



Curriculum for the master's programme in Information Technology (IT Design and Application Development)

Aalborg University

September 2017

Preface:

Pursuant to Act 261 of March 15, 2015 on Universities (the University Act) with subsequent changes, the following curriculum for the Master's programme in IT Design and Application Development is stipulated. The programme also follows the Joint Programme Regulations and the Examination Policies and Procedures for The Technical Faculty of IT and Design

AAU, oktober 2016

Uffe Kjærulff

Head of School of Information and Communication Technology

Contents

Chapter 1: Legal Basis of the Curriculum, etc.	2
1.1 Basis in ministerial orders.....	2
1.2 Faculty affiliation.....	2
1.3 Board of Studies affiliation.....	2
1.4 Body of external examiners	2
Chapter 2: Admission, Degree Designation, Programme Duration and Competence Profile	2
2.1 Admission.....	2
2.2 Degree designation in Danish and English.....	2
2.3 The programme's specification in ECTS credits.....	2
2.4 Competence profile on the diploma.....	2
2.5 Competence profile of the programme:	3
Chapter 3: Content and Organization of the Programme	4
1 st Semester (iDA7)	5
2 nd Semester (iDA8)	11
3 rd Semester (iDA9).....	15
4 th Semester (iDA10).....	18
Chapter 4: Entry into Force, Interim Provisions and Revision	19
Chapter 5: Other Provisions	19
5.1 Rules concerning written work, including the Master's thesis.....	19
5.2 Rules concerning credit transfer (<i>merit</i>), including the possibility for choice of modules that are part of another programme at a university in Denmark or abroad..	19
5.3 Rules for examinations	20
5.4 Exemption	20
5.5 Completion of the Master's programme.....	20
5.6 Rules and requirements concerning the reading of texts in foreign languages and a statement of the foreign language knowledge this assumes	20
5.7 Additional information	20
5.8 Curriculum changes.....	20

Chapter 1: Legal Basis of the Curriculum, etc.

1.1 Basis in ministerial orders

The Master's programme in IT Design and Application Development is organized in accordance with the Ministry of Higher Education and Science's Ministerial Order no. 1328 of November 15, 2016 on Bachelor's and Master's Programmes at Universities (the Ministerial Order of the Study Programmes) and Ministerial Order no. 1062 of June 30, 2016 on University Examinations (the Examination Order) with subsequent changes. Further reference is made to Ministerial Order no. 258 of March 18, 2015 (the Admission Order) and Ministerial Order no. 114 of February 13, 2015 (the Grading Scale Order) with subsequent changes.

1.2 Faculty affiliation

The Master's programme falls under The Technical Faculty of IT and Design, Aalborg University.

1.3 Board of Studies affiliation

The Master's program falls under the Board of Studies for Computer Science.

1.4 Body of external examiners

Body of external examiners for Computer Science

Chapter 2: Admission, Degree Designation, Programme Duration and Competence Profile

2.1 Admission

Applicants without legal claim to admission:

- Bachelor of Science (BSc) in Techno-Anthropology
- Bachelor of Science (BSc) in Economics and Business Administration
- Bachelor of Science (BSc) in Sociology

2.2 Degree designation in Danish and English

The Master's program entitles the graduate to the designation cand.it i it-design og applikationsudvikling (candidates/candidate informationis technologiae). The English designation is: Master of Science (MSc) in Information Technology (IT Design and Application Development).

2.3 The programme's specification in ECTS credits

The Master's programme is a 2-year, research-based, full-time study programme. The programme is set to 120 ECTS credits.

2.4 Competence profile on the diploma

The following competence profile will appear on the diploma:

A Candidatus graduate has the following competency profile:

A Candidatus graduate has competencies that have been acquired via a course of study that has taken place in a research environment.

A Candidatus graduate is qualified for employment on the labour market on the basis of his or her academic discipline as well as for further research (PhD programmes). A Candidatus graduate has, compared to a Bachelor, developed his or her academic knowledge and independence so as to be able to apply scientific theory and method on an independent basis within both an academic and a professional context.

