



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DIT285 Advanced Requirements Engineering, 7.5 credits

Avancerad Kravhantering, 7,5 högskolepoäng

Second Cycle

Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2020-10-27 to be valid from 2021-08-30, autumn semester of 2021.

Field of education: Science 100%

Department: Department of Computer Science and Engineering

Position in the educational system

The course is compulsory within Software engineering and Management Master's Programme and is offered within several programmes. It is also a single subject course at the University of Gothenburg.

The course can be part of the following programmes: 1) Computer Science, Master's Programme (N2COS), 2) Applied Data Science Master's Programme (N2ADS), 3) Game Design & Technology Master's Programme (N2GDT) and 4) Software Engineering and Management Master's Programme (N2SOF)

Main field of studies

Interaction Design

Software Engineering

Specialization

A1N, Second cycle, has only first-cycle course/s as entry requirements

A1N, Second cycle, has only first-cycle course/s as entry requirements

Entry requirements

To be eligible for this course the student should have

- a bachelor degree in Software Engineering, Computer Science, Information Technology, Information Systems, or equivalent;
- a successfully completed course in programming (e.g., DIT042 Object-oriented Programming, DIT012 Imperative Programming with Basic Object-orientation,

DIT143 Functional Programming, or equivalent);

- a successfully completed project course (or bachelor thesis) in applied software development or software engineering (e.g., DIT212 Object-oriented programming project, or DIT543 Software Engineering Project).

Applicants must prove knowledge of English: English 6/English B or the equivalent level of an internationally recognized test, for example TOEFL, IELTS.

Learning outcomes

On successful completion of the course the student will be able to:

Knowledge and understanding

- Identify a common RE challenge in a given software development context.
- Choose an appropriate RE practice in a given software development context.
- Compare suitability as well as advantages and disadvantages of given RE practices in a given software development context.
- Explain the current state of practice and research in requirements engineering.

Competence and skills

- Plan suitable RE practices in a team with respect to a given software development context.
- Effectively apply a suitable RE practice in a team in a given software development context.
- Analyze the effect and quality of the outcome of a set of or individual RE practices in a given software development context.

Judgement and approach

- Assess new requirements engineering knowledge (challenge, principle, practice) and relate them to the framework in this course.
- Suggest suitable actions to overcome a lack of requirements knowledge in a software development context.
- Consider inter-team, program level and social/ethical implications of a set of RE practices in a given software development context.
- Critically assess the effectivity of a set of RE practices from the perspective of the student's master program (e.g. Software Engineering & Technology/Management, Interaction Design, Game Design, Data Science, ...)

The course is sustainability-related, which means that at least one of the learning outcomes clearly shows that the course content meets at least one of the University of Gothenburg's confirmed sustainability criteria.

Course content

Understanding requirements is key to successful software engineering: Building software that is fit for its purpose relies on understanding the exact problem that must be solved. The purpose of this course is to learn challenges, principles, and practices to identify, analyse and manage requirements from relevant sources, both at the start and during a software development project. The course regards these issues in specific development contexts, i.e. specific constellations of customers and suppliers as well as constraints related to the domain and development lifecycle that characterise developing a piece of software.

This course is on an advanced level and takes a holistic view on the state of the art of requirements engineering as part of successful software engineering, technology, and management. Students are expected to familiarize themselves with recent, relevant research in the field, to critically reflect on the implications of new findings, and to develop their abilities and expertise as software professionals.

The course teaches challenges, principles, and concrete practices related to the following subfields of requirements engineering (RE):

- Requirements analysis
- Elicitation
- Analysis
- Documentation
- Negotiation
- Verification and validation
- Requirements management
- Change management
- Traceability

Beyond a brief introduction into concrete notations of requirements, the course focuses on holistically managing requirements-related knowledge on relevant scopes, including:

- Team level: knowledge within a software development team and across several teams working in the same area.
- Program level: knowledge within a particular product team or across a set of related product teams (e.g. agile release trains)
- Enterprise level: knowledge related to managing a portfolio of products.

Sub-courses

1. **Take-home examination** (*Hemtentamen*), 4 credits
Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)
2. **Project (assignments and workshops)** (*Projekt*), 3.5 credits
Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)

Form of teaching

The course is organised as a series of lectures, workshops as well as project assignments.

Language of instruction: English

Assessment

The student is examined by individual active participation in all workshops, the completion of a group project and a written individual take-home exam.

If a student, who has failed the same examined component twice, wishes to change examiner before the next examination, a written application shall be sent to the department responsible for the course and shall be granted unless there are special reasons to the contrary (Chapter 6, Section 22 of Higher Education Ordinance).

In cases where a course has been discontinued or has undergone major changes, the student shall normally be guaranteed at least three examination occasions (including the ordinary examination) during a period of at least one year from the last time the course was given.

Grades

The grading scale comprises: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U).

To pass the course, all mandatory components must be passed. To earn a higher grade than Pass, a higher weighted average from the grades of the components is required.

Course evaluation

The course is evaluated through meeting after the course between teachers and student representatives. Further, an anonymous questionnaire is used to ensure written information. The outcome of the evaluations serves to improve the course by indicating which parts could be added, improved, changed or removed.

Additional information

The course is a joint course together with Chalmers.

Course literature to be announced the latest 8 weeks prior to the start of the course.

The course replaces the course DIT284, 7.5 credits. The course cannot be included in a degree which contains DIT284. Neither can the course be included in a degree which is based on another degree in which the course DIT284 is included.