



COURSE SYLLABUS

Programvaruarkitektur och kvalitet

Software Architectures and Quality

7,5 ECTS credit points (7,5 högskolepoäng)

Course code: PA1308

Educational level: Basic level

Course level: G2F

Field of education: Technology

Subject group: Computer Technology

Subject area: Software Engineering

Version: 6

Applies from: 2012-09-03

Approved: 2012-06-27

Replaces course syllabus approved:

1 Course title and credit points

The course is titled Software Architectures and Quality/Programvaruarkitektur och kvalitet and awards 7,5 ECTS credits. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

2 Decision and approval

This course is established by School of Computing 2008-06-18. The course syllabus was revised by School of Computing and applies from 2012-09-03. Reg no: COM-560-0085-2012

3 Objectives

Software Architectures is an important technical concepts in modern large-scale software development that serves multiple purposes; planning development resources based on the architecture, analyzing the problem domain from a number of perspectives with the help of architecture, and one abstracts large amounts of information to obtain a useful overview by architecture. In addition, the architecture, and how one chooses to construct the architecture, a key component in planning and achieving a certain level of quality in a system, thereby determining how successfully the system will be.

In this course students are expected to acquire detailed knowledge of software architectures and software quality and, in particular, how the latter influenced by the former.

Furthermore, the student acquire an understanding of how to konstruera a software architecture based on modern methods and ideas such as design patterns, object-oriented frameworks, and component-based software technology that takes into account the intended product, the surrounding technology and the development organization in a way that creates long-lasting and durable system with a planned and predictable level of quality.

4 Content

The course comprises the following elements:

- Quality aspects in software and software architecture.
- Architectural styles, languages and patterns.
- Methods for architectural design and evaluation.
- Component-based software engineering.

5 Aims and learning outcomes

On completion of the course the participant will:

- Be able to clearly express an in-depth insight in the area of software architecture (standards, key concepts, software quality definitions etc.) and be able to name and describe a number of key issues related to this area.
- Be able to clearly express an in-depth insight of quality in software, and how this is realised in quantifiable goals.
- Be able to independently, both on a theoretical level and in practice, select between a number of architectural styles, languages and patterns depending on the requirements, and discriminate between them.
- With an attention to details be able to create and document a software architecture consisting of several views and taking several different concerns into consideration.

6 Generic skills

The following generic skills are trained in the course:

- Planning and time management
- Critical thinking
- Teamwork
- Problem solving
- Analytical ability

7 Learning and teaching

The course is organized around a number of lectures where the students are expected to actively participate by discussing, questioning, and

contributing their own experiences. The lectures are given early in the course, in order to provide a solid foundation for the subsequent assignments. The assignments are constructed to help the students reflect upon previous experiences, literature and research articles, and relate these with each other. This gives a mixture of state-of-the-art as described in research articles, and state-of-practice as evidenced by personal experiences, the course literature, and industry studies. The students are also expected to reflect upon what can be done better, i.e. process improvement. The teaching language is English.

Authors: Bass, L., Clements, P., and Kazman, R.
 Publisher: Addison-Wesley
 Published: 2003, Number of pages: 560
 ISBN10: 0321154959
 ISBN13: 9780321154958
 2. Zen and the art of Motorcycle Maintenance
 Authors: R.M. Pirsig
 Publisher: William Morrow
 Published: 1974, Number of Pages: 412
 ISBN-10: 0688002307
 ISBN-13: 978-0688002305

8 Assessment and grading

Examination of the course

Code	Module	Credit	Grade
1210	Assignment 1	1.5 ECTS	F-A
1220	Assignment 2	1.5 ECTS	F-A
1230	Assignment 3	1.5 ECTS	F-A
1240	Assignment 4	1.5 ECTS	F-A
1250	Assignment 5	1.5 ECTS	F-A

The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent. The examination consists of a number of tasks, including at least one individual, where the unweighted and rounded average gives the course final grade.

9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

10 Prerequisites

The student must have successfully completed a total of 30 ECTS of the following courses: Object-oriented Systems Development, 7.5 ECTS, Datastructures and Algorithms, 7.5 ECTS, Database Systems, 7.5 ECTS, Computer Networks, 7.5 ECTS, Real-Time Systems, 7.5 ECTS, Operating Systems and Distributed Systems, 7.5 ECTS

11 Field of education and subject area

The course is part of the field of education and is included in the subject area Software Engineering.

12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

13 Course literature and other teaching material

Applied Software Architecture

Authors: Hofmeister, Nord

Publisher: Addison-Wesley Professional

Published: 2000, Number of pages: 400

ISBN10: 0201325713

ISBN13: 9780201325713

Reference literature

1. Software Architecture in Practice, Second Edition