Assignment 2

In this assignment you can work *in pairs or individually*. The theme is design patterns. This is an important field in software engineering. We have not covered this on the lectures so this assignment may require that read more material.

Starting date: 2010-02-24*Due date*: 2010-03-09

Each task includes an estimate of the number of pages you should prepare as answers. The total number of pages is 11 + code (including a front page with names, email to contact person, and the time log at the back.)

If you have questions, don't hesitate to contact Jesper. Mail him to arrange a meeting.

Submitting answers

Your answers should be sent as a pdf-file to Jesper.Andersson@vxu.se with the following text as subject.

`2DV103: Assignment1 Lastname1 Lastname2`. This is important since we use automatic sorting of incoming mails.

Timelog

Track the time you spend on this assignment. Create a table where you log every activity for each task. Include the log in the pdf-file that contains your answers

---

Grading

Your report will receive a percentage where 100% is perfect and <50% is failed. You are allowed to improve your work after the initial submission before deadline. The grade is final, i.e., you will not get an opportunity to correct/improve after grading.

Your answers should be your own! You are not allowed to quote without appropriate references to the original work! Each report must pass through a plagiarism system before correction. If plagiarism is detected, the assignment receives 0% and a formal investigation will be initiated.

Submissions that arrive after the due date will be downgraded by 25 %-units directly.

Task 1 – Design Patterns (2 pages)

Patterns are used on different levels, for different purposes, we have several different software patterns, including architecture patterns or styles, design patterns, and even organizational patterns. Present and discuss a classification of patterns.

Task 2 – Apply design patterns (5 pages)

Design a static structure (class model) which models different user roles in a chat/forum application. You will have to come up with the functional requirements on your own. Look for widely used design patterns and use them where applicable. Document your design and discuss/motivate your design decisions.

Task 3 – The Visitor pattern (2 pages + code)

Describe the visitor pattern in detail. Provide an implementation of an example system which uses the pattern in an object oriented programming language of your choice.