semester ***Electives***	**Elective Course**	**Elective Course**	**Elective Course**	**BPR2** Bachelor Project		
6 semester ***Innovation and Electives***	**Elective Course**	**Elective Course**	**Elective Course**	**BPR1** Bachelor Project Preparation	**SEP6** Semester Project (Innovation)	
5 semester ***Internship***	**INP1** Internship					
4 semester ***Internet-of-Things***	**ADS1** Algorithms and Data Structures	**ESW1** Embedded Software	**WEB2** Web Development 2	**DOC1** DevOps and Cloud	**SEP4** Semester Project	
3 semester ***Heterogeneous Systems***	**SDJ3** Software Development with UML and Java 3	**CAO1** Computer Architecture and Organisation	**DNP1** .NET Programming	**NES1** Networking and Security	**SEP3** Semester Project	
2 semester ***Client/Server Systems***	**SDJ2** Software Development with UML and Java 2		**SWE1** Software Engineering	**DBS1** Database Systems	**SEP2** Semester Project	
1 semester ***Single User Systems***	**SDJ1** Software Development with UML and Java 1		**RWD1** Responsive Web Design	**DMA1** Discrete Mathematics and Algorithms	**SEP1** Semester Project	

13 Appendix 2: Courses Software Technology Engineering Programme

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
IT-DMA1	Discrete Mathematics and Algorithms	5	Upon completion of this course, students will be able to: • Describe fundamental concepts in number theory and modular arithmetic • Outline the basic principles of different sorting algorithms • Summarize key aspects of various data structures	Upon completion of this course, students will be able to: • Give precise arguments for the correctness or incorrectness of an algorithm • Use key concepts of discrete mathematics for solving programming problems resourcefully • Analyse and compare the time and space usage of algorithms and data structures	Upon completion of this course, students will be able to: • Adapt known algorithms and data structures to special cases of known problems or new problems • Design and implement small programs, using algorithms and data structures taught in the course . • Evaluate the performance of Java code with the objective of designing and implementing algorithms that optimise the code	Prerequisites for exam: If the following requirements are not met, the student will not qualify for the exam: i. The student must have an attendance of at least 75% . ii. The student must have 6 out of 8 mandatory assignments approved . Exam type: The course is evaluated based on a 4 hour written exam test . The test is completed in the Flow-lock browser in WISEflow . The final grade will reflect an over-all assessment of the mandatory assignments and the written exam test . The final test must, however, be awarded a passing grade in order for the final grade to be a passing grade . Internal examiner . Tools allowed: The students are allowed to use any notes, books, and/or other written exam/printed material. Any type of communication between students or between a student and an external party is prohibited and will be considered a violation of the exam rules . Re-exam: Please note that re-examinations may take a different form than the ordinary exams .
IT-RWD1 (A21 and S22)	Responsive Web Design	5	Having completed this course, students will have the knowledge to: • Describe the different file formats used in web development and their purpose . • Reproduce webpage layouts using HTML5 and CSS3 when presented with	Having completed this course, students will have the skills to: • Create web sites using Hyper Text Markup Language (HTML5) . • Use simple and advanced CSS3 selectors and properties to style webpages . • Apply the Bootstrap grid	Having completed this course, students will be able to: • Design and implement platform independent web applications .	Prerequisites for exam: If the following requirements are not met, the student will not qualify for the exam: The student must have an attendance of at least 75% . Mandatory assignments handed in before deadline and accepted. Exam type: Digital written examination duration of 2 hours (2 parts): Part 1: Multiple choice questions 30 minutes Part 2: Short answer questions 90 minutes (explaining

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			images/screenshots of other websites . • Select appropriate attributes for HTML5 elements . • Explain the difference between responsive and non-responsive websites . • Test HTML5 files for errors using the W3C markup validator . • Account for the difference between the JavaScript and Java programming languages .	framework to create responsive websites . • Utilize the Bootstrap classes to apply styling to responsive websites . • Implement JavaScript functions to add functionality to websites . • Use XMLHttp Request to read content from an external source and integrate this content into a website . • Select HTML elements and apply jQuery animations to the selected elements to make websites interactive .		and writing code) External assessment . Tools allowed: Part 1: without aids Part 2: all aids allowed - including internet connection Any types of communication between students or between a student and an external party is prohibited and will be considered a violation of the exam rules . Re-exam: Please note that re-examinations may take a different form than the ordinary exams .
IT-SDJ1	Software Development with UML and Java 1	10	The student should be able to: • Identify the Java lexical structures: keywords, separators, operators, identifiers, literals and comments . • Explain details of UML class diagrams . • Identify selection and loop structures in UML activity diagrams .	The student should be able to: Construct Java programs with proper choice of selection and loop structures .Create and use objects in Java .Implement classes in Java using the object oriented concepts: encapsulation, inheritance and polymorphism .Implement one-to-one relations and differentiate between association, aggregation and composition .Implement one-to-many relations using array structures and a simple collection class .Implement exception handling for different types of exceptions .Implement persistence in text and binary files .Construct simple event-based GUI applications .Construct Java source code documentations . Interpret UML class	The student should be able to: • Exemplify and discuss basic object-oriented concepts, including encapsulation, relationships, inheritance and polymorphism • Implement small scale systems from UML class diagrams .	Prerequisites for exam: In order to qualify for the exam, the student must participate in an oral mid-term test half way through the semester. The date will be published at the beginning of the semester. If students fail to appear for the mid-term test, they must take the mid-term test on class in the first possible session. The student must have an attendance of at least 75% in order to qualify for the exam. Students who do not have at least 75% attendance will automatically fail the ordinary exam. Exam type: Individual oral examination where the student will pick an unfamiliar programming exercise at random. The student must explain the UML involved and demonstrate how to perform the programming task using a laptop. The time allotted for the examination is 30 minutes including assessment . Exam is with an external examiner. Tools allowed: All. Re-exam:

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
				diagrams, and construct corresponding Java code .		Same as ordinary exam .
IT-WEB1	Web Development	5	Having completed this course, students will have the knowledge to: • Describe the different file formats used in web development and their purpose . • Reproduce webpage layouts using HTML5 and CSS3 when presented with images/screenshots of other websites . • Select appropriate attributes for HTML5 elements . • Explain the difference between responsive and non-responsive websites . • Test HTML5 files for errors using the W3C markup validator . • Account for the difference between the JavaScript and Java programming languages .	Having completed this course, students will have the skills to: • Create web sites using Hyper Text Markup Language (HTML5) . • Use simple and advanced CSS3 selectors and properties to style webpages . • Apply the Bootstrap grid framework to create responsive websites . • Utilize the Bootstrap classes to apply styling to responsive websites . • Implement JavaScript functions to add functionality to websites . • Use XMLHttp Request to read content from an external source and integrate this content into a website . • Select HTML elements and apply jQuery animations to the selected elements to make websites interactive .	Having completed this course, students will be able to: • Design and implement platform independent web applications .	Prerequisites for exam: If the following requirements are not met, the student will not qualify for the exam: The student must have an attendance of at least 75%. Mandatory assignments handed in before deadline and accepted. Exam type: Digital written examination duration of 2 hours (2 parts): Part 1: Multiple choice questions 30 minutes Part 2: Short answer questions 90 minutes (explaining and writing code) External assessment . Tools allowed: Part 1: without aids Part 2: all aids allowed - including internet connection Any types of communication between students or between a student and an external party is prohibited and will be considered a violation of the exam rules . Re-exam: Please note that re-examinations may take a different form than the ordinary exams .
IT-SEP1	Semester Project: Single User System	10	The student will use the knowledge acquired in SDJ1, WEB1 and DMA1.	Explain the Waterfall method as a software development process Derive requirements Apply use case modelling and draw activity diagrams Draw a domain model Construct UML class diagram(s) Draw a sequence diagram of one essential method Implement a software system using object-oriented programming	Demonstrate the connection between the different disciplines in software development Evaluate the performance of selected parts of the system in terms of time and space complexity using the Big O notation Describe and reflect on the group's cooperation Apply and reflect on covered learning theories and motivation theories	Prerequisites: Project and process report must be handed in before the deadline and must apply with the Formalities Criteria stated under the Software Engineering specific Guidelines. Type of examination: Group exam with individual assessment. Group presentation of 5 minutes times the number of group members followed by a joint examination of 15 minutes times the number of group members including voting. Individual grades are given on the basis of an overall assessment of the submitted work as well as the

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
				Integrate Java-generated files into a webpage using JavaScript Perform testing in relation to the derived requirements Describe your system in a user guide Formulate and enforce a group contract Apply theories on group dynamics, team cooperation and conflict resolution Communicate the results of the project work using academic and technical writing, apply the correct report structure and rules on plagiarism Communicate successfully in writing, graphically and orally to different target groups Identify relevant problems, formulate a statement, and account for different possible solutions Reflect on the waterfall method as a tool to control a software development process		individual´s performance during the examination. Internal assessment. Allowed tools: All Re-exams: Examination will take place as the ordinary exam. Students who fail a semester project must attend an information meeting at the end of the summer exam period. At this meeting, students will be notified about the process of the re-exam and students will form groups. Furthermore, specific deadlines and exam dates are announced at this meeting. The project must be handed in before the stipulated deadline. An exam schedule will be posted after the hand in deadline.
IT-DBS1	Database Systems	5	Having completed this course, students will be able to: • account for the relationship between relational algebra and SQL • define the relational model • distinguish the 3 normal forms • classify keys in relational databases • explain indexes	Having completed this course, students will be able to • use relational modelling to model business cases • use UML to document ER-Models • use Data Definition Language (DDL) to create databases • use Data Modelling Language (DML) to manipulate data in a	Having completed this course, students will be able to: • Design and implement a database schema on the 3rd normal form • Use a database in application development	Exam prerequisites: : • Course assignments handed in before deadline • The student must have an attendance of at least 75% in order to qualify for the exam. Students who do not have at least 75% attendance will automatically fail the ordinary exam . Exam type and Assessment: Duration: 4 hours Digital written exam (2 parts): Part 1: Multiple choice and written exam answers; 2 hours without aids Part 2: Design and implementation; 2 hours with all aids,

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
				database • use the mapping method to convert ER-Models to Relational Models • use normalisation to normalise a database schema to 3rd normal form • create SQL statements to create, replace, update and delete data in a database • use keys in relational databases • use joins to combine data • use transactions to prevent data corruption • create triggers • create views		including internet connection External assessment Allowed tools: Textbook, written exam or printed notes, and files saved to personal computer . Re-exam: Conducted as the ordinary exam.
IT-SDJ2	Software Development with UML and Java 2	10	The student should be able to understand: • System architecture • Various methods for testing • Concurrent programming • Design patterns • Client/server structure	The student should achieve the skills: • Implement design patterns in Java • Test software using different testing techniques, including (but not limited to) JUnit testing, System testing, etc . • Implement thread-safe classes and multi-threaded programs • Make programs communicate using client-server technologies	The student should be able to: • Implement programs in Java using design patterns, and evaluate which to use • Test software using relevant testing techniques • Develop flexible java code using interfaces • Implement thread-safe classes and multi-threaded programs • Implement client-server systems	Exam prerequisites: : • Course assignments handed in before deadline • The student must have an attendance of at least 75% in order to qualify for the exam. Students who do not have at least 75% attendance will automatically fail the ordinary exam . Exam type: • Individual oral examination without preparation based upon course work . • The student will draw from a pool of previously known questions . • The student will explain concepts and theories from the course, using the course work as reference . • The student will start with a prepared presentation . • External assessment . Allowed tools: NA Re-exam: Conducted as the ordinary exam.
IT-SEP2	Semester Project: Client/Server	10	Apply all knowledge from all 2. semester courses	Document the analysis and design of a software system using UML.	Carry out a software project with a clear connection between requirements,	Criteria to register for the exam: Project report and process report must be submitted before the deadline and in accordance with formality

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
	System			Devise requirements, use case model, activity diagrams, system sequence diagrams, domain model. Design and implement (using SQL) a normalized relational database for data-persistence. Apply unit and use case testing based on requirements and code. Implement a client/server software system in Java according to design. Apply covered theories on personal profiles Manage cross-cultural aspects in the group work Describe potential conflicts in the group work and suggest solutions. Carry out a software project in close cooperation with the group. Use correct academic and technical writing style, report structure, and rules for plagiarism. Work based on the project group's own problem statement. Apply Scrum in combination with Unified Process (UP) and relevant tools in the project work.	analysis, design, implementation, test, and documentation. Produce a software design which supports group collaboration. Apply relevant design patterns and principles, including SOLID, resulting in a maintainable and extendable software system. Perform and explain technical choices. Enforce and develop the group contract. Define and reflect on own learning goals from the previous and the current semester. Communicate the project work's results and the project group's learning process in a structured manner using technical terms both in writing, graphically and orally.	requirements. Students must have minimum 75% attendance to be registered for the exam. Exam type and Assessment: Group exam with individual assessment. Group presentation approx. 20 minutes followed by a joint examination with a joint discussion and individual question rounds for approx. 25 minutes per student including voting. Individual grades are given on the basis of an overall assessment of the submitted work as well as the individual´s performance during the examination. Internal examiner. Allowed tools: All Re-exams: Students who failed a semester must make a new project. Students who have failed a semester project in January or June must attend an information meeting on the last Friday in June. At this meeting, the students will get information on specific deadlines as well as the process of re-exam. They will form new groups, if possible in relation to the number of failed students at the individual semesters. Deadline for hand in of the project is mid-August (exact date will be informed at the meeting). There will be no guidance in the period up to hand in. Oral assessment of the project takes place before the start of the autumn semester.
IT-SWE1	Software Engineering	5	The student should be able to account for: • Abstraction • UML (selected diagrams)	The student should achieve the skills to: • Analyse a problem and document the analysis- and	The student should be able to: • Analyse a problem– what is the problem to be solve?	Exam prerequisites: : • Course assignments handed in before deadline • The student must have an attendance of at least 75 % in order to qualify for the exam. Students who do not

