

Course PM: Project course IT, TDA367/DIT211, LP4 2011

Course responsible	Joachim von Hacht Tel: 1003 Room: 6482	hajo@chalmers.se
Examiner	Joachim von Hacht	
Lecturer	Joachim von Hacht	
Assistants	Joachim von Hacht Pelle Evensen Daniel Arvidsson Emil Djupfeldt Adam Waldenberg	hajo@chalmers.se evensen@chalmers.se arvidsd@student.chalmers.se djupfeld@student.chalmers.se adam.waldenberg@gmail.com
Course site	http://www.cse.chalmers.se/edu/course/TDA367/	

General

This course is intended to make you familiar with the principal activities of software construction: requirements elicitation, analysis, design, implementation, testing, and documentation. The idea is to practice what you have learned in the courses Object-oriented programming (TDA545) and Object-oriented programming, advanced course (TDA550). On top of that, you will gain experience in specifying, designing and implementing a system from scratch, in a (small) team.

Course uses English for all written documents (and code (...and so should you)). Spoken language is Swedish (sadly with a lot of svenglish computer terms).

Collaboration with LSP310

This applies only to students taking the course "Kommuniktion och ingenjörskompetens", LSP310. The oral presentation is part of the grading for both courses. See examination below.

Schedule

See TimeEdit.

Literature

Recommended;

Steven C. McConnell, *Sample chapters from Code Complete 2nd ed*, at <http://www.cc2e.com/>

Jan Skansholm, *Java Direkt med Swing*, Studentlitteratur

Bernd Bruegge, Allen H. Dutoit, *Object-Oriented Software Engineering: Using UML*,

Patterns, and Java 2nd ed, Prentice Hall
Joshua Bloch, *Effective Java 2:nd ed*, Addison-Wesley
See also links from course page.

Environment

We recommend (and can handle) the Eclipse IDE. Our Eclipse installation has the following plug-ins added: EclEmma (code coverage), Subclipse (code versioning for Subversion) and UMLet (UML diagram drawing).

Organization

Lectures

There are a few introductory lectures, see course site.

Workshops

During the first weeks there are some workshops to help you (your group) to quickly get going. The workshops are hands-on (in lab rooms). Workshops are optional but strongly recommended. See course site.

Groups

The course is carried out in groups of four students. If you are a group of four students who want to work together, you may sign up for an empty group. However, if you are alone or a group of fewer than four persons, you should join an existing, not yet full, group. When you add your name to a group (having an open slot), you don't need to ask their permission to join. However, you need to inform them.

We will only allow group size different from four when this is necessary because the total number of students is not congruent to zero modulo four.

Project

Each group is supposed to deliver a project. For project details see Project PM on site.

Supervised meetings

Each week you will have a meeting with your supervisor. The meetings are mandatory!

If you miss a mandatory meeting you will have to do a complementary assignment (write a paper)! This is also applicable if you have too many late arrivals.

Group meetings

The group is assumed to organize at least two group meeting each week. The meetings should be documented (agenda template on course site).

Examination

Basic requirements for the group to pass;

- Have handed in and got the project accepted.
- Each week have organized two meetings for the group.
- Have done an oral "customer" presentation of the project (slides and demo assumed).
- Have attended some of the other groups presentations and acted as "opponents" on one of them (at least a half day).

For individual of groups to pass;

- Have fulfilled the group requirements.
- Have visited all mandatory meetings (or handed in the complementary assignment).
- Have been an active member of the group.
- Have participated sufficiently to the final deliveries (project, documents).
- Have spoken at the oral presentation.
- Have asked some questions (as an opponent).
- Have delivered a self estimation, see course page.

It will be possible to trace individual code contributions. If you work in pairs (or similar) you must take turns when committing into the version handling system.

Grades

Individual (U/3/4/5, U/G/VG). Grading will be done in two phases;

1. Group; Depending on the project, the documentation quality (not size) and the oral presentation a group grade will be assigned.
2. Individual: We'll estimate each students contribution to the project, documentation and presentation. High or low contributing students could get higher or lower grades. Normally the group and individual grade will end up the same.