Master in Applied Computer Science, part-time (120 ECTS)

Study Plan for Master in Applied Computer Science, part-time (120 ECTS) (2021–2025)

Facts about the program

ECTS Credits:
120

Study duration:
4 years

Teaching language:
English

Campus:
Halden

Table of contents

What do you learn?

Degree/title obtained

Students that complete and pass the programme are awarded the degree of Master in Applied Computer Science.

Some students will also have the right to use the protected title sivilingeniør after graduation if they fulfil the following requirements:

- Completed a bachelor's degree in computer engineering based on the national curriculum regulations or an equivalent education that meets the minimum requirements of 25 ECTS mathematics, 7,5 ECTS physics and 5 ECTS statistics. These requirements must be fulfilled no later than during the first year of study on the master's programme.

The additional title sivilingeniør will be stated in the diploma.

Learning outcomes

Knowledge

The candidate has

- deep knowledge about research and development within the field of applied computer science;
- advanced knowledge about literature and methods used within the field of applied computer science.

Skills

The candidate is able to

- work independently with a problem statement over an extended period of time;
- analyze a situation, formulate a problem statement, and develop a plan for solving the problem;
- create models and implement them in a digital environment;
- make realistic and feasible plans by taking into account possibilities, limitations and use of time;
- collect and analyse relevant information with emphasis on sourvce criticism

- present research and results clearly and unambiguous within the field of applied computer science;

- formulate his/her own and other people's reflections and solutions within the field of applied computer science.

General competence

The candidate

- has retained and further developed his/her academic curiosity, knowledge, openness and precision as well as the ability to distinguish between knowledge and opinions;

- is capable of critical reflection on ethical, scientific and philosophical issues within the field of applied computer science;

- has gained knowledge of scientific literature, methods and theories within the field of applied computer science;

- can communicate knowledge clearly in writing as well as orally.

Admission

A bachelor's degree or an equivalent education of at least 180 ECTS credits,
and in addition or included at least 80 ECTS credits in informatics,
and in addition or included at least 20 ECTS credits in programming

To be admitted, the applicant must have an average grade value from the qualifying education of at least 25 (according to ECTS standards).
Proof of English proficiency is also required.

In addition, all applicants must pass both a written essay and an online entrance test.

Applicants from countries outside the EU/EEA must submit proof of funding when submitting their application.

Structure and content

The structure and content of the programme

The master programme in applied computer science is a natural extension of the faculty's bachelor studies and builds on the research activities of the faculty’s staff.

In the first semester, students select courses from two of the following four research areas:

- Interaction Design
- Machine Learning
- Cyber Physical Systems
- Information Systems

In addition, all students have a common course in scientific methods.

In the second semester, students specialise in the research areas selected in the first semester. In addition, students work with an applied computer science project.

In the third semester, students have a common course, which focuses on the use of computer science in today’s digital society. In addition, students select an elective course and start on their master’s thesis.

In the fourth semester, students continue and finish their master’s thesis.

Students can choose a long master’s thesis in their second semester.

In order to qualify for the master's thesis, students must have passed at least 50 ECTS from the first year of study.

Teaching, learning methods and forms of assessment

Learning partly takes place through seminars and traditional lectures. Additionally it will take other forms:
-student-led seminars
- projects
- Each student's benefit from this type of organisation will depend on the student's own efforts and interest. The student must show interest in his/her professional development, and must be able to work independently with theory, implementation and knowledge acquisition. The students are offered supervision in all courses of the master programme.
-Master students are expected to take initiatives and approach tutors, and be responsible for their own learning.
-Most courses and assignments are ICT-based, using various IT tools for exchange of information, submission of assignments, tests etc. Østfold University College may demand that the student has a laptop at his/her disposal.

Østfold University College's Makerspace, a well-equipped lab with tools, materials, components and kits, is available for the students 24 hours, 7 days a week. This is a playground for students who like to create something using technology. Makerspace is also an arena for lectures, courses and experiments.

A modern library is at the students' disposal. The library helps the students in developing their information competence, i.e. the ability to search, find, evaluate and use relevant information. In addition to personal service the students are offered library courses on international databases and evaluation of information quality. They are also offered courses in scientific referencing.

Compulsory assignments
Some of the courses have requirements for attendance and/or compulsory assignments, exercises and projects. These assignments have to be completed and approved before taking the final examination. More details regarding compulsory assignments are found in the course descriptions.

Academic writing
The students will be trained in academic writing throughout the study programme. This is done by emphasising content, structure, reliability, and referencing.

Continuous feedback
The taught courses include exercises, assignments and projects, completed either individually or in groups. The students are given feedback on all exercises, assignments and projects.

Assessment
Final assessment takes several forms: written individual examination, portfolio assessment, project assessment, oral examination, or a combination of these.
The grading scale normally used is the A - F scale although some courses may use the assessment "Bestått/Ikke bestått" (Pass/Fail).
Upon agreement with the instructor you may in some cases use Norwegian for your handins and/or exams".
More details regarding assessment are found in the course descriptions.

Plagiarism control/ cheating
Bachelor's and Master's theses are subject to electronic plagiarism control, as may also be the case with other courses and required coursework. Exam papers that are partly or entirely identical will not be approved and will be regarded as cheating. For further information please see Exam regulations for Østfold University College.

Research and development work

The master programme is based on the research activity of the faculty, but is also contributing to our research. All courses in the first two years is directly related to the research performed at the faculty. This implies that our courses are continually developed and always up to date.

The master theses are always based on research and development performed at the faculty or at one of our partners. The students become part of the R&D-groups, and the theses have on several occasions led to scientific publications.

Internationalisation

The international aspect is taken care of by the use of international literature and several members of the academic staff have close contacts with foreign institutions and research environments. The language of instruction is entirely in English, and therefore accommodates the needs of foreign students.

Programme evaluation

We are in need of feedback from our students, and that you participate in the different evaluations that we arrange.

To be able to offer a topical and relevant education of good quality, HiØ is dependent on feedback from the students and their participation in the evaluation. This study programme is regularly evaluated in order to assure and develop its quality.

- An annual national student survey is conducted among second year students on all bachelor's and master's degree programmes under the auspices of the Norwegian Agency for Quality Assurance in Education (NOKUT). The results of the survey are published on the website studiebarometeret.no.

- HiØ conducts periodical programme evaluations.

- Evaluations are carried out in the individual courses; see the individual course descriptions for more details.

Reading list

See separate course descriptions.
Course literature is subject to change until 1st of June for autumn courses, and 1st of December for spring courses.

Studies abroad

Students may take their second semester of their studies at a university abroad. Both the International Coordinator at the faculty as well as HiØ's International Office help accommodate studies abroad, and the faculty have exchange agreements with several universities in Europe, USA, Canada and Australia.

Detailed information on exchange opportunities at universities abroad can be found on HiØ's international pages.

Work and future studies

The master's programme qualifies for PhD-studies in Norway and abroad. Different admission requirements may apply at different universities.

