



# Assessment Criteria For Seminar 3, Implementation

## Object-Oriented Design, IV1350

The assessed person's score will *not* be affected by your comments. *Try to give concrete suggestions. Motivate your comments, give examples, **try not to write just "yes" or "no"**. You are not required to evaluate all bullets below*, you may choose those you find most interesting. Make sure to discuss and/or ask the teacher about questions regarding your own or the assessed solution(s).

- Is the code easily understood? Do all declarations have explaining names? Are code conventions followed?
- Is there duplicated code?
- Are there methods or classes that are too long?
- Are primitive data or static members used instead of objects?
- Are there methods with unnecessarily long parameter lists?
- Is there one comment per public declaration and no comments inside methods?
- Are the MVC and Layer patterns still used correctly?
- Is there still low coupling, high cohesion and good encapsulation?
- Is a framework, like JUnit, used for the tests?
- Are all tests self-evaluating, that is, do they print an informative message if a test fails, and do they print nothing if a test passes?
- Are the tests placed in the same package as the SUT, but in a different directory?
- Is there one test class per tested class?
- Are all branches of if-statements tested?
- Do tests pass all interesting parameter values to the tested methods?
- If the assessed person wrote unit tests for all classes, are the tests complete? Are there tests for all public and package private methods (except simple setters/getters)?
- If the assessed person wrote unit tests for only two classes, are the tests meaningful?
- Is the use of programming best practices explained in the report?
- Is the method and result explained in the report? Is there a discussion? Is the discussion relevant?