2.5 Competence profile of the programme:

After completion of the Master's programme, the student will have acquired the following qualifications in the subject areas that are included in the education:

Information Systems

Knowledge:

- Has knowledge about research approaches in IT application design and software development
- Can understand, and on a scientific basis, reflect over knowledge in information systems and identify relevant research problems
- Has knowledge about key topics in system development and human-computer interaction based on the highest international research in these areas

Skills:

- Excels in the scientific methods and tools of the information systems discipline
- Can design software systems for complex application areas and advance new innovative solutions
- Can evaluate the qualities of software systems in relation to their use in human and social activities
- Can describe and analyse experiences from software development practice
- Is proficient in communicating research-based knowledge and discussing professional and scientific problems with both peers and non-specialists

Database and Programming

Knowledge:

- Has knowledge about imperative programming
- Has knowledge about database modelling, database queries and database management

Skills:

- Can apply programming methods and tools in a software development process
- Can implement the design of a software system by using current programming technologies

Competences

A Master in IT Design and Application Development can:

- Independently initiate and implement professional activities in information systems, interdisciplinary cooperation and assume professional responsibility
- Manage work and situations in software development that are complex, uncertain and require new solutions
- Independently take responsibility for own professional development and specialization

Chapter 3: Content and Organization of the Programme

The programme is structured in modules and organized as a problem-based study. A module is a program element or a group of program elements, which aims to give students a set of professional skills within a fixed time frame specified in ECTS credits, and concluding with one or more examinations within specific exam periods. Examinations are defined in the curriculum.

The program is based on a combination of academic, problem-oriented and interdisciplinary approaches and organized based on the following work and evaluation methods that combine skills and reflection:

- lectures
- classroom instruction
- project work
- workshops
- exercises (individually and in groups)
- teacher feedback
- reflection
- portfolio work

Overview of the program:

All modules are assessed through individual grading according to the 7-point scale or Pass/Fail. All modules are assessed by external examination (external grading) or internal examination (internal grading or by assessment by the supervisor only).

Semester	Module	ECTS	Assessment	Exam
1 st (iDA7)	Development of a Software Application	15	7-point scale	Internal
	Systems Development	5	7-point scale	Internal
	Design and Evaluation of User Interfaces	5	7-point scale	Internal
	Introduction to Programming	5	Pass/Fail	Internal
2 nd (iDA8)	User-Centered Software Development (elective)	15	7-point scale	External
	Innovative Software Development (elective)	15	7-point scale	External
	Software Engineering	5	7-point scale	External
	Software Innovation	5	7-point scale	Internal
	Database Development	5	Pass/Fail	Internal
3 rd (iDA9)	Empirical Research in Software Development	20	7-point scale	External
	Specialisation Course in Human-Computer Interaction (elective)	5	7-point scale	External
	Specialisation Course in Systems Development (elective)	5	7-point scale	External
	Entrepreneurship	5	Pass/Fail	Internal
4 th (iDA10)	Master's Thesis	30	7-point scale	External
Total		120		

Problem Based Learning:

A compulsory course in Problem Based Learning (PBL) is offered as an integrated part of the project module to students not acquainted with PBL at Aalborg University.

Descriptions of modules

1st Semester (iDA7)

Title:	Development of a Software Application (Udvikling af en softwareapplikation)
Scope:	15 ECTS (Project)
Prerequisites:	Course modules on iDA7
Purpose:	The student should gain knowledge about issues and fundamental techniques for developing applications to solve realistic tasks, and gain experience in developing large systems, labour and quality control including testing.
Reason:	The project module focuses on achieving skills with specific and predefined methods, languages and tools. The project module should ensure that students gain a common foundation on program development that can be utilized in the following semesters
Objectives:	After completing the project module, students should be able to <u>Knowledge:</u> <ul style="list-style-type: none">• analyse and model the requirements of the object oriented paradigm• structure an application in a multi-layer architecture using current program designs, realize and test the application in an object-oriented paradigm• understand and use concepts and features in the paradigm and on this basis construct an application of high internal and external quality <u>Skills:</u> <ul style="list-style-type: none">• implement systematic testing of the application and demonstrate that the application corresponds to the intentions and needs of users• implement systematic evaluation of the user interface• argue for the choices made in all the development process activities, including explaining requirements, architecture and how users are linked <u>Competencies:</u> <ul style="list-style-type: none">• develop a running application that solves the user's problem• describe and reflect on the methods used in the development project
Teaching form:	Project work, including: <ul style="list-style-type: none">• formulation, analysis and contribution to the resolution of a current research problem within the theme of the project module• As an integrated part of the project work, the student must follow the Problem based learning and project management workshop (1 ECTS). Approved participation is required to register for the project exam, See enclosure 1.
Exam form:	Oral exam based on project report
Assessment:	Internal assessment, 7-point scale.
Evaluation criteria:	As indicated in the Faculty evaluation criteria