A master degree from us provides opportunities for leading positions within application development, web development, consulting and project management in leading IT companies in Norway and abroad. It also qualifies for work in the public sector, for instance in the fields of research, health and education

The study plan is approved and revised

The study plan is approved

Dean Harald Holone, 27.October 2020

The study plan is revised

Head of Studies, Monica Lind Kristiansen, October 31, 2019

The study plan applies to

2021-2025

Programme Coordinator

Faculty of Computer Science, Engineering and Economics.
Head of Studies Monica Kristiansen Holone

Study model

Autumn 2021

Select the first specialisation

- ○ Interaction Design
- ○ Machine Learning
- ○ Cyber Physical Systems

Information Systems

Core courses

ITI41020
Scientific Methods in Computer Science
10 stp

Spring 2022

Choose specialisation to view courses

Autumn 2022

Select the second specialisation

Interaction Design

Machine Learning

Cyber Physical Systems

Information Systems

Choose specialisation to view courses

Spring 2023

Core courses

ITI41120
Applied Computer Science Project
10 stp

Autumn 2023

Core courses

ITI42320
Computer Science in the Digital Society
10 stp

ITI53020 / Part 1 of 4
Master's Thesis

You may apply for a long Master’s Thesis (60 ECTS)

ITI54020 / Valgbart emne / Part 1 of 4
Master's Thesis

Spring 2024

Core courses

ITI53020 / Part 2 of 4
Master's Thesis

You may apply for a long Master’s Thesis (60 ECTS)

ITI54020 / Valgbart emne / Part 2 of 4
Master's Thesis

Autumn 2024

Core courses

ITI53020 / Part 3 of 4
Master's Thesis

Elective courses

Choose one elective

ITI41520 / Valgbart emne

Interaction Design	**10** stp
ITI41720 / Valgbart emne Machine Learning and Deep Learning	**10** stp
ITI41920 / Valgbart emne Automation, Adaptation and IoT	**10** stp
ITI42122 / Valgbart emne Cyber Security Governance	**10** stp
ITI60020 / Valgbart emne Digital Fabrication and Making	**10** stp

You may apply for a long Master's Thesis (60 ECTS)

ITI54020 / Valgbart emne / Part 3 of 4 Master's Thesis	

Spring 2025

Core courses

ITI53020 / Part 4 of 4 Master's Thesis	**40** stp

You may apply for a long Master's Thesis (60 ECTS)

ITI54020 / Valgbart emne / Part 4 of 4 Master's Thesis	**60** stp

Last updated from FS (Common Student System) July 16, 2024 2:34:48 AM

Emner som ikke er tatt med

Emnesiden finne ikke

- ITI60020 2024h

ITI41020 Scientific Methods in Computer Science (Autumn 2021)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Cathrine Linnes

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Mandatory course in the master programme in applied computer science full-time and part-time.

Lecture Semester

First semester (autumn) in the full-time and part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- has advanced knowledge of research design
- has knowledge of qualitative, quantitative and mixed methods
- has knowledge of relevant data collection techniques
- has knowledge of relevant data analysis techniques
- is familiar with publishing channels relevant to applied computer science, different types of publications, their roles and functions

Skills

The student is able to

- search for and assess scientific literature within the field of applied computer science
- use scientific references
- construct a problem statement or research question and evaluate its soundness
- select relevant methods to address a research problem
- use relevant techniques for data collection
- use relevant techniques for data analysis

General competence

The student

- is able to write a scientific paper
- is able to orally present a scientific project for specialists within the field, as well as the general public
- has competence in critical reading and reflection

Content

- Research design in the field of computer science
- Qualitative, quantitative and mixed methods
- Scientific writing and peer-review
- Publishing channels relevant to applied computer science, different types of publications, their roles and functions

Forms of teaching and learning

Lectures, presentations, and lab exercises.

Workload

Approx. 280 hours.

Examination

Portfolio and individual written exam.

The exam consists of both a portfolio and an individual written exam. The portfolio counts 60% and consists of:

- Two research projects
- One idea paper

The individual written exam counts 40% and is based on the course curriculum. Duration 2 hours. No supporting materials permitted.

Both parts of the exam must be passed to pass the exam as a whole.

The student will get an individual joint grade for the entire course. Grades: Assessment on the A - F grading scale.

Examiners

External and internal examiner, or to internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 06.08.2021.

The following textbook is required for this course:

- Salkind, N.J. (2016). Exploring Research (9th ed), ISBN: 978-1292156293.

Additional course material will be posted on the learning platform.

Last updated from FS (Common Student System) July 16, 2024 2:32:49 AM

ITI41520 Interaction Design (Autumn 2021)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Georgios Marentakis

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Recommended requirements

The student should have some experience on investigating or creating interactive systems evidenced through the completion of relevant courses in software and hardware prototyping, data science, cyber-physical systems, interaction design, design methods, or human computer interaction.

Lecture Semester

First or third semester (autumn) in the full-time programme.

First, third or seventh semester (autumn) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

By participating in the theoretical modules the student learns to:

- identify and describe contemporary trends in interaction design and classify IxD projects accordingly
- formulate research questions and hypotheses for interaction design projects
- understand and reflect on choosing a design methodology for a specific project and appropriate documentation techniques
- explain and reflect on the results of interaction design projects

Skills

The student is able to

- perform a scientific literature review
- apply appropriate methods for designing new and existing technology for interaction between humans and machines
- prototype interaction design ideas at an appropriate level of fidelity using contemporary tools and techniques
- design, conduct, and analyse surveys, interviews, or experiments as appropriate
- document the findings in a way appropriate for scientific publication and present appropriately

General competence

The student gains

- competence in critical reading and reflection
- group-work skills

Content

The theoretical modules investigate contemporary trends in interaction design and design methods. Presentations demonstrate multimodal interfaces, virtual and augmented reality, mobile and wearable, as well as tangible, embodied, and sonic interaction design. Design methods from user-centred and participatory design as well as designing for specific user groups and research through design are discussed.

In project work, the student applies appropriate methods to investigate interaction between humans and machines. Projects may survey specific use contexts, create an interactive artifact or system, perform an empirical study, or a meta-analysis of a literature review. Projects are structured based on deadlines which offer the student several opportunities to get formative feedback during supervision. The results of the project should be documented in a way that is appropriate for scientific publication and may be submitted to an international academic conference.

Forms of teaching and learning

The Interaction design course is an advanced course in Interaction Design which consists of theoretical and project work modules. The following methods are used:

- Presentations
- Supervision
- Workshops
- Reading Circle
- Project Work

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must during the semester:

- present at least one paper from the reading list (individual or in a group) and lead a plenary discussion based on this.
- complete and present at least one approved group project.

Coursework requirements must be accepted to qualify for the exam.

Examination

Individual written exam and scientific documentation in groups

The exam is divided into two parts:

- Individual written exam (50%) based on the course curriculum. Duration 4 hours. No supporting materials allowed.
- Scientific documentation in groups (50%) based on the group project.

Grading scale A - F in both parts. Both parts of the exam must be passed to pass the course. The student will get an individual joint grade for the entire course.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 21.10.2020. The reading list may be subject to changes before 1st of June 2021.

The curriculum will be supported by material (scientific papers) during the semester. The material will be posted on the learning platform.

Last updated from FS (Common Student System) July 16, 2024 2:32:49 AM

ITI41720 Machine Learning (Autumn 2021)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Roland Olsson

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science full-time and part-time.

Recommended requirements

Knowledge in:

- Statistics and statistical programming
- Mathematics
- Programming
- Algorithms and data structures

Lecture Semester

First or third semester (autumn) in the full-time programme.