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			• S .O .L .I .D principles • Unified Process • Scrum • Design principles • Architectural design • Requirement capturing • Analysis vs. Design models • The difference between software development and coding • Test descriptions • How to conduct a test following a test description	design-process with text and UML • Apply use of Scrum • Apply use of Unified Process • Use UML to document requirements, analysis, and design artefacts • Use agile software development with Unified Process in combination with Scrum • Create a Domain model from a problem description, requirement specification and understand the elements in the resulting Domain model • Create a design model and understand the elements within it • Apply the S .O .L .I .D principles on a design model • Design for test • Create test descriptions • Create architectural design models	• Derive a requirement specification with Use Cases and non-functional requirements • Plan tests by Test specifications • Analyse and design a project to be implemented in teams with many participants and stakeholders • Work in a Scrum team	have at least 75 % attendance will automatically fail the ordinary exam. Exam type: Internal assessment . IF you are full degree Software Engineering student: • SWE 1 is evaluated together with SEP 2 project . • The SEP 2 project and the exam must demonstrate understanding of SWE 1 skills and competencies and their use in practice . During the SEP 2 exam, specific SWE 1 questions will be asked ELSE Individual oral examination without preparation, where 50% of the grade is based on examination of one or more course assignments and 50% of the grade is based on a drawn question . ENDIF Allowed tools: All Re-exams: Individual 15 minutes oral examination without preparation, where 50% of the grade is based on examination of one or more course assignments and 50% of the grade is based on a drawn question. The questions will be known before the examination .
IT-CAO1	Computer Architecture and Organisation	5	Having completed this course, the student is able to: Describe and apply numbering representations, including two's complement to represent negative numbers in the binary numbering representation Identify the functionality of basic logic gates and be able to combine them into half- and full-adders, flip/flops, etc . Describe Boolean algebra and it's relation to digital	Having completed this course, the student should be able to: Create functioning assembler programs for microcontrollers Analyse ASM programs (AVR MCU) and calculate execution time Execute and debug assembler programs Analyse and describe simple logical circuits (Boolean expressions) Apply Boolean algebra to reduce digital circuits.	Having completed this course, students should be able to: Describe the functionality of the components of basic computer architectures Apply mathematical theory to understand low-level computer architecture and programming Create simple logic circuits used in CPUs Create applications using assembler programming	Exam prerequisites: Two assignments must be approved before attending the exam. The student must have an attendance of at least 75% in order to qualify for the exam. Students who do not have at least 75% attendance will automatically fail the ordinary exam. Exam type and Assessment: Written exam, 2 hours. External assessment The exam is digital and it is the responsibility of the student to bring a computer that works together with WISEflow and Flowlock.

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			circuits Describe the architecture of simple CPUs and how they function, explain the build and working behaviour of basic building blocks of CPUs (registers, ALUs, etc .)		Integrate simple I/O devices in embedded applications.	Allowed tools: Course literature and personal notes. Re-exams: Conducted as the ordinary exam
IT-DNP1	.NET Programming	5	The student will be able to: Describe the fundamentals of .NET development and the common type system Identify and describe .NET technologies relevant to web application development	The student will be able to: Write and debug C# code Implement RESTful Web Services in relation to a distributed system Consume RESTful Web Services Utilize asynchronous programming Create and interact with a relational database using an Object Relational Mapping library Define and implement basic authentication and authorization Navigate and use the managed .NET API Create and consume class libraries Compare object-relational mapping to traditional data access techniques Implement a Web App within ASP.NET	The student will be able to: Implement a robust, error-safe system Implement console applications, web applications and web services as part of a distributed system with Server-side and client-side C#-programming Data persistence using object-relational mapping User management, including authentication and authorization Analyse and evaluate the relevance of .NET technologies when designing software applications Apply best practices when developing .NET apps	Exam prerequisites: Course assignments handed in before deadline. The student must have an attendance of at least 75% in order to qualify for the exam Exam type: Individual written examination Duration: 3 hours Programming exercises Internal examiner Tools allowed: All aids, including internet connection Re-exam: The re-examination may take a different form than the ordinary exam
IT-NES1	Networking and Security	5	Having completed this course, students will be able to: - Account for layered abstractions in protocol stacks - Can explain the Internet's naming system	Having completed this course, students will be able to - Compare and contrast different encryption technologies - Discuss how confidentiality, integrity and availability can	Having completed this course, students will be able to - Analyse network traffic using packet sniffer software - Create application layer protocols for distributed systems	Prerequisites for exam: The student must have an attendance of at least 75% to qualify for the exam. Students who do not have at least 75% attendance will automatically fail the ordinary exam. Exam type: Written examination, 3 hours. Internal assessment

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			- Explain addressing in the Internet - Identify common Internet threats - Describe common access control systems e.g., packet filter, proxy, etc. - Describe privacy, integrity, and authentication methods - Identify common causes for network delays - Explain principles of VPN	be accomplished using security technology - Calculate and measure delays in a network	- Identify security threats and propose mechanisms to mitigate these threats	Tools allowed: Course literature according to the course description Personal notes on paper Access to local pdf-files Access online to: https://www.wolframalpha.com/ Laptop (no access to general internet) Re-exam: Conducted as the ordinary exam
IT-SDJ3	Software Development with UML and Java 3	5	The students will be able to - describe various distributed system types - explain various distributed system architectures - explain various distributed communication methods - explain the use of contracts in service calls - explain peer-to-peer systems	The students will be able to - use various distributed communication methods for direct and indirect communication - argue the choice of middleware for a given distributed system - compare peer-to-peer systems with client/server systems	The students will be able to - design the architecture of a distributed system using known architectural patterns - design and implement a distributed system on different platforms using various middleware	Criteria to qualify for the exam: - Course assignments handed in before deadline and approved - The student must have an attendance of at least 75% in order to qualify for the exam. Exam type: Individual oral examination without preparation covering mandatory course work and theory covered in the course. Duration: 20 minutes. External assessment Allowed tools: All Re-exam: Conducted as the ordinary exam.
IT-SEP3	Semester Project: Heterogeneous System	10	Can refer to involved theories in order to increase efficiency for the group as a whole but also for the individual student.	In addition to the skills acquired in IT-SEP2, the student will be able to: - Implement heterogeneous systems using multiple network protocols - Analyse the security risks of a distributed system - Use a version control system to manage versions - Search, find and include relevant knowledge - Argue for the choice of sources and references in connection with the project work.	In addition to the competences acquired in IT-SEP2, the student will be able to: - Analyse, design, implement and test a distributed system using UML, Java and C# - Construct a distributed system with the proper selection of architectural patterns - Argue for the choice of various technical solutions for implementing distributed systems - Use a defined methodology	Criteria to register for the exam: Project report and process report must be submitted before the deadline and in accordance with formality requirements. Students must have minimum 75% attendance to be registered for the exam. Type of exam: Group examination with individual assessment. Group presentation approx. 20 minutes followed by joint examination with joint discussion and individual question and answer sessions for approx. 20 minutes per student including evaluation. Individual grades are given on the basis of an overall assessment of the submitted work as well as the individual's performance during the exam. Internal assessment.

