



COMPUTER SCIENCE AND ENGINEERING

DIT544 Advanced Software Architecture, 7.5 credits

Avancerad mjukvaruarkitektur, 7,5 högskolepoäng

Second Cycle

Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2013-02-11 and was last revised on 2017-06-07 to be valid from 2017-08-20, autumn semester of 2017.

Field of education: Science 100%

Department: Computer Science and Engineering

Position in the educational system

The course is a part of Software Engineering Master's Programme and Computer Science Master's Programme. The course 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) Software Engineering Master's Programme (N2SOM) and 4) Software Engineering and Management Master's Programme (N2SOF)

Main field of studies

Software Engineering

Computer Science-Software Engineering
and Tech

Specialization

A1F, Second cycle, has second-cycle
course/s as entry requirements

A1F, Second cycle, has second-cycle
course/s as entry requirements

Entry requirements

The requirement for the course is to have completed two years of studies within the subject Computer Science or Software engineering or equivalent. Successful completion of a course in the area of software architecture (DIT945 Model Driven software development or similar) is required.

English B level or English proficiency equivalent to IELTS 6.5 no part under 5.5 or TOEFL 575 p, TWE score 4.5 is also required.

Learning outcomes

After completion of the course the student is expected to be able to

1. Knowledge and understanding

- describe the importance of predictive architecting early in the system life-cycle
- state/describe the applicability of model-based approaches
- describe the intentionality of models throughout the system life-cycle
- describe/state the characteristics and challenges of architecting system-of-systems and ultra-large-scale systems
- distinguish between software architecture, enterprise architecture, system architecture, and run-time architectures.
- explain the link between business strategy, business process and system- and software architecture
- know of available tools for supporting architecture modelling and analysis
- explain the role of architecture as a central artefact in system development

2. Skills and abilities

- describe inter-dependencies among quality-attributes
- assess an architecture quantitatively and qualitatively
- develop architectural models using ADLs,
- conduct incremental and multi-fidelity architecture-centric modeling, verification, and validation
- explain architectural models described in UML or ADL

3. Judgement and approach

- identify critical aspects of an architecture that requires, or benefits, to be modeled, verified and validated
- trade-off architectural decisions and quality attributes
- assess strengths and weaknesses of model-based approaches and methods

Course content

The course focuses on principles and methods that aid the designer/developer/architect to gain increased confidence in the architectural design. This includes architectural patterns, qualitative and quantitative assessment of architectures, quantitative modelling using architecture description languages such as AADL and MARTE, and qualitative architecture evaluation methods, e.g., ATAM. Finally, the course will also address the specific challenges related to scale, dynamics, and heterogeneity as found in system of systems, and ultra-large scale systems.

Form of teaching

The course is examined by an individual hall-exam (U-VG) and an individual project (U-VG), presented at a seminar. The student also has to participate in other student's presentation.

In addition, to pass the course the student must also produce an approved essay.

A student who failed a same examination twice has the right to request of the department a change of examiner. The request is to be in writing and submitted as soon as possible. The department is to grant such a request without undue delay.

In cases where a course has been discontinued or major changes have been made a student should be guaranteed at least three examination occasions (including the ordinary examination occasion) during a time of at least one year from the last time the course was given.

Language of instruction: English

Assessment**Grades**

The grading scale comprises: Pass with Distinction (VG), Pass (G) and Fail (U).

In order to be awarded the grade Pass for a full course, the exam and the project has to be approved. In order to get Pass with Distinction as final grade, the exam and the project has to be awarded with the grade Pass with Distinction. In both cases the essay has to be approved.

Regarding the application of ECTS scales, please see Vice-Chancellor's decision 2007-05-28, dnr G 8 1976/07.

Course evaluation

The course is evaluated through meetings both during and 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 indication which parts could be added, improved, changed or removed.

Additional information

It is recommended the student to have basic knowledge in software engineering development (such as DIT270 Software engineering using formal methods) beforehand.

The course replaces the DIT542 Advanced Software Architecting course. The course cannot be included in a degree which contains DIT542. Neither can the course be included in a degree which is based on another degree in which the course DIT542 is included.

The course corresponds to the Advanced Software Architecture course at Chalmers.