First, third or seventh semester (autumn) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- is familiar with both the possibilities and advantages of employing the machine learning methods in the course, as well as possible problems that may be encountered and how to overcome them.
- knows how the algorithms presented in the course work and their characteristics, for example which problems they work best for, overfitting, expected accuracy and computational requirements, for example how much benefit that accelerators may provide.

Skills

Given a machine learning application, the student is able to

- determine which theory and which methods that are presented in the course that are relevant and also how to apply them.
- perform hyperparameter tuning or in some cases even perform modifications of the source codes.
- use at least one implementation for each of the major machine learning techniques that are taught in the course.

General competence

The student

- is able to independently read machine learning papers and other literature and evaluate what works well and what does not for new problems.
- knows the terminology of machine learning and is familiar with the mathematics that is common in the field.
- knows the general behaviour of machine learning methods for example regarding how much data that is required, how to preprocess the data and ensure that its quality is sufficient.

Content

This course gives an advanced insight into the main methods used in machine learning. The topics covered in this course are:

- Concepts related to basic types of learning (supervised, unsupervised, reinforcement): preprocessing, feature extraction, overfitting, error functions.
- Decision and regression trees, random forest and XGBoost
- Artificial neural networks, deep learning.
- Optimization (evolutionary algorithms and other search methods)
- Bayesian inference / classification.

Ethics and privacy in machine learning is also mentioned.

Additionally, the course contains up to date topics that are not known when this text is being written.

Forms of teaching and learning

The students will learn by attending lectures, read the books, papers and online material in the course reading list and above all by working on two projects. The project work is supervised each week and results in a 10 pages report for each project. These reports are part of the examination in the course.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must have finished both of their projects.

Coursework requirements must be accepted to qualify for the exam.

Examination

Portfolio and individual written exam

The exam consists of both a portfolio and an individual written exam.

The portfolio (determines 65% of the final grade) consists of two projects. The projects can be carried out individually or in groups of two students. The students will get an individual grade.

The individual written exam determines 35% of the final grade and focuses on theory. Duration 3 hours. No supporting materials allowed.

Both parts of the exam must be passed to pass the exam as a whole. The student will get an individual joint grade for the entire course. Grades: A - F.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 22.10.2020. The reading list may be subject to changes before 1st of June 2021.

Books, papers and online materials posted on the learning platform.

ITI41920 Hands-On Introduction to Cyber-Physical Systems (Autumn 2021)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Maben Rabi

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Lecture Semester

First or third semester (autumn) in the full-time programme.

First, third or seventh semester (autumn) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The students has knowledge of

- examples of Cyber-physical systems, arising in industry and society
- basic Sensing, Communication, Control, and Computing elements in a typical Cyber-physical system
- techniques for modelling Cyber-physical systems from their components
- the challenges of simulating, designing, testing a verifying cyber-physical systems
- basic simulation methods

Skills

The student is able to

- decompose any given Cyber-physical system into its Sensing, Communication, Control, and Computing elements

- apply basic modelling methods to capture the dynamic behaviour
- predict dynamic behaviour using simulation tools
- predict performance from approximate modelling and analysis
- perform simulation and testing of simple Arduino or Raspberry-Pi based mechatronic Cyber-physical systems

General competence

The student

- knows the way of abstracting device and embedded software details, and extracting the overall functional behaviour, in concrete examples of cyber-physical systems
- is familiar with the terminology of the area of cyber-physical systems

Content

- Introduction to embedded computing devices
- Introduction to basic sensing, actuating and other physical devices
- Common communication protocols for real-time applications
- Modelling of continuous and discrete dynamics
- Challenges in specification, verification, and systems engineering

Forms of teaching and learning

Lectures, seminar/workshops, and project work with software tools and mechatronic hardware.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must:

- deliver up to 5 mandatory assignments
- finish their final project work

These assignments require working with software packages, hardware implementation and programming, as well as reading and summarizing papers from the research literature. The final project also involve similar activities.

Coursework requirements must be accepted to qualify for the exam.

Examination

Oral exam and project report in groups

The exam is divided into two parts:

- Oral exam in groups (50%): Based on the course curriculum. Duration 30 min. No supporting materials allowed. The students will get an individual grade.
- Project report in groups (50%): Based on the project work.The students will get an individual grade.

Grading scale A - F in both parts. Both parts of the exam must be passed to pass the exam as a whole.The students will get an individual joint grade for the entire course.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 9.3.2021. The reading list may be subject to changes before 1st of June 2021.

Main textbooks:

- E. A. Lee and S. A. Seshia: Introduction to Embedded Systems: A Cyber-Physical Systems Approach, ISBN: 978-0262533812
- P. Fritzson, Principles of Object-Oriented Modeling and Simulation with Modelica 3.3: A Cyber-Physical Approach, Second edition (2015), IEEE-Wiley press, ISBN: 978-1118859124
- Derek Molloy, Exploring Raspberry Pi, first edition (2016), Wiley, ISBN: 978-1119188681
- Stamatios Manesis and George Nikolakopoulos, Introduction to Industrial Automation, first edition (2018), CRC press, ISBN-13: 978-1498705400
- Course notes made available through Canvas

Additional literature:

- Terry Bartelt, Industrial automated systems, first edition (2010), Delmar, ISBN-13: 978-1435488885.
- Peter Corke, Robotics, Vision and Control, second edition (2016), Springer, ISBN-13: 978-3319544120.
- P. Marwedel, Embedded System Design: Embedded Systems, Foundations of Cyber-Physical Systems and the Internet of Things, Fourth edition (2020), Springer, 978-3030609092
- Marilyn Wolf, Embedded system interfacing: Design for the Internet of things (IoT) and Cyber-physical systems (CPS), 2019, Morgan-Kaufmann
- Richard Crowder, Electric drives and electromechnical systems: applications and control, Second edition (2019), Butterworth-Heinemann, ISBN: 0081028849
- A. Platzer, Logical foundations of cyber-physical systems, first edition (2018), Springer, 978-3319635873
- Chris Hobbs, Embedded Software Development for Safety-Critical Systems, Second edition (2019), Routledge, ISBN: 978-0367338855.
- K. Rozhdestvensky et al., Computer modelling and simulation of dynamic systems using Wolfram SystemModeler, first edition (2020), ISBN: 978-9811528026.
- Daniele Lacamera, Embedded Systems Architecture, first edition (2018), Packt publishing, ISBN: 978-1788832502
- Brian Amos, Hands-on RTOS with microcontrollers, first edition (2020), Packt publishing, ISBN-13: 978-1-83882-673-4
- C. Kormanyos, Real-time C++, first edition (2018), Springer, ISBN-13: 978-3662567173

Last updated from FS (Common Student System) July 16, 2024 2:32:50 AM

ITI42120 Advanced Topics in Information Systems (Autumn 2021)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Ricardo Colomo-Palacios

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Recommended requirements

Knowledge about:

- DevOps tools and frameworks
- Machine learning

Lecture Semester

First or third semester (autumn) in the full-time programme.

