JÖNKÖPING UNIVERSITY
School of Engineering

PROGRAMME SYLLABUS
Preliminary, not confirmed

Software Engineering for AI (master), 120 credits

Software Engineering for AI (master), 120 högskolepoäng

Programme Code:	TASEF	Programmestart:	Autumn 2026
Confirmed:		Education Cycle:	Second-cycle level

Title of qualification

Degree of Master of Science (120 credits) with a major in Computer Science, specialisation in Software Engineering for AI

Teknologie Masterexamen med huvudområdet datavetenskap, inriktning Mjukvaruutveckling för AI

Programme overview

Main field of study
"Computer science (CS) is the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society." (Tucker, Allen, ACM, 2006.)

The Computing Sciences Accreditation Board—which is made up of representatives of the Association for Computing Machinery (ACM), and the IEEE Computer Society (IEEE CS)—identifies four areas that it considers crucial to the discipline of computer science: theory of computation, algorithms and data structures, programming methodology and languages, and computer elements and architecture.

In addition to these four areas, CSAB also identifies a large number of fields as being important areas of computer science, e.g., software engineering, artificial intelligence, database systems, parallel and distributed computation, computer networking and communication, operating systems, human–computer interaction, and computer graphics.

The scientific basis for computer science is logic and discrete mathematics. Consequently, mathematical deduction is an important tool. Scientific studies in computer science are usually conducted using quantitative methods from science or engineering. The most typical research approach is controlled experiments, with subsequent statistical analysis and inference; but traditional social science methods like case studies and interviews can be used where appropriate. Another approach is design science, where an IT-artifact is developed to demonstrate how concepts, theories, models and methods work in practice.

Computer science is related to computer engineering, which in turn has a strong connection to electrical engineering. Internationally, computer science is broadly defined, specifically, hardware and computer architectures are usually included, as can also be seen from CSAB's description above. In Sweden, the distinction is often made that computer engineering mainly studies the design and construction of computers and other hardware, while computer science focuses on software systems. Areas such as hardware programming, internet-of-things, embedded systems and robotics are thus shared between the two subjects.

Computer science is also related to information systems, and the border between the two subjects is not sharp. In computer science, the information technology is usually studied from a scientific or an engineering perspective, typically with a focus on the computer system itself. Information systems, on the other hand, mainly uses methods from social science while emphasizing the business context and processes around the IT systems.

Studies in computer science grants knowledge about, e.g., the theoretical underpinnings of computer science, their implementations in computer systems, methods for software development, and methods for evaluating systems and other deliverables. The student develops skills in analyzing problems, systems, and data, in designing and programming tools and systems, in evaluating technology and its uses, and in doing all of this both independently and in collaboration with others.

At JU, computer science is organized under the discipline Industrial Product Realization. Product realization is defined as including all tasks and activities needed to develop solutions to customer needs, and to realize these solutions through physical products and associated services. Research and education in this discipline can focus on specific parts of the product realization process or apply a holistic perspective. Industrial product realization is characterized by customer adaptation, standardization, flexible production, and automation. The process often includes market- and customer analyses, structured requirements elicitation and management, analyses of production processes and materials, optimization of components, systems, and logistics flows, and development of product lifecycle support services (including usage analyses supporting maintenance, product development, and product recycling). Research and education in computer science support these tasks by way of methods and techniques for scalable data analysis, systems development, and intelligent embedded digital solutions.

Background

As a general term, Artificial Intelligence (AI), is the use of intelligence by machines. In computer science, an ideal intelligent machine is one that perceives its environment and takes actions that maximize its chances of achieving some objective. Machine learning (ML), a branch of AI, aims to gain knowledge by identifying patterns within raw data and applying this knowledge to specific problems. AI is a rapidly developing research field and one of the driving forces of today's economy. AI excels at specialized tasks in domains such as medicine, finance, retail, manufacturing and content generation, whilst rapidly evolving general AI solutions using large language models and agentic approaches affect all aspects of society and industry.

IEEE defines software engineering (SE) as “The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.” Thus, SE covers much more than just programming, including requirements analysis, methods for software development, versioning, verification, testing, quality assurance, deployment, management and maintenance. AI-/ML-enabled software systems are significantly different from traditional software systems, in that system behavior will not only be defined from code, but also through dynamically changing data. Systems may exhibit emerging behaviors, which radically impacts development, operation and maintenance. Thus, developing software containing AI (intelligent and/or learning systems) is radically different from standard program development. Some important differences are that, in addition to code, both data and models need to be verified and validated, versioning extends to data, data cleaning procedures and trained models, deployment is of code, data, new models, or any combination thereof, including the possibility of training on live data, with the potential of constantly changing system behavior. In addition, development teams will consist not only of software developers and programmers, but also include data scientists, AI engineers and most likely also AI agents, requiring new forms of collaboration and refined methods for principled software development.