**Titel: Problem Based Learning (PBL) at Aalborg University
Problembaseret læring på Aalborg Universitet**

Prerequisites: None, but the course is compulsory for students not acquainted with the PBL model at Aalborg University

Objectives:
After completion of the course the student should

Knowledge:

- know how to describe in own words some of the fundamental principles of Problem Based Learning (PBL) as implemented in the Aalborg PBL model at the Faculty of Engineering and Science
- know how to identify similarities and differences between the Aalborg PBL study environment and previous study environments, incl. strengths and weaknesses in both environments

Skills:

- be able to structure project management activities based on a well-formulated problem formulation
- be able to assess project documentation based on scientific codes of conduct

Competences :

- to plan for effective collaborative learning in an intercultural environment and manage group conflicts
- be able to reflect on, plan and manage a study project in a PBL learning environment

Type of Instructions:

Lectures, discussions and group work. The course will take place during three Wednesday afternoons.

Assessment:

Internal assessment during the course/class participation according to the rules in the Examination Policies and Procedures, Addendum to the Framework Provision of the Faculty of Engineering and Science, Aalborg University. In this case the assessment is primarily based on the oral performance during the course. This means that the student has to be active during the course time and participate in discussions. The course is an integrated part of the project and a precondition for participation in the project examination for those who are not acquainted with the Aalborg PBL model. Consequently, no diploma will be issued for the course nor will it appear on the academic transcripts. The assessment is pass/fail.

Evaluation criteria:

As stated in the framework provisions, it is a precondition for students who have not studied the PBL model at Aalborg University that they have passed the course in Problem Based Learning (PBL) at Aalborg University prior to the project examination

Title: **Systems Development
(Systemudvikling)**

Scope: 5 ECTS (Course)

Prerequisites: No special prerequisites for the module.

Objective: Knowledge:
Students should achieve knowledge on the following theories and methods:

Object-oriented modelling in analysis and design:

- modelling of context (application domain and problem domain)
- object-oriented concepts: class, object, event, structure, function, use patterns, component, component architecture
- UML: class diagram, state chart diagram, sequence diagram, diagram for use patterns

Modelling with patterns:

- patterns for modelling application and problem domains
- patterns for composing components
- specifically the patterns for analysis: object-descriptor, hierarchy, stepwise-role, materials, procedure
- specifically the patterns for design: collection, layered, observer, client-server, model-view-controller

System development methods:

- waterfall method and model-driven development
- iterative method and prototype-driven development
- activities in systems development and relations between activities

Systems practices:

- techniques to determine the specific method
- the relation between methodology and practice
- strengths and weaknesses of model-driven and prototype-driven development

Skills:

The student should achieve the following skills:

- be able to explain accurately, using the concepts and modelling language of the discipline
- be able to model the requirements to a system, its context and all its various parts (model, features and interfaces)
- be able to model a system design at component level and describe relations between components.

Competencies:

The student should be able to apply concepts, patterns and modelling language to describe a specific system that solves a well-defined task

Teaching form: Course

Exam form: Oral or written exam

Assessment: Internal assessment, 7-point scale.

Evaluation criteria: As indicated in the Faculty evaluation criteria

Title: **Design and Evaluation of User Interfaces
(Design og evaluering af brugergrænseflader)**

Scope: 5 ECTS (Course)

Prerequisites: No special prerequisites for the module.

Objective: Knowledge:
Students should achieve knowledge on the following theories and methods:

Fundamentals of human-computer interaction:

- interaction design
- usability and user experience
- design principles
- interaction forms
- human cognition, perception and memory

Interaction design process:

- activities in interaction design
- user-centered design
- contextual design, participatory design
- different lifecycle models for interaction design

Use context and users:

- understand needs and requirements: e.g. interview, observation, questionnaire, probes, card sorting
- task analysis: e.g. hierarchical task analysis, objectives, tasks, actions
- scenarios and personas
- use patterns

Design of interfaces:

- visual design principles
- Gestalt laws
- sketching and prototyping
- conceptual and physical interface design

Usability evaluation:

- activities
- roles and tasks
- identification of usability problems

Skills:

The student should achieve the following skills:

- understand basic and advanced concepts and theories of human-computer interaction
- be able to explain the activities in the design of an interface accurately
- be able to explain the activities of a usability evaluation

Competencies:

The student should be able to apply concepts, techniques and methods to design and evaluate a specific system that solves a well-defined task and discuss relations between concepts, techniques and methods in the subject.