First, third or seventh semester (autumn) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- knows the basics of Digital Transformation
- understands the role of Information Systems in Digital Transformation initiatives
- is familiar with principles of IT Governance
- is familiar with global trends in business software with regards to its deployment and development
- has a good overview of DevOps tools and approaches
- knows how to measure costs in software development in DevOps and traditional settings
- understands the role and importance of Software Engineering in Artificial Intelligence-based solutions

Skills

The student

- is able to place and articulate Information Systems function in a Digital Transformation initiative
- is able to identify and design a basic IT Governance plan
- knows the different approaches and trends in business software
- knows how to measure business value and costs in software developments
- is able to identify the need to use a set of DevOps software tools
- is capable of using a set of DevOps software tools for business needs
- knows how to use and justify the use of software engineering techniques in Artificial Intelligence-based solutions

General competence

The student is able to apply

- scientific theories and methodologies in a practical business setting.
- technologies in a practical business setting.

Content

1. Digital Transformation and IT Governance
2. Trends in Business Software
3. DevOps and Continuous Software Engineering
4. Software Engineering for Artificial Intelligence

Forms of teaching and learning

Teaching will be based on blended learning approaches. There will be recorded lectures of the topics of the course and in a weekly or bi-weekly basis, physical meetings will take place to mentor the development of the paper and guide students in the course.

Workload

Approx. 280 hours.

Examination

Scientific paper and individual oral exam

The students need to develop a scientific paper on a selected topic. The topic is chosen by the students and agreed with the course responsible. The paper can be developed individually or in groups two students. The students are given an individual tentative grade on the paper using the A - F grading scale. This grade can be adjusted up to 2 stages at the oral exam.

The individual oral exam is based on regular topics in the course, aspects of the paper developed and a case. Duration approx. 20-30 min. No supporting materials allowed.

If the student decides to challenge the assessment, the scientific paper must be re-assessed. If the new assessment affects the tentative grading of the paper, a new oral exam will be arranged.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, both parts of the examination must be retaken.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 21.10.2020. The reading list may be subject to changes before 1st of June 2021.

DIGITAL TRANSFORMATION & IT GOVERNANCE

Andriole, S. J. (2017). Five myths about digital transformation. MIT sloan management review, 58(3).

Ebert, C., & Duarte, C. H. C. (2018). Digital transformation. Ieee Software, (4), 16-21.

Hess, T., Matt, C., Benlian, A., & Wiesböck, F. (2016). Options for formulating a digital transformation strategy. MIS Quarterly Executive, 15(2).

Hinings, B., Gegenhuber, T., & Greenwood, R. (2018). Digital innovation and transformation: An institutional perspective. Information and Organization, 28(1), 52-61.

Vial, G. (2019). Understanding digital transformation: A review and a research agenda. The Journal of Strategic Information Systems, 28(2), 118-144.

De Haes, S., & Van Grembergen, W. (2009). An exploratory study into IT governance implementations and its impact on business/IT alignment. Information Systems Management, 26(2), 123-137.

Joshi, A., Bollen, L., Hassink, H., De Haes, S., & Van Grembergen, W. (2018). Explaining IT governance disclosure through the constructs of IT governance maturity and IT strategic role. Information & Management, 55(3), 368-380.

Mohan, K., Cao, L., Sarkar, S., & Ramesh, B. (2019). Adapting IT Governance Practices for the Changing IT Function. IT Professional, 21(1), 27-33.

Juiz, C., & Toomey, M. (2015). To govern IT, or not to govern IT?. Communications of the ACM, 58(2), 58-64.

Vejseli, S., Proba, D., Rossmann, A., & Jung, R. (2018). The agile strategies in IT Governance: Towards a framework of agile IT Governance in the banking industry. Twenty-Sixth European

Conference on Information Systems (ECIS2018), Portsmouth,UK, 2018

https://www.pmi.org/disciplined-agile/process/it-governance

https://www.projectmanagement.com/blog-post/61871/What-is-Lean-IT-Governance-

TRENDS IN BUSINESS SOFTWARE

Berberat, S., & Baudet, C. (2019). Assessing a Business Software Application using Strategic IT Alignment Factors: A New Way for IS Evaluation?.

De Lauretis, L. (2019). From Monolithic Architecture to Microservices Architecture. In 2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW) (pp. 93-96). IEEE.

Jansen, S., Cusumano, M., & Popp, K. M. (2019). Managing software platforms and ecosystems. IEEE Software, 36(3), 17-21.

Jason, G., Nikolay, M., & Guohua, W. The Partner Ecosystem Evolution from On-premises Software to Cloud Services: a case study of SAP.

Loukis, E., Janssen, M., & Mintchev, I. (2019). Determinants of software-as-a-service benefits and impact on firm performance. Decision Support Systems, 117, 38-47.

Raghavan R., S., K.R., J. & Nargundkar, R.V. (2020). Impact of software as a service (SaaS) on software acquisition process, Journal of Business & Industrial Marketing, 35(4), 757-770. https://doi.org/10.1108/JBIM-12-2018-0382

Yrjönkoski, T., & Systä, K. (2019, August). Productization levels towards whole product in SaaS business. In Proceedings of the 2nd ACM SIGSOFT International Workshop on Software-Intensive Business: Start-ups, Platforms, and Ecosystems (pp. 42-47).

DEVOPS AND CONTINUOUS SOFTWARE ENGINEERING

Fitzgerald, B., & Stol, K. J. (2017). Continuous software engineering: A roadmap and agenda. Journal of Systems and Software, 123, 176-189.

O'Connor, R. V., Elger, P., & Clarke, P. M. (2017). Continuous software engineering—A microservices architecture perspective. Journal of Software: Evolution and Process, 29(11), e1866.

Johanssen, J. O., Kleebaum, A., Paech, B., & Bruegge, B. (2018, May). Practitioners' eye on continuous software engineering: an interview study. In Proceedings of the 2018 International Conference on Software and System Process (pp. 41-50).

Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. Ieee Software, 33(3), 94-100.

Zhu, L., Bass, L., & Champlin-Scharff, G. (2016). DevOps and its practices. IEEE Software, 33(3), 32-34.

Humble, J., & Molesky, J. (2011). Why enterprises must adopt devops to enable continuous delivery. Cutter IT Journal, 24(8), 6.

https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=527148

https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=638576

SOFTWARE ENGINEERING FOR ARTIFICIAL INTELLICENCE

Menzies, T. (2019). The Five Laws of SE for AI. IEEE Software, 37(1), 81-85.

McDermott, T., DeLaurentis, D., Beling, P., Blackburn, M., & Bone, M. (2020). AI4SE and SE4AI: A Research Roadmap. INSIGHT, 23(1), 8-14.

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., ... & Zimmermann, T. (2019, May). Software engineering for machine learning: A case study. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP) (pp. 291-300). IEEE.

Last updated from FS (Common Student System) July 16, 2024 2:32:50 AM

ITI41620 Design for Cooperation (Spring 2022)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Joakim Karlsen

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Recommended requirements

ITI41520 Interaction Design

The student's learning outcomes after completing the course

Knowledge:

The student is familiar with

- central theories and concepts for understanding how people cooperate in workplaces or in everyday life
- methods, tools and techniques for designing IT solutions in support of cooperative practices

Skills:

The student is able to

- use methods, tools, and techniques for designing IT solutions in support of cooperative practices

General competence

The student can

- conduct methodologically and ethically sound research according to the scientific standards in CSCW

Content

The course introduces the students to Computer-Supported Cooperative Work (CSCW), an interdisciplinary research field concerned with understanding how to support cooperative practices by technology design.