The Software Engineering for AI (SE4AI) master programme equips graduates with a solid theoretical foundation and the practical skills needed to develop safe, robust and ethical AI-/ML-enabled software systems. By combining lectures with seminars, lab sessions and larger projects, graduates will gain knowledge about state-of-the-art research and best practices in SE4AI. The acquired knowledge and skills are applied to practical work on real applications through the entire software development process, focused on systems containing intelligent and/or learning components.

A master's degree in software engineering for AI opens up career opportunities within organizations that build the next generation of AI-enhanced products and services, both in industry and the public sector.

Objectives

...

Post-graduation employment areas

This master programme focuses on the development of intelligent software products and services, covering all aspects of developing software that contain intelligent or learning components. Graduates will have developed the capabilities needed to work in both large corporations and smaller companies. They will be comfortable delivering major enterprise systems or specialized software components providing AI services.

Possible professional roles after graduation include AI Engineer, AI (software) architect, Data engineer, MLOps engineer, Software architect for AI systems, and AI solutions architect. Additionally, since both data science and modern AI are covered in detail, graduates will be able to take on professional roles as Data scientist or Machine learning engineer. Longer-term career opportunities include Technical product manager with AI focus or Chief AI officer. Going into research and development, graduates can pursue a PhD in computer science with the goal of becoming e.g. an AI researcher or a Software engineering research scientist.

Objectives

Common learning outcomes
After the completion of the programme, students must meet the intended learning outcomes, as described in The Higher Education Ordinance by Degree of Master, and also the intended learning outcomes, as described by JTH:
Knowledge and Understanding
1. demonstrate knowledge and understanding in the main field of study, including both broad knowledge of the field and a considerable degree of specialised knowledge in certain areas of the field as well as insight into current research and development work
2. demonstrate specialised methodological knowledge in the main field of study
Competence and skills
3. demonstrate the ability to critically and systematically integrate knowledge and analyse, assess and deal with complex phenomena, issues and situations even with limited information
4. demonstrate the ability to identify and formulate issues critically, autonomously and creatively as well as to plan and, using appropriate methods, undertake advanced tasks within predetermined time frames and so contribute to the formation of knowledge as well as the ability to evaluate this work
5. demonstrate the ability in speech and writing both nationally and internationally to clearly report and discuss his or her conclusions and the knowledge and arguments on which they are based in dialogue with different audiences
6. demonstrate the skills required for participation in research and development work or autonomous employment in some other qualified capacity
Judgement and Approach
7. demonstrate the ability to make assessments in the main field of study informed by relevant disciplinary, social and ethical issues and also to demonstrate awareness of ethical aspects of research and development work
8. demonstrate insight into the possibilities and limitations of research, its role in society and the responsibility of the individual for how it is used
9. demonstrate the ability to identify the personal need for further knowledge and take responsibility for his or her ongoing learning
JTH. prove ability to embrace interdisciplinary approaches

Programme-specific learning outcomes
Upon completion of the program, the intended learning outcomes provided for programme must also be met.
Knowledge and Understanding
10. display knowledge and understanding of the fundamental principles of software engineering and how they apply to AI/ML systems, from requirements engineering to deployment
11. display knowledge and understanding of the challenges and complexities of managing AI/ML systems in production environments, including aspects such as versioning, pipelines, scalability, continuous training and integration, and performance monitoring and optimization
12. demonstrate detailed knowledge of the basic algorithms and methods in machine learning and AI
13. display knowledge and understanding of state-of-the-art techniques in AI and ML, such as Deep Learning and Large Language Models
Competence and skills
14. demonstrate the ability to apply best practices and state-of-the-art software engineering methods to develop and deploy AI-enabled systems
15. demonstrate the ability to design and manage continuous training pipelines to manage model, data, and code lifecycles
16. demonstrate the ability to select appropriate AI/ML components and incorporate them effectively into a well-structured and maintainable software architecture
17. demonstrate the ability to design, execute and evaluate a data science project
Judgement and Approach
18. demonstrate the ability to evaluate and select appropriate software engineering practices for different stages of the AI/ML lifecycle
19. demonstrate the ability to critically analyze the practical implications of AI-enabled systems in design and deployment, such as performance, scalability, user experience, and reproducibility
20. demonstrate the ability to critically reflect on the principles of responsible AI and the importance of fair, accountable, traceable and ethical AI systems

Contents

Programme principles
A key principle for the programme is the treatment of software engineering for AI as both an academic discipline and a quickly evolving field where research and practice are intertwined. The programme builds upon theoretical knowledge and best practices in software engineering and AI to give graduates the necessary knowledge and skills to develop safe, robust and efficient AI-enabled software systems.