Teaching form: Course

Exam form: Oral or written exam
Assessment: Internal assessment, 7-point scale.
Evaluation criteria: As indicated in the Faculty evaluation criteria

**Title: Introduction to Programming
(Grundlæggende programmering)**

Scope: 5 ECTS (course)

Prerequisites: No special prerequisites for the module.

Objective: Students who complete the module should obtain a solid foundation in working with computers and other digital devices, which will be built upon in future coursework to enable programming for different media platforms.

Furthermore, to provide the student with a foundation and basic introduction for the systematic development of programs using object-oriented modelling and programming. The student should acquire an understanding of basic concepts and mechanisms in an object-oriented programming language such that the student is able to use the language and associated class library to implement small programs.

Students who complete the course module should obtain the following qualifications:

Knowledge:

- Understanding of flow control structures, both logical (e.g., if, case), and loop (e.g., for, while)
- Understanding data types and structures (e.g., array, struct, list)
- Understanding functions
- Understanding basic principles of Object Oriented programming, such as using application programming interfaces (APIs) and the need to create custom classes
- Basic introduction to concepts of access (e.g., public, private, protected) and inheritance, composition and encapsulation
- Understanding of design methodologies for programming and understanding of the distinction between good and bad programming practices

Skills:

- Ability to apply knowledge to the design of a simple event-driven interactive interface, e.g., a simple game
- Interpret and analyse programming code and work out manually
-
- Ability to apply programming skills to the design and implementation of simple functions and classes
- Synthesize simple built-in functions and classes from APIs
-
- Ability to plan and perform systematic test of small programs (application)
-

Competences:

- evaluate small fragments of existing code, judge its design and recommend changes
- use object-oriented programming for solving specific small programming tasks

Teaching form: Course

Exam form: Oral or written exam

Assessment: Internal assessment, pass/fail.

Evaluation criteria: As indicated in the Faculty evaluation criteria

2nd Semester (iDA8)

Title:	User-Centered Software Development (Brugercentreret softwareudvikling)
Scope:	15 ECTS (project, elective)
Prerequisites:	iDA7 project; SD; DEB; and programming skills
Objective:	Students who complete the module: <u>Knowledge:</u> <ul style="list-style-type: none">• Must have knowledge about the application of concepts, processes and theories for user-centered development and software engineering <u>Skills:</u> <ul style="list-style-type: none">• Must be able to define a problem within user-centered software development and solve it• Must be able to argue for the chosen requirements, design and implementation and how they relate• Must be able to describe and explain how a chosen process solved the defined problem <u>Competencies:</u> <ul style="list-style-type: none">• Must have experience with user-centered design• Must have experience with selected processes for software engineering• Can study, reflect on, and manage user-centered development processes
Type of instruction:	Project work.
Exam format:	External oral exam based on written project with 7-point scale.
Evaluation criteria:	As indicated in the Faculty evaluation criteria

Title:	Innovative Software Development (Innovativ softwareudvikling)
Scope:	15 ECTS (project, elective)
Prerequisites:	iDA7 project; SD; DEB; and programming skills
Objective:	Students who complete the module: <u>Knowledge:</u> <ul style="list-style-type: none">• Must have knowledge about the application of concepts, processes and theories for innovation and software engineering <u>Skills:</u> <ul style="list-style-type: none">• Must be able to define a problem within innovative software development and solve it• Must be able to argue for the chosen requirements, design and implementation and how they relate• Must be able to describe and explain how a chosen process solved the defined problem <u>Competencies:</u> <ul style="list-style-type: none">• Must have experience with creative processes for software innovation

- Must have experience with selected processes for software engineering
- Can study, reflect on, and manage innovative processes and development processes

Type of instruction: Project work.

Exam format: External oral exam based on written project with 7-point scale.

Evaluation criteria: As indicated in the Faculty evaluation criteria

Title: **Software Engineering
(Software engineering)**

Scope: 5 ECTS (Course)

Prerequisites: Course modules Systems Development and Design and Evaluation of User Interfaces.