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
				- Work with a holistic view of the project, the subjects and the outside world.	to structure the development process - Take responsibility for structuring and adapting the form of collaboration to the members' personal and interpersonal competencies - Take responsibility, in collaboration with the group, for the work process in connection with report writing and presentation. - Work analytically, methodically and structured with the semester project in the project group. - Can plan, adapt and optimize a project process with reasoned selection of the specific project management tools.	Tools allowed: All Re-exam: Students who failed a semester must make a new project. Students who have failed a semester project in January or June must attend an information meeting on the last Friday in June. At this meeting, the students will get information on specific deadlines as well as the process of re-exam. They will form new groups, if possible in relation to the number of failed students at the individual semesters. Deadline for hand in of the project is mid-August (exact date will be informed at the meeting). There will be no guidance in the period up to hand in. Oral assessment of the project takes place before the start of the autumn semester.
IT-ADS1	Algorithms and Data Structures	5		The purpose is to qualify the student to design, implement and analyze different algorithms and to become acquainted with different advanced data structures		Permit criteria for attending examination: Mandatory course assignment handed in before deadline and accepted At least 75 % attendance Exam type and Assessment: Written exam (3 hours) Internal assessment
IT-DOC1	DevOps and Cloud	5		The purpose of the course is to introduce students to a set of practices that help deliver software in a fast and reliable manner. By automating workflows and integrating the processes of software development and IT operations, the students will be able to create a consistent toolchain that supports planning, building and operating a software project.		Permit criteria for attending examination: At least 75 % attendance Mandatory course activities completed Exam type and Assessment: Written examination (1 hour) Allowed tools: None Internal examiner

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
IT-ESW1	Embedded Software	5	The student should be able to account for: Analysis of an embedded problem and documentation of the analyse- and design- process in UML Basic concepts of programming with a Real-time Operating System (RTOS)Issues like deadlocks etc .Real-time C-programs for embedded Micro Controller Units Dynamic memory management in C Unit test of C-programs	The student should be able to exemplify: Analyse and design of an embedded software solution Implement functioning real-time programs in C using FreeRTOS Implement programs in C using different C API's and libraries for hardware drivers etc .Document C source code with Doxygen	The student should be able to: Design and construct real-time systems using FreeRTOS and C-programming Construct real-time programs Apply FreeRTOS timers in real-time programs in C Apply synchronization and avoid dead-locks Apply memory management, resource sharing and control	Permit criteria's for attending examination: Mandatory course activities completed At least 75% attendance Exam type and Assessment: Three hour written exam. Allowed tools: Laptop Internal examiner.
IT-WEB2	WEB Development 2	5		The purpose of the course is to introduce the students to modern front-end web development, by going in depth with the JavaScript language & front-end JavaScript frameworks.		Permit criteria for attending examination: Mandatory course assignment handed in before deadline and accepted At least 75 % attendance Exam type and Assessment: Written exam (3 hours) External assessment
IT-SEP4	Semester Project 4	5		Conceive, design and implement a software solution including hardware sensors, a web-based user interface and a persistent multi user backend infrastructure. The integration and delivery of the software will be managed through the implementation of a deployment pipeline.		Permit criteria for attending examination: Project and process report handed in before deadline Type of examination: Group exam with individual assessment. Group presentation (~30 minutes) followed by a joint examination of 3-4 students from the group at a time (~15 minutes per student). Individual grades are given based on an overall assessment of the submitted work (50%) as well as the individual´s performance during the examination (50%). Internal examiner Allowed tools: All Re-examination Students who failed a semester project in January or June must attend an information meeting on the last Friday in June .

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
						At this meeting, the students will get information on specific deadlines as well as the process of re-exam . They will form new groups, if possible in relation to the number of failed students at the individual semesters . Based on the feedback, the students have received after the ordinary exam, they must prepare a new project, or the failed project must be improved . Deadline for hand in of the project is mid-August (exact date will be informed at the meeting). There will be no guidance in the period up to hand in . Oral exam assessment of the project takes place in late August.
IT-INP1	Engineering Internship (IT-)	30	The student must: • gain knowledge of theory, methodology and practice within a profession or one or more fields of study • be able to understand and reflect on theories, methodology and practice • be aware of non-technical – societal, health and safety, environmental, economic and industrial – implications of engineering practice .	The student must: • be able to apply the methodologies and tools of one or more fields of study and to apply skills related to work within the field/fields of study or profession • be able to assess theoretical and practical problems and to substantiate and select relevant solutions • be able to communicate professional issues .	The student must: • be able to handle complex and development oriented situations in study or work contexts • be able to independently participate in professional and interdisciplinary collaboration with a professional approach • be able to identify own learning needs and to organise own learning in different learning environments • promote an engineering-oriented approach during the remaining semesters on the Bachelor programme • develop personal skills required for the professional career as engineer • form the basis for developing personal/professional network	Exam type and Assessment: In order to get an internship evaluated, the student must fulfill the following requirements concerning mandatory assignments: • Expected outcome/specific learnings targets for the internship position • Company presentation • Logbook • Main academic assignment(s) • Final reflections • Participation in workshop for coming interns
IT-BPR1	Bachelor Project Preparation	5	After having completed this course, the student should be able to: Explain the concept of plagiarism and how to avoid it Identify a problem and a problem domain	After having completed this course, the student should be able to: Perform information search and retrieval Apply selected theories on their own group and identify potential	After having completed this course, the student should be able to: Describe a larger Software Engineering project in a Project Description Apply the preliminary steps in a	Exam type and Assessment: The basis of the evaluation is three mandatory assignments: Group Description and Group Contract Project Description Requirements/User Stories

