UNIVERSITY OF GOTHENBURG

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DIT355 Mini Project: Distributed Systems Development, 7.5 credits

Miniprojekt: Distribuerad Systemutveckling, 7,5 högskolepoäng

First Cycle

Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2018-01-22 to be valid from 2018-08-19, autumn semester of 2018.

Field of education: Science 100%
Department: Department of Computer Science and Engineering

Position in the educational system

The course is compulsory within the Software Engineering and Management Bachelor's Programme.

The course can be part of the following programme: 1) Software Engineering and Management Bachelor's Programme (N1SOF)

Main field of studies	*Specialization*
Software Engineering	G1F, First Cycle, has less than 60 credits in first-cycle course/s as entry requirements

Entry requirements

To be eligible for this course, students must have successfully completed the following courses, or equivalent:

- DIT344 Fundamentals of Software Architecture, 7.5 credits,
- DIT341 Mobile and Web Development, 7.5 credits,
- DIT092 Mini Project: Team Programming, 7.5 credits.

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

- define the role of Software Architecture in distributed systems projects,
- explain the importance of planning (and follow-up on planning) in a larger software development process,
- explain the importance of design and structure, as well as testing and integration for the successful implementation of a distributed web/mobile system,

Competence and skills

- work as part of a software development team,
- take responsibility for a role within a software development team,
- use a modern programming language to implement (parts of) a distributed software system,

Judgement and approach

- assess the additional challenges presented by the introduction of distribution in a software project,
- identify the challenges of developing web/mobile systems, and
- identify the challenges of planning and execution of a software development project.

Course content

In this project course, the students work together in groups. Each group produces a software prototype, and relevant project documentation. Each student is assigned a role in the project group, where they are expected to be responsible for the tasks associated with the role. Students are encouraged to change roles during the development process to build a holistic picture of the developed system. During the project students are presented with the opportunity to use and develop their skills in the areas Software Architecture and Web/Mobile development. The project is challenging, and requires the students to collaborate and organize their work efficiently. There is little external steering in this course: the students are responsible for the overall planning and setting their own internal deadlines in order to finish the project on time.

The students combine their theoretical knowledge about Software Architecture and Distributed Web/Mobile Systems. The students take a realistic industrial problem and solve it by creating a software architecture description of a web/mobile system, and by designing and implementing this system. The system may partly consist of existing systems, after which the implementation problem becomes an integration problem. During this project, the students are responsible for their own software process: they have to supervise and follow-up on their own processes in order to finish the project.

Form of teaching

The teaching consists of lectures, weekly group meetings and examination parts, as well as supervision in connection to the meetings.

Language of instruction: English

Assessment

The course is examined through artifact review and oral examination based on the course learning outcomes.The course is assessed by a demonstrable system, software architecture document, and other project documentation, which form the basis for an examination where questions are asked about the submitted artifacts. Furthermore, in order to assure individual grading, each student is required to traceably (i.e. with visible support that is possible to grade) show and argue for his or her:

- artifact contributions to the project and subgroup he/she was a member of
- role in the project group,
- fulfillment of responsibilities for this role in terms of artifacts and activities,
- interaction and knowledge transfer activities with others, and how this contributed to the project as a whole.

Students are required to complete self- and peer-assessment forms during the course.

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 (VG), Pass (G) and Fail (U).
For the final grade Pass (G), students must deliver a working and tested software that fulfills the minimum requirements along with the required documentation. To attain the grade, the individual student must have contributed in a reasonable way to the final project results of the team, and must be able to explain the team's work related to the software architecture document and other project documentation.

To be awarded Pass with Distinction (VG), students must in addition fulfill their role with excellence, manifesting this by choosing suitable tools and/or techniques, and

supporting this with good documentation.

Course evaluation

The course is evaluated through a 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

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

It is also up to the students to identify and address such needs based on their particular project focus.