The course gives an overview of central theories and concepts for understanding how people cooperate in workplaces or in everyday life and a toolbox of methods, tools and techniques that will help them design IT solutions in support of these cooperative practices.

Forms of teaching and learning

There will be lectures, plenary discussions and project work.

The lectures and plenary discussions will focus on:

- classical case studies in CSCW
- central theories and concepts in CSCW
- methods, tools and techniques for designing support for cooperative practices as developed in CSCW

The project work will lead to the writing of one scientific paper and will be conducted according to the requirements and deadlines set by the professors responsible for the course.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must have finished their project work.

Coursework requirements must be accepted to qualify for the exam.

Examination

Individual oral exam and scientific paper in groups

The exam is divided into two parts:

- Individual oral exam (50%): Based on the course curriculum. Duration 30 min. Some supporting material will be allowed. This will be specified by the lecturer.
- Scientific paper in groups (50%): Based on the project work.

Grading scale A - F in both parts. Both parts of the exam must be passed to pass the exam as a whole.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 23.11.2021

Co-design

Brandt, Eva, Thomas Binder, and Elizabeth B.-N. Sanders. 2013. 'Tools and Techniques: Ways to Engage Telling, Making and Enacting'. In Routledge International Handbook of Participatory Design, edited by Jesper Simonsen and Toni Robertson, 145–81. London: Routledge.

Bratteteig, T., & Wagner, I. (2012, August). Disentangling power and decision-making in participatory design. In Proceedings of the 12th Participatory Design Conference: Research Papers-Volume 1 (pp. 41-50).

Kensing, Finn, and Joan Greenbaum. 2013. 'Heritage: Having a Say'. In Routledge International Handbook of Participatory Design, edited by Jesper Simonsen and Toni Robertson, 21–37. London: Routledge.

Sanders, Elizabeth B.-N., and Pieter Jan Stappers. 2008. 'Co-Creation and the New Landscapes of Design'. CoDesign 4 (1): 5–18.

Cooperation

Schmidt, Kjeld, and Liam Bannon. 1992. ‘Taking CSCW Seriously’. Computer Supported Cooperative Work (CSCW) 1 (1–2): 7–40.

Schmidt, Kjeld. 2011. ‘The Concept of “Work” in CSCW’. Computer Supported Cooperative Work (CSCW) 20 (4–5): 341–401.

Boundary Objects and Coordination mechanisms

Schmidt, Kjeld, and Carla Simonee. 1996. ‘Coordination Mechanisms: Towards a Conceptual Foundation of CSCW Systems Design’. Computer Supported Cooperative Work (CSCW) 5 (2–3): 155–200.

Star, Susan Leigh, and James R. Griesemer. 1989. ‘Institutional Ecology, `Translations’ and Boundary Objects: Amateurs and Professionals in Berkeley’s Museum of Vertebrate Zoology, 1907-39’. Social Studies of Science 19 (3): 387–420.

Awareness

Dourish, Paul, and Victoria Bellotti. 1992. ‘Awareness and Coordination in Shared Workspaces’. In Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work - CSCW ’92, 107–14. Toronto, Ontario, Canada: ACM Press.

Heath, Christian, and Paul Luff. 1992. ‘Collaboration and Control: Crisis Management and Multimedia Technology in London Underground Line Control Rooms’. Computer Supported Cooperative Work (CSCW) 1 (1): 69–94.

Place and space

Harrison, Steve, and Paul Dourish. 1996. ‘Re-Place-Ing Space: The Roles of Place and Space in Collaborative Systems’. In Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work, 67–76. CSCW ’96. New York, NY, USA: ACM.

Luff, Paul, and Christian Heath. 1998. ‘Mobility in Collaboration’. In Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work, 305–14. CSCW ’98. New York, NY, USA: ACM.

Example of CSCW studies:

Verne, Guri, and Tone Bratteteig. 2016. ‘Do-It-Yourself Services and Work-like Chores: On Civic Duties and Digital Public Services’. Personal and Ubiquitous Computing 20 (4): 517–32.

Gasser, Les. 1986. ‘The Integration of Computing and Routine Work’. ACM Transactions on Information Systems 4 (3): 205–25.

Future discussions:

Mutlu, Bilge, and Jodi Forlizzi. 2008. ‘Robots in Organizations: The Role of Workflow, Social, and Environmental Factors in Human-Robot Interaction’. In 2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI), 287–94.

Niklasson, Axel. 2020. ‘AI for Teams: The Future of Assisted Collaborative Work’.

Bringing PD and CSCW together

Kensing, Finn, and Jeanette Blomberg. 1998. ‘Participatory Design: Issues and Concerns’. Computer Supported Cooperative Work (CSCW) 7 (3–4): 167–85.

Bratteteig, T., & Wagner, I. (2016). Unpacking the notion of participation in participatory design. Computer Supported Cooperative Work (CSCW), 25(6), 425-475.

Last updated from FS (Common Student System) July 16, 2024 2:32:55 AM

ITI41820 Advanced Topics in Machine Learning (Spring 2022)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Kazi Shah Nawaz Ripon

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science full-time and part-time.

Recommended requirements

ITI41720 Machine Learning

Lecture Semester

Second semester (spring) in the full-time programme.

Second or fourth semester (spring) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- is the possibilities and advantages of employing the machine learning methods in the course as well as possible problems that may be encountered and how to overcome them.
- knows how the algorithms presented in the course work and their characteristics, for example which problems they work best for, overfitting, expected accuracy and computational requirements, for example how much benefit that accelerators may provide.

Skills

Given a machine learning application, the student is able to

- determine which theory and which methods that are presented in the course that are relevant and also how to apply them.
- perform hyperparameter tuning or in some cases even perform modifications of the source codes.
- use at least one implementation for each of the major machine learning techniques that are taught in the course.

General competence

The student

- is able to independently read machine learning papers and other literature and evaluate what works well and what does not for new problems.
- knows the terminology of machine learning and be familiar with the mathematics that is common in the field.
- knows the general behaviour of machine learning methods for example regarding how much data that is required, how to preprocess the data and ensure that its quality is sufficient.

Content

The course goes in depth on selected topics and methods within machine learning and their applications. Examples include:

- advanced neural net and deep learning models, such as: ResNET, Zero shot, GAN, LSTM.
- Evolutionary and bio-inspired algorithms algorithms (like GA, EA, ES, PSO, ACO, AIS) in search, optimization and classification.
- Program induction. Symbolic regression. Automatic programming.
- Markov models, Kernel methods. SVM

- Implementing machine learning in Industries and business
- Machine learning challenges and future
- Philosophical fundamental problems and ethical questions related to machine learning

The course syllabus will continuously be updated with methods from state-of-the-art research. Other topics may be chosen by machine learning group members each year and may vary depending on who is involved.

Forms of teaching and learning

The students will learn by attending seminars, papers and online material in the course reading list and above all by working on a project with a selected topic throughout the course and giving presentations at the seminars.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must:

- give presentations at two seminars.

Coursework requirements must be accepted to qualify for the exam.

Examination

Project report and individual oral exam

The assessment is based on the project report and an individual oral exam. The project report is graded on the A - F grading scale. It is given a tentative grade of the report. This grade can be adjusted up to 2 stages at the oral exam.

The individual oral exam based on the course curriculum and project work. Approximately 30 minutes duration. No supporting materials allowed.