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			Evaluate teamwork and team dynamics Select relevant methods for developing a project	challenges Describe a proposed problem to solve, as well as its context Plan and delimit a software development project Find relevant guidelines and templates	system development process Define clear and concise requirements using a selected standard Demonstrate the ability to work coherently in a group	
IT-BPR1	Bachelor Project Preparation (from A22)	5	After having completed this course, the student should be able to: Explain the concept of plagiarism and how to avoid it Identify a problem and a problem domain Evaluate teamwork and team dynamics Select relevant methods for developing a project	After having completed this course, the student should be able to: Perform information search and retrieval Apply selected theories on their own group and identify potential challenges Describe a proposed problem to solve, as well as its context Plan and delimit a software development project Find relevant guidelines and templates	After having completed this course, the student should be able to: Describe a larger Software Engineering project in a Project Description Apply the preliminary steps in a system development process Define clear and concise requirements using a selected standard Demonstrate the ability to work coherently in a group	Permit criteria for attending exam: Mandatory course activities completed Type of Exam: Assessed on the basis of one mandatory assignment, the Project Description. Internal assessment Re-exam: Re-submission of a revised Project Description
IT-BPR2	Bachelor Project 2	15	After having completed this course, the students must master the knowledge about: Searching and scoping relevant project information Project and team work planning Communication and documentation skills Testing	After having completed this course, the student must master to: Identify and justify problems and their context Select and argue for choice of method and reflect critical and said methods Find and assess relevant literature within the problem domain Present the result for an audience of engineers	After having completed this course, the students must be able to: Describe and delimit a large Software Engineering Project Select and use relevant theories and methods to solve the problem Plan and structure the project within the BPR2 time frame Initiate the preliminary steps in a system development process, leading to a clearly defined requirements capture, use cases as well as object and behaviour analysis .Work successfully in a project group with the objective of solving a well-defined engineering problem .	External examination . The basis of the evaluation is the reports, the solution of the Software Engineering problem, and the oral examination. The student’s ability to express oneself (in writing and oral examly) and to spell is part of the evaluation . Exam type and Assessment: Oral exam examination . Group presentation of the project (20 minutes). Individual examination of each member of the group (20 minutes) . The individual examination typically starts from topics in the report and may involve all the topics from 1st to 7th semester .
IT-	Semester	10		The purpose of SEP6 is to		Permit criteria for attending exam:

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
SEP6	Project 6 - Software Eng.			develop and document a cross-organisational innovation project in collaboration with a company or institution.		Mandatory course activities completed Exam type and Assessment: Oral exam External assessment
IT-ALI1	Applied Linear Algebra	5	After successfully completing the course, the student will have gained knowledge about: What a vector space is, and How a linear representation of such spaces can be analysed using matrix operations Application of linear algebra in engineering	After successfully completing the course, the student will be able to: Apply techniques and results from linear algebra to solve problems in linear systems, matrices, vector space, orthogonality, eigen vectors, and eigenvalue Apply theory to analyse basic theoretic tasks within the below mentioned topics Express mathematically correct arguments Use mathematical terminology and symbol language	After successfully completing the course, the student will have acquired competences in: Applying linear algebra to the study of various phenomena in engineering science Using matrices to solve concrete problems Using vector operations to solve concrete problems Applying methods and results from linear algebra in the solution of engineering problems	Grading will be done according to the 7-scale, using an internal examiner. Exam type and Assessment: The final exam is a 3 hour written exam and takes place at Campus Horsens. All supplementary materials and aids are allowed, e.g. using a computer as a reference work. Communication of any sort is not allowed during the exam and will lead to expulsion of all involved parties from the exam. Internet access is not allowed . The exam must be completed in the Jupyter Notebook environment and the answers must be submitted in WISEflow . The re-exam may be held as an oral examination .
IT-BEL1	Basic Electronics	5	Having completed this course, students should have understanding of: Statistics, Observation variance and error Ohm’s law and Kirchhoff’ law used on small electronic circuits Theory of basic analogue electronic components (resistor, capacitor)Operation Amplifiers, Instrument Amplifier, Diode, transistor (NPN, PNP)Cooling of electronic components (Heatsink, Compound)Simulation and practical build in Lab of small electronic circuits Low pass, Band pass, High pass, Butterworth filter properties	Having completed this course, students should be able to: Simulate analogue electronic circuits using simulation software Construct active filters with desired property for specific application Construct electric car model on breadboard Build and test prototype circuits Perform measurements on electronic circuits, using Digital Multi Meter and Pico-scope	Having completed this course, students should be able to: Design simple electronic circuits for measurement systems using amplifiers and filters Analyse experiment results, using statistical calculations and methods Write reports to document engineering experiments	Permit criteria’s for attending examination: * Mandatory course activities completed . Exam type and Assessment: Ongoing tests in the form of three written assignments that weigh equally and make up 50% of the assessment Oral exam (20 minutes), weighing 50% Internal assessment Allowed tools: All

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			Strain Gauges used in Wheatstone Bridge			
IT-BUI1	Business Intelligence	5	Students will obtain knowledge about understanding, reading, and presenting data from a dimensional model (such as a star schema or data cube) and other data models. Knowledge about building data products for operational vs real-time systems	Data migration using data integration tools Create Data pipelines to cleanse data and move it into a data warehouse Create KPIs and measures Create data analyses, presentations and dashboards with Business Intelligence tools Create data structures for analysis purposes with selected tools Create, deploy and manage reports	Evaluate pros/cons of different BI products, architectures and approaches	Internal examination. In order for the student to qualify for the examination, the course assignment must have been handed in and approved . Exam type and Assessment: Oral examination based on the course assignment . Approximately 20 minutes incl. discussion of examinee's performance, without preparation .
IT-CMC1	Compiler Construction	5	The students will be able to describe the main purposes of a compiler explain the differences between syntax and semantics of a programming language explain context free grammars list examples of common programming language features	The students will be able to construct a context free grammar for a programming language define the semantics of a programming language in an informal way design the runtime organisation for a programming language	The students will be able to design a small, simple programming language design and implement a compiler for a small, simple programming language using various design patterns and an object oriented language for implementation	Permit criteria for attending examination: • Hand in of course project Exam type and Assessment: The exam is an individual 20 minutes oral exam. The student draws a question from the theory and answers it based on the produced course project . Allowed tools: All Internal exam .
IT-DAI1	Data Analytics Infrastructure	5	Having completed this course, students should be able to describe basic techniques within the field, and argue the choice and applicability of these for different use scenarios . This includes: • Application of analytical data processing, and differences to transactional processing • Types of analytical data processing, such as reporting	Having completed this course, students should be able to: • Design and implement data models for integrating multi-source data, including dimensional data modelling, for structured and semi structured data • Design and implement data models for time-variant data • Design, implement and test systems for data acquisition, validation, integration and	Having completed this course, students should be able to • Discuss and argue pros, cons and trade-offs of choices • Use basic statistics and visualization to find and explain patterns of information in data • Evaluate and act upon peer feedback	Permit criteria for attending examination • Mandatory course activities completed • Course assignment handed in before deadline Type of exam Individual oral examination without preparation based upon course assignment(s), covering mandatory course work and theory covered in the course. Duration (grading included) app. 20 min/ 5 ECTS . Allowed Tools: All Internal Examiner