Objectives: Knowledge:
The student should gain knowledge of leading paradigms (e.g. traditional and agile) in professional development of software. The student should also gain knowledge of theories, methods and techniques involved in these paradigms (e.g. process modelling, management of requirements, design, project management, testing, process improvement) as well as an overview of theory of science for software engineering.

Skills:

The student should achieve the following skills:

- the ability to explain course concepts precisely using the terminology of the discipline, and be able to distinguish between and compare the software engineering paradigms
- Be able to explain accurately and using the terminology of the discipline, the theories, methods and techniques of software engineering paradigms and their application in the professional development of software intensive systems

Competencies:

The student should be able to select, justify and use appropriate paradigms, theories, methods and techniques in their own development contexts.

Teaching form: Course

Exam form: Oral exam

Assessment: External assessment, 7-point scale.

Evaluation criteria: As indicated in the Faculty evaluation criteria

Title:	Software Innovation (Softwareinnovation)
Scope:	5 ECTS (course)
Prerequisites:	Thorough understanding of computer science principles that were presented in the previous semester.
Objectives:	In this context, software innovation implies the wide definition of innovation related to software development. Emphasis is on innovation, products and processes but also leadership of innovative work and personal and organizational prerequisites for innovation are included in the course.

Knowledge:

The student will acquire knowledge on the following theories and methods:

Software innovation theory:

- central theories about innovation and innovation processes
- personal and organizational conditions for innovation
- theories of software innovation

Innovation methods:

- general methods and techniques to support innovation
- specific methods and techniques for software innovation

Innovation practice:

- experience with methods and techniques in creative and innovative processes
- assessment of strengths and weaknesses of the methods and techniques for creative and innovative processes for software development

Skills:

The student must achieve the following skills:

- able to explain theories accurately using professional concepts
- able to explain approaches to selection and management of innovative processes in software development
- able to discuss personal and organizational prerequisites for software innovation
- use own experience to explain and discuss tools and techniques supporting software innovation

Competencies:

The student should be able to assess the innovative potential of a software product or software-supported process.

Teaching form:	Course
Exam form:	Oral or written exam
Assessment:	Internal assessment according to 7-point scale
Evaluation criteria:	As indicated in the Faculty evaluation criteria

Title: **Database Development
(Databaseudvikling)**

Scope: 5 ECTS (Course)

Prerequisites: Basic knowledge of programming.

Objectives: Knowledge:

The student should gain knowledge about:

- the relational data model and its concepts
- data modelling (ERD / UML)
- concept of operations
- integrity constraints including primary keys, promoting keys, checks and not null
- SQL language for defining databases, basic and advanced data extraction and modification of data
- Extracting information from a DBMS from a programming language such as PHP, Java or C #
- "best practice" for good design and use of DBNS and SQL
- Understand and use the advanced queries using more than two tables, e.g., for inner join, outer join, and the set operators

Skills:

The student should achieve the following skills:

- be able to construct and evaluate a database design and database scheme
- demonstrate understanding of the relational data model and how to evaluate the model
- construct and evaluate complex queries in SQL and other relevant query languages
- constructing transactions that comply with relevant technical and commercial criteria
- Informally argue for the goodness/quality of the database design using knowledge on unnecessary repetition of information plus first and third normal form

Competencies:

The student should achieve the following competencies:

- use a database management system (DBMS) to store and retrieve information
- use SQL from a conventional programming language

Teaching form: Course

Exam form: Oral or written exam

Assessment: Internal assessment, pass/fail.

Evaluation criteria: As indicated in the Faculty evaluation criteria

3rd Semester (iDA9)

Title: **Empirical Research in Software Development
(Empirisk forskning i softwareudvikling)**

Scope: 20 ECTS

Prerequisites: iDA8 project. The course module Software Development Research Methods must be followed in parallel or before.

Objective: Students who complete the module:

Knowledge:

- Must have knowledge about the application of concepts, processes and theories from the course Software Development Research Methods

Skills:

- Must be able to identify, formulate and analyse a problem in context.
- Must be able to relate the problem definition to empirical research processes for software development.
- Must be able to identify and describe relevant empirical research processes to address the defined problem.
- Must be able to report on the findings of the empirical research processes

Competencies:

- Must document experience with empirical research processes to address knowledge creation on software development or software products.