Examiners

External and internal examiner, or to internal examiners.

Conditions for resit/rescheduled exams

In case of re-examination, a new project must be carried out in agreement with the course instructor.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 22.10.2020. The reading list may be subject to changes before 1st of December 2021.

Online materials posted on the learning platform.

Last updated from FS (Common Student System) July 16, 2024 2:32:55 AM

ITI42020 Modelling Cyber-Physical Systems (Spring 2022)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Øystein Haugen

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Recommended requirements

ITI41920 Hands-on Introduction to Cyber-Physical Systems and general programming skills

Lecture Semester

Second semester (spring) in the full-time programme.

Second or fourth semester (spring) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student understands

- the challenges associated with cyber-physical systems
- the relevance of good software design principles
- how evolution and maintenance should be organized
- the value of abstraction

Skills

The student has the capability to

- model and implement reactive systems with concurrency
- perform analysis of consistency of models of systems with concurrency
- give and take constructive criticism of the system design and functioning
- receive the experience of building a cyber-physical system and making it execute

General competence

The student

- can build systems on «Internet of Things»
- can assess realistically what errors may occur in cyber-physical systems and how to minimize their vulnerability
- has some insight into precise descriptions and their semantics

Content

The course focus on how reactive systems can be built with emphasis on modeling. The models are executable and based on state machines. The requirements of these concurrent systems are modeled as sequence diagrams, and it is emphasized that the requirements and design should be consistent.

We emphasize reactive systems on the Internet of Things, and we use a running example where the functionality is enhanced during the course following an agile approach.

Towards the end of the course, we show how systems can be made more resilient to unexpected incidents and errors. To perform risk-analysis of such systems will also be covered.

Forms of teaching and learning

Project work, lectures and guided lab and exercises.

For each instance of the course, we create new project tasks. We teach the project teams how to give and take constructive feedback.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

- Mandatory project: There will be one project, with deliverable at each teaching session (2 deliverables). Project group size should be 2-4 people, but with few students, single person project will be possible. The students should expect to spend 100 hours on the project.

- Plenary presentation and evaluation of the project. The project should normally result in an executable model that should be demonstrated at the plenary presentation.

Coursework requirements must be accepted to qualify for the exam.

Examination

Individual oral exam

Individual oral exam based on the course curriculum and mandatory exercises. Approximately 30 minutes duration. No supporting materials allowed.

Assessment on the A - F grading scale.

Examiners

External and internal examiner, or two internal examiners.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 21.10.2020. The reading list may be subject to changes before 1st of December 2021.

A number of articles and lecture notes constitute the curriculum. The articles and notes will be posted on the learning platform.

Last updated from FS (Common Student System) July 16, 2024 2:32:55 AM

ITI42220 Security in Information Systems and Software Engineering (Spring 2022)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Elective course in the master programme in applied computer science, full-time and part-time.

Recommended requirements

ITI41820 Advanced Topics in Information Systems

Lecture Semester

Second semester (spring) in the full-time programme.

Second or fourth semester (spring) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- knows how to specify and analyse security requirements in the development and acquisition process
- understands the differences between security concerns in software acquisition and development environments
- is able to recognize common security engineering risk analysis tools and methods
- knows main secure design considerations
- is able to understand main testing approaches for security

- is familiar with common DevSecOps toolchains and configurations
- is able to identify and understand maturity models in security and DevSecOps scenarios

Skills

The student is

- able to use common security requirements engineering methods (e.g. MSRA, SQUARE, GBRAM…) in real settings
- able to perform a security risk analysis
- able to justify the need of DevSecOps approaches and guide them in practical business settings
- able to secure DevOps scenarios
- able to introduce security in software design and coding phases
- able to use main testing approaches with an accent in security
- capable of using a set of DevSecOps software tools for business needs
- able to measure security levels by means of metrics and models

General competence

The student is able to apply theories and methodologies in the course in a practical business setting.

Content

1. Security in Requirements: Development and Acquisition
2. Security Engineering Risk Analysis

3. DevSecOps: Strategy and Implementation

Forms of teaching and learning

Teaching will be based on blended learning approaches. There will be recorded lectures of the topics of the course and in a weekly or bi-weekly basis, physical meetings will take place to mentor the development of the paper and guide students in the course.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must deliver up to four assignments.

Coursework requirements must be accepted to qualify for the exam.

Examination

Individual written exam and scientific paper

The exam is divided into two parts:

- Individual written exam (50%) based on the course curriculum. Duration 4 hours. No supporting materials allowed.

- Scientific paper (50%) on a topic related to the course. The topic is chosen by the students and agreed with the course responsible. The paper can be developed individually or in groups of two students. The students will get an individual grade.

Grading scale A - F in both parts. Both parts of the exam must be passed to pass the course. The student will get an individual joint grade for the entire course.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination can be retaken.

Course evaluation

This course is evaluated by a:

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Last updated 21.10.2020. The reading list may be subject to changes before the 1st of December 2021.

Ahmed, Z., & Francis, S. C. (2019, November). Integrating Security with DevSecOps: Techniques and Challenges. In 2019 International Conference on Digitization (ICD) (pp. 178-182). IEEE.

Carter, K. (2017). Francois raynaud on DevSecOps. IEEE Software, 34(5), 93-96.

Hsu, T. H. C. (2018). Hands-On Security in DevOps: Ensure continuous security, deployment, and delivery with DevSecOps. Packt Publishing Ltd.

https://insights.sei.cmu.edu/devops/

https://insights.sei.cmu.edu/sei_blog/2018/12/threat-modeling-12-available-methods.html

Kumar, R., & Goyal, R. (2020). Modeling continuous security: A conceptual model for automated DevSecOps using open-source software over cloud (ADOC). Computers & Security, 97, 101967.

McGraw, G., Bonett, R., Figueroa, H., & Shepardson, V. (2019). Security Engineering for Machine Learning. Computer, 52(8), 54-57.

Mead, N. R., & Woody, C. (2016). Cyber Security Engineering: A Practical Approach for Systems and Software Assurance. Addison-Wesley Professional.

Mead, N. R. (2007). How to compare the Security Quality Requirements Engineering (SQUARE) method with other methods. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

Merkow, M. (2019). Secure, Resilient, and Agile Software Development. CRC Press.

Morales, J., Turner, R., Miller, S., Capell, P., Place, P., & Shepard, D. J. (2020). Guide to Implementing DevSecOps for a System of Systems in Highly Regulated Environments. Software Engineering Institute.

Myrbakken, H., & Colomo-Palacios, R. (2017). DevSecOps: a multivocal literature review. In International Conference on Software Process Improvement and Capability Determination (pp. 17-29). Springer, Cham.

O'Connor, R.V., & Colomo-Palacios, R. (2018). Security Awareness in the Software Arena. In Engemann, K. (Eds.). The Routledge Companion to Risk, Crisis and Security in Business. :Routledge.

Sánchez-Gordón, M., & Colomo-Palacios, R. (2020). Security as Culture: A Systematic Literature Review of DevSecOps. In Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops (pp. 266-269).

ITI41120 Applied Computer Science Project (Spring 2023)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Stefano Nichele

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Mandatory course in the master programme in applied computer science full-time and part-time.

Absolute requirements

Passed at least one specialisation course in the first semester

Lecture Semester

Second semester (spring) in the full-time programme.