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			and visualization • Sources of data for analytical processing • Server and locally hosted platforms for data storage and analytical processing • Modelling techniques for designing data models for integration of multi-source data, including structured, semi-structured and unstructured data, and for modelling time-variant data/history • Design of systems for data acquisition, validating and cleansing data, integration and publishing of data .	delivery from multiple sources and platforms • Design, implement and test basic descriptive statistical analysis on integrated data • Design, implement and test basic visualizations and graphs of data and analysis results . • Give relevant peer feedback on handins and exercises throughout the semester		
IT-DIM1	Digital Multi Media	5			Having completed this course, students should have profound knowledge of: • Computer Graphics • Design Principles for multimedia • Video, Animation and Sound • XML and Multimedia	Evaluation is based on a written exam group course assignment, where it must be clearly marked which sections of the course assignment each group member contributed with. Furthermore, each group member must also hand in an additional 1-2 pages of individual reflections on the work they have done in the course assignment.
IT-DSP1	Digital Signal Processing	5	After successfully completing the course, the student will have gained knowledge about: The nature and recording of different types of digital signals Cleaning up digital signals Extracting useful values from digital signals MATLAB as a tool for development of signal processing algorithms	After successfully completing the course, the student will be able to: Record digital signals Apply different filters (high-pass, low-pass, band-pass, notch) to remove unwanted components of digital signals Use the Fast Fourier Transform to analyse the frequency content of a signal	After successfully completing the course, the student will have acquired competences in: Explain sampling processes and how to determine the correct sampling frequency Describe signal processing applications Applying digital signal processing methods to analyse and interpret engineering problems Develop signal processing algorithms	The course is evaluated via an oral exam after course completion .Grading will be done according to the 7-scale, using an internal examiner . Exam type and Assessment: At the end of the semester, the students will hand-in an assignment and the final exam will be based on this assignment . The students will present the assignment in the form of a demonstration, followed by questions about the signal processing and feature extraction methods as well as the MATLAB programming .
IT-EOS1	Embedded Operating	5	Having completed this course, students should be	Having completed this course, students should be	Having completed this course, students should be	Permit criteria for attending examination Mandatory assignments handed in before deadline and accepted .

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
	Systems		able to Account for advantages and disadvantages of Linux as operating system in embedded systems .Describe the anatomy of a 32-bit embedded system .Describe the features of a Beagle Bone system .Describe the boot process of a 32-bit ARM based Linux system .Explain Pulse Width Modulation .Explain I2C communication bus technology .Explain the structure of Linux file system and access permissions .Explain how to connect and read input from sensors in an embedded Linux environment .Explain how to connect and control actuators in an embedded Linux environment .	able to Use basic Linux commands and utilities .Select, install, configure and use tools needed for developing embedded systems .Execute a firmware upgrade on a Beagle Bone system .Install and configure “off the shelf” software in Linux .Use the GPIO structure in Linux to interface sensors and actuators .Use Pulse Width Modulation for Control of servo motors, and LED light intensity .Implement BASH scripts to control simple GPIO devices .Implement simple hardware circuits for measurement and control .Use appropriate programming language to implement web based user interface .	able to Implement shell scripts in BASH Design and implement IoT-devices, based on a 32-bit MCU platform with Linux	Exam type and Assessment: Oral examination Individual oral examination based upon a subject found by draw . No preparation . Allowed tools: Laptop Internal examiner .
IT-ERP1	ERP systems SAP ABAP/4 Programming	5	Having completed this course, students will be able to: Understand the ABAP Workbench. Create basic ABAP Programs. Understand the control flow and structures in ABAP	Having completed this course, students will be able to: Create Database with domains, data elements and tables Retrieve Data from the Database with open sql. Develop a simple ABAP Programs with modularization. Develop a DYNPRO with navigation (CRUD-functionality) Develop reports with selection screen, alv-list, etc.	Having completed this course, students will be able to: Use the fundamental concepts of the ABAP programming Language Create simple application programs with user dialogs and database connections. Trace the flow of a program and troubleshoot simple problems. Describe change management for new systems. Use ABAP Workbench and basic ABAP language elements.	The exam is oral and it takes 20 minutes per student. The exam is in two parts. First part is a presentation and discussion of selected parts of the course work. Second part is drawn question from the theory of the course. The evaluation of the course is based on mandatory course work (50%) and the oral exam (50%) at the end of the course. Only students with approved course work will be allowed to attend the exam. Internal examination

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
					Explain the relationship and difference between the classical procedural programming model and the object-oriented programming model in ABAP/4. Apply screen flow logic and working with external data. Apply different tools and techniques available to implement dataflow in an ABAP program with database. Design and implement an object-oriented SAP application with a database and ALV Grid.	
IT-GMD1	Game Development	5	After successfully completing the course, the student will have gained knowledge about: General game development principles, tools, patterns and best practices Game engines and real-time development platforms Unity, a cross-platform game engine Code structure in a framework based on a loose implementation of the ECS pattern Basic game design principles The game development production pipeline An engineer's role in game development Intermediate programming concepts such as coroutines, delegates and events The animator state machine Digital audio Various assets in game development Interactive 3D rendering The	After successfully completing the course, the student will have acquired skills in: Navigating Unity and making use of its various features Creating C# scripts to modify game behaviour Structuring game development projects Deploying applications on various hardware Navigating the Unity and C# documentation Utilizing game design theory to conceptualize games Importing and working with various assets from other game development professions including 3D models and animations Working with materials, shaders and textures Handling physics in games Creating and manipulating animations for characters	After successfully completing the course, the student will have acquired competences in developing industry standard interactive experiences using Unity. The student will be able to possess the developer position within a multidisciplinary game development pipeline, identifying and executing on the technical requirements of the developed product. The student will also have a solid foundation to further professional skills in game development independently .	Exam type and Assessment: Evaluation is based on a written exam group course assignment, where it must be clearly marked which sections of the course assignment each group member contributed with . Furthermore, each group member must also hand in an additional 1-2 pages of individual reflections on the work they have done in the course assignment .