Type of instruction: Project work and workshops.

Exam format: External oral exam based on written project with 7-point scale.

Evaluation criteria: As indicated in the Faculty evaluation criteria

Title: **Specialisation Course in Human-Computer Interaction
(Specialiseringskursus i menneske-maskine interaktion)**

Scope: 5 ECTS (elective)

Prerequisites: iDA7 and iDA8 or the like

Objective:

Knowledge:

The student should achieve in-depth insight into key issues in contemporary research in human-computer interaction.

Skills:

Based on a scientific article in the course's central themes, the student should be able to:

- give a clear and understandable presentation of the article's key elements, including its premises, issue(s), theory, methods, results and conclusions
- explain relevant theories, methods and arguments presented in the article

Competences:

Based on a scientific article in the course's central themes, the student should be able to:

- relate the theories, methods and results presented in the article to the course topics
- assess the proposed solutions, results and/or conclusions of the article as well as assess their qualities and practicality and put them into perspective.

Teaching form: Course

Exam form: The student gives a lecture of 30 minutes on a defined scientific subject area (typically in the form of an article) in relation to issues addressed in the course. The selection of subject area and the framing of the task to each student are made by the course lecturer, usually in consultation with the student's project supervisor. The student is given 7 days of preparation. After the lecture, the examiner and censor can ask questions related to the student's presentation of the theme. This does not normally exceed 10 minutes.

Assessment: External assessment, 7-point-scale

Evaluation criteria: As indicated in the Faculty evaluation criteria

Title: Specialisation Course in Systems Development (Specialiseringskursus i systemudvikling)

Scope: 5 ECTS (elective)

Prerequisites: iDA7 and iDA8 or the like

Objectives: Knowledge:

The student should achieve in-depth insight into key issues in contemporary research in systems development.

Skills:

Based on a scientific article in the course's central themes, the student should be able to:

- give a clear and understandable presentation of the article's key elements, including its premises, issue(s), theory, methods, results and conclusions
- explain relevant theories, methods and arguments presented in the article

Competences:

Based on a scientific article in the course's central themes, the student should be able to:

- relate the theories, methods and results presented in the article to the course topics
- assess the proposed solutions, results and/or conclusions of the article as well as assess their qualities and practicality and put them into perspective.

Teaching form: Course

Exam form: The student gives a lecture of 30 minutes on a defined scientific subject area (typically in the form of an article) in relation to issues addressed in the course. The selection of subject area and the framing of the task to each student are made by the course lecturer, usually in consultation with the student's project supervisor. The student is given 7 days of preparation. After the lecture, the

examiner and censor can ask questions related to the student's presentation of the theme. This does not normally exceed 10 minutes.

Assessment: External assessment, 7-point-scale

Evaluation criteria: As indicated in the Faculty evaluation criteria

**Title: Entrepreneurship
(Entreprenørskab)**

Scope: 5 ECTS (course)

Prerequisites: Academic maturity corresponding to the level of bachelor in a software-related discipline.

Objectives: Knowledge:
The student should achieve knowledge about entrepreneurship and business development related to software (information and communication technologies) including typically:

- different scientific approaches to entrepreneurship, including effectuation
- intra-/entrepreneurship
- competition and market conditions
- business models and business plans
- intellectual property rights
- market development and marketing
- growth strategies
- open entrepreneurship

Skills:
The student should achieve the following skills:

- the ability to explain course concepts precisely using the professional terminology of the discipline
- the ability to use those concepts to explain practical and empirical (case based) contexts

Competencies:
The student should be able to formulate, develop and present their own software-related business ideas to a qualified audience.

Teaching form: Course

Exam form: Oral or written exam

Assessment: Internal assessment, pass/fail

Evaluation criteria: As indicated in the Faculty evaluation criteria

4th Semester (iDA10)

Title: Master's Thesis
(Kandidatspeciale)

Scope: 30 ECTS

Prerequisites: iDA9 project.

Objective: Students who complete the module:

Knowledge

- Must have knowledge about the application of concepts, processes and theories of software development

Skills

- Must be able to identify, formulate and analyse a problem in context.
- Must be able to relate the problem definition to empirical research processes for software development and argue for the relevance the problem in a wider context.
- Must be able to identify, explain and argue for the relevance and rigour of the chosen empirical research processes to address the defined problem.
- Must be able to report on the findings of the empirical research processes and explain the contributions to research and practice.
- Must be able to perform a literature review relevant to the defined problem.