Fourth semester (spring) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- has deep knowledge of at least one research area at the faculty
- is familiar with relevant publishing channels
- is familiar with relevant methods and techniques

Skills

The student is able to

- write a scientific paper aimed at publishing

- define and defend a research problem
- develop and carry out a research design

General competence

The student gains experience with project work, including planning, performing and reporting in an existing research area

Content

In this course, the students will complete a project based on their two specialisations. The topic for the project is chosen in collaboration with the supervisor and shall, as a general rule, be linked to an existing research area at the department.

The project should be applied within a domain relevant to the digital society.

Forms of teaching and learning

Practical project work with regular supervision.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must deliver:

- a project description within two weeks after starting the project
- a mid-term report

Coursework requirements must be accepted to qualify for the exam.

Examination

Paper and individual oral exam

The assessment is based on the paper and an individual oral exam.

The paper (approx. 10 pages) is graded on the A - F grading scale. It is given a tentative grade of the paper. This grade can be adjusted at the oral presentation. The papers can be carried out individually or in groups of two students. The students will get an individual grade. The paper must be passed before the oral presentation can be carried out

The individual oral exam consists of a presentation and discussion of the paper. Duration approx. 20-30 min. Except the presentation, no supporting materials are allowed.

If the student decides to challenge the assessment, the paper must be re-assessed. If the new assessment affects the tentative grading of the paper, a new oral exam will be arranged.

Examiners

External and internal examiner, or to internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, both parts of the examination must be retaken. Upon re-examination, a new project must be carried out.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee at the Department of Computer Science and Communication.

Literature

The current reading list for 2023 Spring can be found in Leganto

Last updated from FS (Common Student System) July 17, 2024 12:15:30 AM

ITI42320 Computer Science in the Digital Society (Autumn 2023)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Hasan Ogul

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Mandatory course in the Master`s Programme in Applied Computer Science full-time and part-time.

Absolute requirements

The student must have passed at least 30 ECTS before starting this course.

Lecture Semester

Third semester (autumn) in the full-time programme.

Fifth semester (autumn) in the part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- has a good overview of the field of applied computer science
- has an in-depth understanding of relevant scientific publication channels
- has deep knowledge of relevant scientific approaches and methodologies relevant to applied computer science
- is familiar with seminal papers relevant to the research group core areas
- understands the link between computer science and the domains in which it is applied

- understands cooperation between academic research and partners in public and private sectors

Skills

The student

- is able to read and review papers within the field of computer science
- is able to communicate the connection between computer technology and relevant application areas in society
- is able to discuss with peers and researchers about technological challenges, opportunities and applications in society
- is able to communicate research design, results and impact on society
- has the capacity to critically reflect on the role of technology in society

General competence

The student has

- gained experience in critical thinking and discussion
- a more mature view on digital technology and its role in society
- gained experience in oral presentations and plenary discussions
- a broader understanding of the field of applied computer science, relevant publication channels and research topics

Content

In this course, the students will apply their knowledge from their specialisations in the MA programme and use it in a larger context.

In the first part of the course, the department's four research groups in:

- Information Systems ans Software Engineering
- Cyber-Physical Systems
- Interaction Design
- Machine Learning

will present topics and projects related to the digital society, as well as discuss technical and ethical challenges in their domains. Research projects where the department is involved will be presented as cases in the course (including ongoing PhD projects at the department). External partners in the projects will take part to clearly communicate the impact and importance of the research projects in today's digital society.

The second part of the course consists mainly of group sessions and supervision, where the students are tasked with reading and critical analysis of papers relevant to and largely based on the introductory part of the course.

In this part, the students will write:

- a systematic literature review (SLR) about computer science methods/methodologies/technologies (in the field of any of the four research groups) applied on a specific digital society problem/challenge. The SLR topic is chosen by the students and agreed with the course responsible.
- a project proposal, which addresses the specific digital society problem reviewed in the SLR and offers a computer science solution in the scope of any of the four research groups.

Forms of teaching and learning

Lectures, presentations, group sessions and supervision.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must:

- Deliver a systematic literature review (SLR) report within 6 weeks after starting the semester.
- Deliver a project proposal (initial version - abstract) within 8 weeks after starting the semester.
- Attend a group session to present own proposal and get feedback from the course responsible and other students.
- Attend at least four group sessions to give feedback to other students' projects.

Coursework requirements must be accepted to qualify for the exam.

Examination

Project proposal and individual oral exam

The assessment is based on the project proposal and an individual oral exam.

The proposal can be developed individually or in groups two students. The students are given an individual tentative grade on the proposal using the A - F grading scale. This grade can be adjusted up to 2 stages at the oral exam.

The project proposal must be passed before the oral exam can be carried out.

The individual oral exam consists of a presentation and discussion of the project proposal. Duration approx. 20-30 min. Except the presentation, no supporting materials are allowed.

If the student decides to challenge the assessment, the project proposal must be re-assessed. If the new assessment affects the tentative grading of the proposal, a new oral exam will be arranged.

Examiners

External and internal examiner, or two internal examiners.

Conditions for resit/rescheduled exams

Upon re-examination, both parts of the examination must be retaken.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee at the Department of Computer Science and Communication.

Literature

The reading list may be subject to changes before 15th of May 2022.

Last updated from FS (Common Student System) July 17, 2024 12:15:26 AM

ITI53020 Master's Thesis (Autumn 2023–Spring 2025)

Facts about the course

ECTS Credits:
40

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Jan Høiberg

Teaching language:
English

Duration:
2 years

Table of contents

The course is connected to the following study programs

Elective course in the master programme in Applied Computer Science, part-time.

Absolute requirements

The student must have passed at least 50 ECTS before starting this course (students who started in 2020 or earlier must have passed at least 45 ECTS).

Lecture Semester

Fifth, sixth, seventh and eighth semester.

The student's learning outcomes after completing the course

Knowledge

The student has

- knowledge and expertise required for challenging jobs in research and development in the field
- knowledge about literature and methods related to the subjects that are part of the master thesis

Skills

The student is able to

- work independently with a complex problem over a longer period of time
- analyse a situation, describe a problem and plan its solution

- collect and analyse relevant information with an ethically sound and critical approach
- present research and results in a clear and comprehensive written thesis
- communicate knowledge clearly and precisely, orally and in writing
- express own and others reflections and solutions in the chosen area of research

General competence

The student has

- obtained a relation to scientific literature and methods
- developed academic curiosity
- gained consciousness towards values such as openness, precision and ability to discriminate between knowledge and opinions
- the ability to think critically about central ethical, philosophical and scientific problems in his/her field

Content

The content will largely depend on the selected master project.

Forms of teaching and learning

Supervised self-study.

Workload

Approx. 1200 hours.

Coursework requirements - conditions for taking the exam

The student must :

- attend a mandatory seminar in academic writing and a library course.
- deliver a written report (individually or in groups of two people) which includes a thorough literature study, a clearly defined problem statement, research questions, and a detailed project plan.
- present their project mid-way for supervisors and students within the same specialisation.

Coursework requirements must be accepted to qualify for the exam.

Examination

Master thesis and oral exam

The assessment is based on the master thesis (individually or in groups of two people) and an individual oral exam. At the oral examination the candidate first holds a presentation of the master thesis (approx. 30 min). Then the candidate defends his/her thesis (approx 30 min). The oral examination is public.