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			.NET framework	using state-based machines Creating responsive user interfaces for games Working with digital audio in real-time engines Optimization utilizing the profiler of Unity Utilizing the SOLID design principles in a script-based environment		
IT-HWP1	Hardware Oriented programming	5	Having completed this course, students should be able to Seek information in datasheets for electronic components Describe the difference between polling and interrupt-based drivers Describe layered software design and Hardware Abstraction Layer Explain the Interrupt system in a microcontroller Explain the concept of Pulse Width Modulation Explain Timer/Counters and give examples of their use Explain how analogue signals are sampled and quantified .	Having completed this course, students should be able to Implement low-level drivers for digital I/O-Ports Implement low-level drivers for analogue sensors Implement low-level drivers for analogue actuators .	Having completed this course, students should be able to Design a Hardware abstraction Layer Implement low-level drivers for 8-bit microcontrollers .	Permit criteria for attending examination: Mandatory assignments handed in before deadline and accepted . Exam type and Assessment: Oral examination Individual oral examination (20 min. in total) based upon a subject found by draw .No preparation . Allowed tools: Laptop Internal examiner.
IT-IDX1	Interaction Design	5			Gain skills within interaction design and usability evaluation. You will achieve: knowledge of and experience in User eXperience Design (UX)including knowledge of and experience in participatory design workshops knowledge on planning, preparation, implementation, analysis, and documentation of user-based	The evaluation of the course is based on mandatory course work (50%) and the oral exam (50%) at the end of the course. Only students with approved course work will be allowed to attend the exam. The exam is oral and it takes 20 minutes per student. The exam is in two parts. First part is a presentation and discussion of selected parts of the course work. Second part is drawn question from the theory of the course. Internal examination .

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
					usability evaluation understanding of and practical experience with the interplay between usability evaluation and interaction design in an iterative design process	
IT-LWA1	Internet of Things WAN’s	5	After having completed this course, the students should have knowledge about: IoT protocols LoRaWAN network elements LoRaWAN technology Lora device classes Sensors/Actuators OTA (OverTheAir) activation Personal activation Radio Propagation/Antenna Security IoT Testing	After having completed this course, the student should be able to: Analyse and explain IoT LoRaWAN problems and their context Select and argue for choice of methods and reflect critically on said methods Find and assess relevant literature within the problem domain Plan and present the result for an audience of engineers	After having completed this course, the student should be able to: Assess and delimit a IoT project Plan and structure the project within the set time limit Predict the preliminary steps in a systems development process, leading to a clearly defined requirements capture, use cases, as well as object and behaviour analysis Work successfully in a project group with the objective of solving a well-defined engineering problem	The basis of the theory, simulation and evaluation of the 3 mandatory course assignments report and presentation. Those project assignments will be part of the examination questions . Exam type and Assessment: Oral examination based on one mandatory assignment .
IT-MAL1	Introduction to Machine Learning	5	After having successfully completed the course, the student will have gained knowledge about algorithms, methods, techniques, tools, and applications within the following fundamental machine learning methods: - predictive methods, e.g. regression and classification - descriptive methods, e.g. clustering and PCA - deep learning methods, e.g. neural networks . - clustering methods, e.g. partitional and hierarchal clustering The students must be able to	After having successfully completed the course, the students should be able to apply the algorithms, methods and models from the above-mentioned areas to identify, analyse, evaluate and make suggestions for solving specific data-based issues. They must be able to argue for the relevance of the chosen algorithms as well as for the proposed solution. In addition, they must be able to reflect on the importance of the context in which the solution is included. Specifically, it is expected	After completion of the course, the goal is that the students have acquired the competences to: - Make informed choices about the use of machine learning techniques - Parametrisise machine learning algorithms for a given data material - Design and develop a complete solution for a complex, realistic problem - Communicate and discuss the solutions with professionals and non-specialists .	The students must participate in a group assignment which will also constitute the foundation for the exam. While the assignment is not directly part of the final grade, it will have an indirect influence since the assignment will heavily affect the exam. Each student must explicitly state which sections of the assignment that the student is responsible for, e.g. in the table of contents. If a student does not participate in the assignment, the student will not be able to attend the exam. Exam type and Assessment: The course is evaluated based on a 25 minutes oral examination. The examination will depart from the parts of the group assignment that the student is responsible for, and will evolve into a discussion of the syllabus in general. The oral examination does not include a presentation,

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			relate critically and reflectively to the above topics; in particular, it is important that they become proficient in selecting the right type of machine learning method for use in a given context .	that after completion of the course the students will be able to: - Understand and apply a number of machine learning algorithms to both unstructured and structured data examples - Understand and compare the algorithms behind different data mining and machine learning methods - Match and possibly combine methods for practical use in an appropriate context .		meaning that the student does not prepare or present a presentation. The student is given one grade based on the examination.
IT-NSQ1	No-SQL versus relational databases	5	The student should be able to describe document-based and graph databases explain updating and querying in different database paradigms explain schemas and constraints in non-relational databases compare relational and different non-relational approaches to database design .	The student should be able to apply modelling techniques in document-based and graph databases schemas and constraints to enforce designs in a no-SQL database APIs and languages to maintain and query databases administrative tools to set up data replication administrative tools to set up sharding .	At the end of the course, the students should be able to make an informed choice of database management system design and create a data model in the chosen database system set up a run-time environment to use for the data model .	Oral examination based on a question from the course syllabus based on the course assignment. Approximately 20 minutes (including discussion of examinee's performance) without preparation . Internal exam
IT-PCL1	Programming Concepts and Languages	5	Having completed this course, the student should be able to: describe the key concepts and have a basic understanding of different programming paradigms and languages. understand and use the functional paradigm .develop small and medium size programs using F# and Python programming languages .			The final examination is a three-hour written examination . Internal examination .
IT-PME1	Process Management for ICT Engineering	5	After successfully completing the course, the students will have gained knowledge about: How to ensure quality	After successfully completing the course, the student will be able to: Apply techniques and results from Capability	To complete this course the students must make hand-in: Requirement Specification - IEEE 830 standard"	The course is assessed on the basis of 3 individual assignments, weighing 25%, 30% and 45%, respectively .

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			in projects How to improve your project performance How to handle change management in a project .	Maturity Model Integration (CMMI) to solve challenges in project processes Apply techniques and results from Lewin model to handle change management in project Apply "How to break software" to prevent making mistakes in your project Be able to describe and make use of testing concepts Use of terminology to kick-start Bachelor project .	document for a project Test Specification - IEEE 829 standard" document for a project relations to CMMI model" document for a project .	If the course is failed, the student must go for an internal oral exam re-examination .
IT-RTP1	Real-Time Programming, Interfacing and Electronics	5	Having completed this course, the student has gained knowledge in the below areas. Specifically, the student is able to: Understand the basic concepts of real-time programming Explain issues like deadlocks, priority inversion etc .	Having completed this course, the student should be able to: Write functioning real-time programs in C using FreeRTOS Analyse a simple real-time design for schedulability, deadlocks, utilization etc.	Having completed this course, students should be able to: master and use simple real-time operating systems be able to analyse/design/describe and construct real-time programs understand timers and clocks, and how they are used in real-time programming understand synchronization avoiding dead-locks and priority inversion understand memory management, resource sharing and control be able to design and construct real-time systems using FreeRTOS and C-programming understand low-level protocols, CRC etc .	Permit criteria for attending examination: Mandatory assignment handed in before deadline and accepted . Exam type and Assessment: Oral examination based on mandatory assignment .
IT-SCP1	IT Security and Cryptography in Practice	5	After successfully completing the course, the student will have gained an understanding of the cross-disciplinary nature of cyber security, and the complexities, challenges and wider implications of the	After successfully completing the course, the student will be able to Draw on and apply relevant IT security approaches, tools and frameworks for IT security enquiry to different settings in real world situations .Frame	After successfully completing the course, the student will have acquired competences in Applying complex cryptographic primitives to real-world cases Document and explain an IT-security project clearly and	The student must attend three mandatory seminars: i) introduction, ii) midway, iii) final. The student must hand in two compulsory papers: i) midway paper, ii) final paper . Students are assessed by the lecturer based on 1) Final paper (50%) 2) Participation and performance at final seminar (20%)