Competencies

- Must document experience with empirical research processes to address knowledge creation on software development or software products.
- Must have experience with research processes and research setting.

Type of instruction: Project work.

Exam format: External oral exam based on written thesis with 7-point scale.

Evaluation criteria: As indicated in the Faculty evaluation criteria

Chapter 4: Entry into Force, Interim Provisions and Revision

The curriculum is approved by the Dean of The Technical Faculty of IT and Design and enters into force as of September , 2017.

Students who wish to complete their studies under the previous curriculum from 2012 must conclude their education by the summer examination period 2017 at the latest, since examinations under the previous curriculum are not offered after this time.

Chapter 5: Other Provisions

5.1 Rules concerning written work, including the Master's thesis

In the assessment of all written work, regardless of the language it is written in, weight is also given to the student's spelling and formulation ability, in addition to the academic content. Orthographic and grammatical correctness as well as stylistic proficiency are taken as a basis for the evaluation of language performance. Language performance must always be included as an independent dimension of the total evaluation. However, no examination can be assessed as 'Pass' on the basis of language performance alone; similarly, an examination normally cannot be assessed as 'Fail' on the basis of poor language performance alone.

The Board of Studies can grant exemption from this in special cases (e.g., dyslexia or a native language other than Danish).

The Master's thesis is written in English and must include a Danish summary. The summary must be at least 1 page and not more than 2 pages. The summary is included in the evaluation of the project as a whole.

5.2 Rules concerning credit transfer (*merit*), including the possibility for choice of modules that are part of another programme at a university in Denmark or abroad

In the individual case, the Board of Studies can approve successfully completed (passed) programme elements from other Master's programmes in lieu of programme elements in this programme (credit transfer). The Board of Studies can also approve successfully completed (passed) programme elements from another Danish programme or a programme outside of Denmark at the same level in lieu of programme elements within this curriculum. Decisions on credit transfer are made by the Board of Studies based on an academic assessment. See the Joint Programme Regulations for the rules on credit transfer.

5.3 Rules for examinations

The rules for examinations are stated in the Examination Policies and Procedures published by The Technical Faculty of IT and Design on their website.

5.4 Exemption

In exceptional circumstances, the Board of Studies study can grant exemption from those parts of the curriculum that are not stipulated by law or ministerial order. Exemption regarding an examination applies to the immediate examination.

5.5 Completion of the Master's programme

The Master's programme must be completed no later than four years after it was begun.

5.6 Rules and requirements concerning the reading of texts in foreign languages and a statement of the foreign language knowledge this assumes

It is assumed that the student can read academic texts in modern English and use reference works, etc.

5.7 Additional information

The current version of the curriculum is published on the Board of Studies' website, including more detailed information about the programme, including exams.

5.8 Curriculum changes

Replacement of "Framework Provisions" with "Joint Programme Regulations". Changes in description of learning objectives.

Changes for Sept, 2017: Insertion of PBL-course for non-AAU students

5.8 Enclosures

Enclosure 1:
Title: Problem based learning and project management (Problembaseret læring og projektledelse)
Size: 1 ECTS
Prerequisites: None
Objectives: The objective is to make newly started Master students coming from institutions other than AAU prepared to enter the problem based learning environment at AAU and manage study projects in close collaboration with peers. After completion of the course the student should have acquired: <ul style="list-style-type: none">• Knowledge about AAU as a frame of study and student life in Aalborg• Knowledge to describe in own words some of the fundamental principles of Problem Based Learning (PBL) as implemented in the Aalborg PBL model at the Faculty of Engineering and Science• Knowledge to identify similarities and differences between the Aalborg PBL study environment and previous study environments, incl. strengths and weaknesses in both environments• Skills to structure project management activities based on a well-formulated problem formulation• Skills to assess project documentation based on scientific codes of conduct.• Competences to plan for effective collaborative learning in an intercultural environment and manage group conflicts• Competence to reflect on, plan and manage a study project in a PBL learning environment
Type of instruction: Three half day workshops
Exam format: The assessment is performed based on active participation in the arranged workshops.
Evaluation criteria: The criteria for the evaluation are specified in the Joint Programme Regulations.