



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DIT976 Software Language Engineering for Domain-Specific Languages, 7.5 credits

Programvaruspråkstechnik för domänspecifika språk, 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 2022-01-17, spring semester of 2022.

Field of education: Science 100%

Department: Department of Computer Science and Engineering

Position in the educational system

The course is a semi-elective course in the Software Engineering and Management 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) and 3) Software Engineering and Management Master's Programme (N2SOF)

Main field of studies

Software Engineering

Specialization

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

Entry requirements

To be eligible for this course, the student must have;

1. a Bachelor degree in Software Engineering, Computer Science or equivalent,
2. successfully completed a 7.5 hec course in Object oriented Programming (such as DIT948 programming, DIT952 Object oriented Programming and Design or equivalent)

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

After completing the course the student should be able to:

Knowledge and understanding

- explain the following concepts: models, metamodel, constraints, transformation, semantics, abstract and concrete syntax
- explain the architecture of contemporary modeling frameworks
- explain how domain specific modeling languages can be realized within a contemporary modeling framework
- explain current research trends in domain-specific language (DSL) engineering

Competence and skills

- construct domain specific languages, e.g. specify meta models including syntax and semantics
- define syntactic constraints using a constraint language
- implement meta models within a modeling framework
- construct model editors within a modeling framework
- create model validators within a modeling framework
- specify model transformations, and realize them within a modeling framework
- apply the domain specific modeling approach to an authentic case

Judgement and approach

- identify emerging technologies for model-driven engineering using relevant information sources
- select appropriate domain specific modeling technology for a given tool
- propose effective strategies and concrete technologies for realizing parts of a DSL based on a concrete problem

Course content

The course consists of a series of lectures and a project. The lectures introduce each theme; the project explores the themes by development of a modeling tool. The tool is aimed at an authentic situation, and is realized by a contemporary modeling framework. The project consists of weekly iterations, during which the modeling tool is incrementally developed. Project work is carried out in teams with support of a group supervisor.

Students will be familiarized with contemporary technologies and notations for creation, adaptation, and transformation of modeling languages.

Sub-courses

1. **Written hall examination** (*Skriftlig salstentamen*), 4.5 credits
Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)
2. **Project** (*Projekt*), 3 credits
Grading scale: Pass (G) and Fail (U)

Form of teaching

The teaching of this course will consist of different forms: lectures, literature studies, group work, group supervision, and practical assignments.

Language of instruction: English

Assessment

The examination consists of two parts: a final individual written hall examination carried out in an examination hall (4.5 hec) and a project (3 hec).

The project is carried out in groups of normally 4-6 students. The project part is graded individually, taking into account the group work as well as the student's individual contribution to the group work.

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).

The grading scale comprises Fail (U), 3, 4 or 5. In order to pass the course both the project and the written hall examination have to be approved. The final grade in the course is decided from the grade of the written hall examination.

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 indicating which parts could be added, improved, changed or removed.

Additional information

The course is a joint course together with Chalmers.

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

In the Software Engineering Master's Programme, the course is part of a set of four courses, out of which the student need to take at least two.

The course replaces the course DIT975, 7.5 credits. The course cannot be included in a degree which contains DIT975. Neither can the course be included in a degree which is based on another degree in which the course DIT975 is included.