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			contexts in which cyber security problems occur in the workplace .knowledge about several key implementations of cryptography and other IT-security related issues .	and address IT security problems, questions and issues as a IT security project, being aware of the environment and context in which the problem exists .	unambiguously to peers Review, evaluate and reflect upon knowledge, skills and practices in cyber security .	3) Midway paper (10%) 4) Participation and performance at midway seminar (20%) If a student fails to meet one or more of the above requirements for passing the course, the student will be given an extra assignment whose scope depends on the scope of the missing requirements .
IT-SMP1	Stochastic Modelling and Processing	5	After successfully completing the course, the student will have gained knowledge about: The main working tools and concepts of stochastic modelling Probability theory and distributions Confidence Intervals and Hypothesis Testing Inferential statistics	After successfully completing the course, the student will be able to: Apply results from basic probability theory including conditional probability Use probability density and distributions functions of one and two variables Account for random variables and random processes Calculate and estimate errors and uncertainties .	After successfully completing the course, the student will have acquired competencies in: Planning experiments and state hypothesis Presenting statistical results from experiments Modelling experimental data with regression Analysing experimental results and test hypotheses	Grading will be done according to the 7-scale, using an internal examiner . Exam type and Assessment: The final exam is a 3 hour written exam and takes place at Campus Horsens. All supplementary materials and aids are allowed, e.g. using a computer as a reference work. Communication of any sort is not allowed during the exam and will lead to expulsion of all involved parties from the exam . The exam must be completed in the Jupyter Notebook environment and the answers must be submitted in WISEflow . The re-exam may be held as an oral examination .
IT-SWA1	Single-Page Web Applications	5	The student should be able to - describe dynamic and static languages - apply common design patterns to document object models - explain techniques and pitfalls of asynchronous programming - explain the prototype model and contrast it with class-based inheritance - compare single-threaded and multi-threaded asynchronous model - explain the elements of the	The student should be able to apply - Object-oriented programming in JavaScript and TypeScript - Functional programming in TypeScript - Callbacks and higher-order functions in TypeScript - Asynchronous programming using Promises - Manipulating web pages using JavaScript and TypeScript - Calling web services using XmlHttpRequest	At the end of the course, the students should be able to - design and construct single-page application using AJAX techniques, including JavaScript, TypeScript, HTML DOM, XML, JSON and web services.	Oral examination based on a question from the course syllabus based on one of the course assignments. Approximately 20 minutes (including assessment) without preparation. Internal exam

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			TypeScript type system			
IT-XRD1	XR Development	5	After successfully completing the course, the student will have gained knowledge about: - XR, AR, VR, AV & MR terminology - State of the art and the evolution of AR & VR hardware - Use cases for AR and VR applications - Tracking technologies for XR - Display technologies for XR - The Vuforia Engine - ARCore and ARFoundation - XR Interaction Toolkit - The Unity XR tech stack & OpenXR - XR Interaction techniques - Rendering challenges in XR - Spatialized audio - OVR and similar integrations for hand tracking, locomotion and specialized interactions	After successfully completing the course, the student will have acquired the skills to: - Analyse and optimize an AR or VR development workflow - Compare and utilize various SDK offerings and libraries for XR development - Work with and reflect on the theory behind prominent challenges in the XR industry such as tracking, rendering, locomotion and input - Reflect on underlying sensor and display technologies for XR hardware - Classify XR applications and reflect on their use cases - Describe, compare and apply various interaction techniques in XR	After successfully completing the course, the student will have acquired competencies to develop marker based and markerless augmented reality applications and mobile virtual reality applications .	Prerequisites: If the compulsory projects and course activities are not approved by the lecturer, the student will be denied access to the exam and will thus have used one exam attempt . Exam type and Assessment: Written exam assignment spanning the semester . Assessment is based on projects developed in groups where it must be clearly marked which parts of the applications each group member contributed to . Furthermore, each group member must hand in an additional 1-2 pages (one page is considered 2400 characters) of individual reflections on the work they have done . Assessed by the lecturer and an internal assessor . Allowed tools: N/A Re-exams: Students who fail the ordinary exam will be given a new deadline for handing in a revised assignment .
SE-LCA1	Circular Economy and LCA	5	Students completing this course will be familiar with: The international guidelines for LCA analyses (ISO standards 14040 and 14044) .The step-by-step working process that must be followed when carrying out an LCA analysis .The principles behind defining functional units, system boundaries and time scopes for LCA analyses .Chosen data sources providing data for LCI´s and LCIA´s .Different	Students completing this course will be able to: Define functional units, system boundaries and time scopes for LCA analyses according to the guidelines .Carry out LCA analyses for simple production or service system scenarios according to the guidelines .Compare competing production or service systems based on an LCA analysis .Present and interpret results of LCA analyses and discuss these in	Students completing this course will be able to: Define comparable scenarios for competing production/service systems in order to analyse the respective environmental impacts of these Relate results from LCA analyses with the ideas of CE to suggest sustainable choices in given situations Discuss how working towards fulfilling the SDGs requires individual as well as a political change of behaviour Reflection about	Prerequisites: Mandatory course activities completed. Mandatory assignments handed in before deadline and accepted. Type of examination: A case based written exam with internal examiner. Allowed tools: Re-exam: Not passing the course - a new course assignment will be given, to be accepted and evaluated in equal manner as within the course.

Code	Title	ECTS-points	Knowledge	Skills	Competencies	Exam prerequisites and Assessment
			environmental impact categories .The common way to graphically present end results of LCA analyses .How the UN system influences global development within CE .The UN SGDs	relation to decision-making .Search for and identify relevant data for Life Cycle Inventories (LCI) .Prepare simple Life Cycle Inventories (LCI) and carry out Life Cycle Impact Assessments (LCIA) based on these, according to the guidelines .Graphically present the results of LCA analyses and explain how these are related to the former steps of the analyses .Carry out an LCA by using the program “LCABYG” .Identify barriers to change of CE development .Identify opportunities for CE business development. Make a simpel business model .Formulate individual change of behaviour to promote CE .Evaluate business cases in relation to fulfilling the SDG .Promote circular economy as an innovation tool for companies .	business models and product development in CE .	