DD2489 Scalable software Development with Functional Programming 7.5 credits

Skalbar programutveckling med funktionell programmering

This is a translation of the Swedish, legally binding, course syllabus.

If the course is discontinued, students may request to be examined during the following two academic years

Establishment

The official course syllabus is valid from the autumn semester 2023 in accordance with the decision from the Dean of the school:J-2022-2165. Date of decision: 2022-09-19.

Grading scale

P, F

Education cycle

Second cycle

Main field of study

Computer Science and Engineering

Additional regulations

This course contains a group project. Course registration after the official registration period is therefore not possible, since we need to create the groups in the beginning of the course.

Specific prerequisites

Knowledge in software development technology, 6 higher education credits, equivalent to completed course
DD2480/DD1367/DD1369/DD1392/DD1393/DD1346/DD1387/DD1388/DD2387/IV1303/HI1027/H

Language of instruction

The language of instruction is specified in the course offering information in the course catalogue.

Intended learning outcomes

After passing the course, the student shall be able to

- design, using functional programming, large systems so that they become easier to test, understand and manage
- model information in a program using only immutable data
- reflect on and discuss how architecture and design can be improved
- reflect on the concepts identity, state and value
- design entities and components so that they become easy to test and write tests for them
- protect internal design of a program at integration with other systems
- develop a program in collaboration with other developers
- review and reflect on given source code,
- maintain given source code

in order to

- understand and master the parameters in software development that make source code sustainable, re-usable and flexible during changing requirements.

Course contents

The lectures consist of much live coding, practical guidance, code-review and dialogue around how we can improve the architecture and design of the program code. This requires an active participation and an open positive atmosphere during the lectures. The idea of the lectures is to enthuse and broaden the perspectives around large-scale software development.

The lectures are compulsory, since the course is based on our joint dialogue and the active participation that is included in it.

Main contents: functional programming, code quality, readability, maintainability, cooperation, version management (git), global state, dependencies, mathematical functions, persistent data structures, handling of the state of applications, reactive programming, web devel-

opment, testability, test as documentation, comparison with object-oriented programming, atomic updating of state, concurrency, DSL, LISP and REPL.

The programming languages Clojure and ClojureScript are used to realize the course content in a project. Thereby, the course will also give solid knowledge in LISP Clojure, Clojure Special and REPL workflow.

Every week, the students will be shown a video with a conference presentation. The idea is that it should give inspiration and open new doors. During the lectures, we will reflect on the contents.

Examination

• PRO1 - Software Engineering Project, 7.5 credits, grading scale: P, F

Based on recommendation from KTH's coordinator for disabilities, the examiner will decide how to adapt an examination for students with documented disability.

The examiner may apply another examination format when re-examining individual students.

Ethical approach

• All members of a group are responsible for the group's work.

• In any assessment, every student shall honestly disclose any help received and sources used.

• In an oral assessment, every student shall be able to present and answer questions about the entire assignment and solution.