The master thesis must be passed before the oral exam can be carried out

The thesis is graded on the A - F grading scale. It is given a tentative grade of the thesis. This grade can be adjusted up to 2 stages at the oral exam.

If the student decides to challenge the assessment, the thesis must be re-assessed. If the new assessment affects the tentative grading of the thesis, a new oral exam will be arranged.

Plagiarism control/cheating

Master's theses are subject to electronic plagiarism control. Exam papers that are partly or entirely identical will not be approved and will be regarded as cheating. For further information please see Exam regulations for Østfold University College.

Examiners

External and internal examiner

Conditions for resit/rescheduled exams

Upon re-examination, both parts of the examination must be retaken.

If the student wishes to improve a passed result of his/her master thesis, the thesis must be rewritten with a new problem statement. In this case, the student is not entitled to receive new academic supervision.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

Individually chosen, depending on the thesis.

Last updated from FS (Common Student System) July 17, 2024 12:15:27 AM

ITI54020 Master's Thesis (Autumn 2023–Spring 2025)

Facts about the course

ECTS Credits:
60

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Jan Høiberg

Teaching language:
English

Duration:
2 years

Table of contents

The course is connected to the following study programs

Elective course in the master programme in Applied Computer Science, part-time.

Absolute requirements

The student must have passed at least 50 ECTS before starting this course (students who started in 2020 or earlier must have passed at least 45 ECTS).

Lecture Semester

Fifth, sixth, seventh and eighth semester.

The student's learning outcomes after completing the course

Knowledge

The student has

- knowledge and expertise required for challenging jobs in research and development in the field
- knowledge about literature and methods related to the subjects that are part of the master thesis

Skills

The student is able to

- work independently with a complex problem over a longer period of time
- analyse a situation, describe a problem and plan its solution

- collect and analyse relevant information with an ethically sound and critical approach
- present research and results in a clear and comprehensive written thesis
- communicate knowledge clearly and precisely, orally and in writing
- express own and others reflections and solutions in the chosen area of research

General competence

The student has

- obtained a relation to scientific literature and methods
- developed academic curiosity
- gained consciousness towards values such as openness, precision and ability to discriminate between knowledge and opinions
- the ability to think critically about central ethical, philosophical and scientific problems in his/her field

Content

The content will largely depend on the selected master project.

Forms of teaching and learning

Supervised self-study.

Workload

Approx. 1800 hours.

Coursework requirements - conditions for taking the exam

The student must :

- attend a mandatory seminar in academic writing and a library course.
- deliver a written report (individually or in groups of two people) which includes a thorough literature study, a clearly defined problem statement, research questions, and a detailed project plan.
- present their project mid-way for supervisors and students within the same specialisation.

Coursework requirements must be accepted to qualify for the exam.

Examination

Master thesis and oral exam

The assessment is based on the master thesis (individually or in groups of two people) and an individual oral exam. At the oral examination the candidate first holds a presentation of the master thesis (approx. 30 min). Then the student defends the thesis (approx 30 min). The oral examination is public.

The master thesis mus be passed before the oral exam can be carried out.

The thesis is graded on the A - F grading scale. It is given a tentative grade of the thesis. This grade can be adjusted at the oral presentation.

If the student decides to challenge the assessment, the thesis must be re-assessed. If the new assessment affects the tentative grading of the thesis, a new oral exam will be arranged.

Plagiarism control/cheating

Master's theses are subject to electronic plagiarism control. Exam papers that are partly or entirely identical will not be approved and will be regarded as cheating. For further information please see Exam regulations for Østfold University College.

Examiners

External and internal sensor.

Conditions for resit/rescheduled exams

Upon re-examination, both parts of the examination must be retaken.

If the student wishes to improve a passed result of his/her master thesis, the thesis must be rewritten with a new problem statement. In this case, the student is not entitled to receive new academic supervision.

Course evaluation

This course is evaluated by a

- Mid-term evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee of the faculty of Computer Sciences.

Literature

The current reading list for 2023 Autumn can be found in Leganto

Last updated from FS (Common Student System) July 17, 2024 12:15:27 AM

ITI42122 Cyber Security Governance (Autumn 2024)

Facts about the course

ECTS Credits:
10

Responsible department:
Faculty of Computer Science, Engineering and Economics

Campus:
Halden

Course Leader:
Mary Luz Sanchez Gordon

Teaching language:
English

Duration:
½ year

Table of contents

The course is connected to the following study programs

Mandatory course in the master programme in applied computer science with specialisation in cyber security, full-time and part-time.

Lecture Semester

First semester (autumn) in the full-time and part-time programme.

The student's learning outcomes after completing the course

Knowledge

The student

- knows main ISO IT Security standards
- is familiar with the Information Technology Infrastructure Library (ITIL)
- is able to understand main risk scenarios and challenges
- knows how to apply Control Objectives for Information and Related Technology (COBIT) to specific information security topics/practices within an enterprise.
- is able to identify and understand security and controls across the strategic, tactical, and operational levels within an organization
- is able to understand main adversary tactics and techniques
- is able to distinguish governance and management by their types of activities and responsibilities

Skills

The student is able to

- use well-known frameworks and standards (COBIT, ITIL, ISO 27K) in real settings

- justify the need of IT Security and continuity planning issues for effective IT and guide them in practical business settings
- select the appropriate controls
- know whether business operations and information are secure and reliable
- know whether an enterprise is maintaining an effective system of internal control
- assess and articulate security risks from the board level to the code level.

General competence

The student gains experience with project work, including planning, performing and reporting in an existing research area.

Content

- Control Objectives for Information and Related Technology (COBIT)
- Information Technology Infrastructure Library (ITIL) for security management
- ISO/IEC 27001 Information Security Management Systems

Forms of teaching and learning

Teaching will be based on blended learning approaches. There will be recorded lectures of the topics of the course and in a weekly or bi-weekly basis, physical meetings will take place to mentor the development of the paper and guide students in the course.

Workload

Approx. 280 hours.

Coursework requirements - conditions for taking the exam

The student must deliver:

- up to three assignments.

- a security governance plan on a selected case study. The case study is chosen by the students and agreed with the course responsible. The plan can be developed individually or in groups of two students. All group members must contribute to the plan.

Coursework requirements must be accepted to qualify for the exam.

Examination

The exam is divided into two parts:

- An individual oral presentation of the security governance plan (50%). Duration approx. 5-10 min. Except the presentation, no supporting materials are allowed.
- An individual oral exam (50%) based on the course curriculum. Duration approx. 10-20 min. No supporting materials are allowed.

Grading scale A - F in both parts. Both parts of the exam must be passed to pass the course. The student will get an individual joint grade for the entire course.

Examiners

One external and one internal examiner, or two internal examiners will be involved in the assessment.

Conditions for resit/rescheduled exams

Upon re-examination, each part of the examination must be retaken.

Course evaluation

This course is evaluated by a:

- Final course evaluation (compulsory)

The responsible for the course compiles a report based on the feedback from the students and his/her own experience with the course. The report is discussed by the study quality committee

at the Department of Computer Science and Communication.

Literature

The current reading list for 2024 Autumn can be found in Leganto

Last updated from FS (Common Student System) July 17, 2024 12:15:35 AM