Collaboration with industry and close connection to on-going research at JU ensures that the programme reflects real-world software engineering needs and practices for AI/ML systems, with a solid theoretical foundation. Graduates will bring cutting-edge knowledge and methods to the industry and be well equipped to integrate new technology into their work and research practices.

Research basis
Computer science, including AI and software engineering, is a major area of research within the School of Engineering. It is underpinned by the Knowledge-Intensive Product Realization research environment (SPARK), in particular the AI area-of-strength. The Department of Computing forms the core of the Jönköping AI Lab research group. The main research directions in AI at JTH are AI technologies, Applied AI, AI transformation and Software Engineering for AI-enabled systems. The research environment has a strong network of academic and industrial partners, regionally, nationally and internationally. A high proportion of research is externally funded and often co-produced with industry.

Equal terms, gender equality and diversity
The School of Engineering (JTH) strives in all its activities to ensure that all individuals are given equal opportunities and treated equally. At both the JU and JTH levels, this is reflected in governing documents concerning organizational and personnel matters, the establishment and delivery of programmes and courses, as well as the monitoring of educational quality. At JTH, student influence is also ensured through student representation in various educational and industry councils.

The programme manager adds programme specific text here, which relates to the title above. (see example)

Study abroad
JTH has internationalization as a focus area where the educational programmes include opportunities for both international experiences at home as well as various opportunities to do internships and study abroad, giving students valuable experiences and skills to prepare them for a global labour market.

Semester 3 of the programme is intended as an exchange semester. The courses in the semester are substituted with similar courses at a receiving university. The student needs to assemble the replacement courses for approval by JU. The replacement package of courses must be equivalent to at least one semester of full-time work. A margin should be added should any courses be canceled at the receiving university. (see example from International Office)

Programme progression

Year 1
The first period of year 1 introduces students to developing AI solutions and the necessary mathematical skills for the coming courses. *Python Programming for AI* provides an introduction to programming AI solutions, focusing on classical machine learning and data science applications, but also using pre-trained (language) models. Theory is combined with practical programming exercises to develop a strong foundation in AI development. In parallel, the *Mathematics for Intelligent Systems* course contains elements from various fields of mathematics and mathematical statistics used when intelligent systems and machine learning are developed, used and analyzed.

The remaining year 1 courses are organized in two main tracks, one focusing on core AI content and one on software development for AI-enabled systems. These tracks run in parallel, starting in study period 2.

The core AI and data science track consists of courses shared with the AI Engineering programme: Data Science, Machine Learning, and Deep Learning. *Data Science* covers the use of scientific methods, processes, algorithms and systems to extract knowledge and insights from data in various forms, both structured and unstructured, through data analysis. The *Machine Learning* course covers standard techniques and algorithms from the field, as a scientific research and engineering discipline, providing students with a thorough understanding of the discipline and its applications. *Deep Learning* covers basic and state-of-the-art algorithms for training various deep neural network architectures, alternating theory with practice.

The SE4AI track in year 1 consists of the courses Developing AI-enabled systems, AI Systems in Production and Research Methods in SE4AI. The *Developing AI-enabled systems* course introduces students to the principles and practices of software engineering with a focus on AI/ML-enabled systems, covering the lifecycle of AI systems—from data collection and model development to continuous integration. *AI Systems* *in Production* explores the challenges and solutions involved in deploying, managing, and maintaining AI/ML-enabled systems in production environments, covering architectural design, deployment pipelines, data versioning, configuration management, and monitoring strategies to ensure scalability, reliability, and performance of AI systems. *Research Methods in SE4AI* introduces research methodologies and practices in the context of software engineering, with a focus on AI/ML-enabled systems. Students will learn to design, conduct, and evaluate empirical studies while exploring qualitative, quantitative, and mixed-method approaches. The course also covers techniques for systematic literature reviews, data analysis, and academic writing.

Year 2
During the first semester of year 2, students either study abroad or continue their specialization within SE4AI, with four courses providing in-depth theory and practical skills within the subject area. The *Advanced Software Engineering for AI* course builds on the two previous courses Developing AI-Enabled Systems and AI Systems in Production. The course explores advanced concepts in software engineering for AI/ML systems, focusing on quality, security, and reliability in production, giving in-depth knowledge and skills in requirements engineering, QA processes, continuous training, and advanced testing techniques for AI models. The course *Developing LLM-Enabled Systems* focuses on practical aspects of developing AI systems enabled by large language models (LLMs). Students explore both theoretical foundations and hands-on development techniques, gaining experience in integrating LLMs into various applications, including agentic AI systems capable of autonomous task completion.

State-of-the-Art in SE4AI covers a selection of state-of-the-art research and practice, including emergent trends, of software engineering, focusing on AI- and ML-enabled systems. The course also discusses areas in need of further research and gives students the opportunity to evaluate and criticize a subset of the research topics covered.In parallel with this, students can choose between the courses *SE4AI Project* and *Industrial Placement Course*. The elective *SE4AI Project* course is a project-based course in SE4AI, where students work in teams to develop and evaluate a system integrating intelligent and/or learning components. The focus is on practical development, system integration, and rigorous evaluation. The elective *Industrial Placement Course* (NFK) offers students who have not previously taken such a course the possibility to gain workplace experience within a collaborating organization.

The fourth and final semester in the programme further develops the students' analytical skills and ability to perform independent work in a research study. In the course *Final Project Work in Computer Science*, students spend the semester writing a 30 credit master thesis, specializing in software engineering for AI.

Courses

Course changes can occur, as long as they do not substantially affect the programme's content and learning goals.

Mandatory courses

Semester	Course Name	Credits	Main field of study	Specialised in	Course Code
1	Python Programming for AI	7.5	Computer Science	A1N	T2PFAF
1	Developing AI-Enabled Systems	7.5	Computer Science	A1F	T2UAAT
1	Data Science	7.5	Computer Science	A1N	TDSR22
1	Mathematics for Intelligent Systems	7.5		A1N	TMAR21
2	AI Systems in Production	7.5	Computer Science	A1F	T2AIPN
2	Research Methods in SE4AI	7.5	Computer Science	A1F	T2FIOT
2	Deep Learning	7.5	Computer Science	A1F	TDIS22
2	Machine Learning	7.5	Computer Science	A1F	TMLS22
3	Advanced Software Engineering for AI	7.5	Computer Science	A1F	T2AMFA
3	State-of-the-art in Software Engineering for AI	7.5	Computer Science	A1F	T2HEAS
3	Developing LLM-Enabled Systems	7.5	Computer Science	A1F	T2UALS
4	Final Project Work in Computer Science	30	Computer Science	A2E	TEXV23

Elective courses

Semester	Course Name	Credits	Main field of study	Specialised in	Course Code

3	SE4AI Project	7.5	Computer Science	A1F	T2SPIL
3	Industrial Placement Course in Computer Science	7.5	Computer Science	A1F	TNDS22

Teaching and examination

The academic year is divided into two semesters, and the semesters into two study periods. In each study period two courses are generally taken in parallel. Assessment is part of each course or module. Modes of assessment and grades are shown in each course syllabus.

Entry requirements

The applicant must hold a minimum of a bachelor's degree (i.e. the equivalent of 180 ECTS credits at an accredited university) with at least 90 credits in the main field of computer engineering, computer science, informatics, information systems, information technology, or equivalent. The bachelor’s degree should comprise a minimum of 15 credits in mathematics and at least 30 credits in programming/software development. Proof of English proficiency is required.

Continuation Requirements

In order to begin the second year, at least 37,5 credits from the programme's first year must be completed.

Qualification Requirements

To obtain a Degree of Master of Science (120 credits) with a major in Computer Science, specialisation in Software Engineering for AI, students must complete a minimum of 120 credits in accordance with the current programme syllabus, at least 60 of which must be in the main field of study Computer Science and 21 credits in Mathematics.
In addition a Degree of Bachelor of Science in Engineering/Degree of Bachelor of Science or an equivalent Swedish or foreign qualification is required.

Quality Development

At JTH, systematic quality assurance is carried out within JU's established quality system. This system, based on the requirements of the Higher Education Act, the Higher Education Ordinance, and the Standards and Guidelines for Quality Assurance in the European Higher Education Area, has been reviewed and approved by the Swedish Higher Education Authority. Active and continuous course evaluation, including student feedback through course surveys, forms one of the cornerstones of this system. Annual programme evaluations and student representation in JTH's various educational and industry councils are two additional examples.

Other Information

Admission is under ‘Admission regulations for first- and second cycle courses and study programmes at Jönköping University (Admission regulations)’.

This syllabus is based on ‘Regulations and guidelines for first-, second- and third-cycle education at Jönköping University’.