

<Author name(s) or group number>

<Project Title or acronym>



<Project title>

by

<author1>, <author2>

Supervisor: <name of supervisor>

Project report for <name of the course> in <semester, e.g. Autumn 2006>

based on report template version 3.0 (2006)

Agder University College
Faculty of Engineering and Science

Grimstad, 22 August 2006

Status: <Draft, Review or Final>

Keywords: <list of up to 5 keywords related to the problem and your solution>

Abstract: <abstract, 150-300 words>

You write the first version of the abstract when you have got a clear view of the problem, how to solve it, and the results you hope to get. The abstract must be revised later, during the final write-up of the report. During the final write-up you also adjust the text across all chapters in the report, to be consistent and logically well connected. This short abstract cannot be a complete short summary of all work and all results. Concentrate on the problem, how it was solved, the most important results and consequences for the problem owner or the user.

In particular, answer the following questions:

- *What is the area your work is done in?*
- *What is the problem you have tackled / goal you want to achieve?*
- *What is the essence of your solution / main result?*
- *What are the most important conclusions?*
- *What are the consequences or new benefits for the problem owner or user?*

[This work is licensed under the Creative Commons Attribution-ShareAlike License \(http://creativecommons.org/licenses/by-sa/2.5/\).](http://creativecommons.org/licenses/by-sa/2.5/)

Version Control

<i>Version</i> ¹	<i>Status</i> ²	<i>Date</i> ³	<i>Change</i> ⁴	<i>Author</i> ⁵

-
- 1 **Version** indicates the version number starting at 0.1 for the first draft and 1.0 for the first review version.
 - 2 **Status** is DRAFT, REVIEW or FINAL
 - 3 **Date** is given in ISO format: yyyy-mm-dd
 - 4 **Change** describes the changes carried out since the previous version
 - 5 **Author** is the one who did the change

Table of Contents

1	Introduction.....	4
1.1	Acknowledgements.....	4
1.2	Report outline.....	4
2	Problem description.....	5
3	Background	6
4	Solution.....	7
1.3	Requirements.....	7
1.4	Design Specification	7
1.5	Implementation	7
1.6	Validation and Testing	7
5	Discussion	8
6	Conclusion.....	9
	Appendices	10
Appendix 1	Glossary & Abbreviations.....	10
Appendix 2	Project Management (optional)	10
Appendix 3	References.....	10

1 Introduction

First some comments on the use of this report template. It is very general, and can be used in most student (and other) projects. It is also quite detailed, and this means it also serves as a check-list, so you can be reasonably sure you have not forgotten to document some important element which should be part of a report. But then, in very simple and short-duration projects, you do not necessarily have to use all elements of the template. If the report is used in a 5th grade master project, you must use most elements of the template. If on the other hand, the template is used in a bachelor project or the very first project as a 4th grade master student, you can use less of the document structure given here, and write a simpler type of report.

The first time you see and use this template, you will most certainly feel a bit lost and confused. But do not worry. This is the same for everybody. You will work your way through the template, in cooperation with the teacher and your group mates, and you will learn through this process. Also remember the advantages of the template: It guides you through the steps of building a report with the right structure and the right contents. Actually the report template is a great helping tool.

In this chapter you introduce the reader to the problem and your work, by first providing the general background, then narrowing down to the specific field of interest, and finally the problem to be solved.

The problem statement shall be included.

First draw up the main lines of relevant previous work in this field. Then describe where and how your problem and work fits in. In later chapters, you can provide more detail if necessary.

Answer the following questions:

- What is the domain or environment you are working in?
- What is the problem you solve, the goal of your work?
- Who has stated the problem (problem owner): teacher, company ...
- How do you intend to solve this problem, what is your strategy for this?

1.1 Acknowledgements

Optionally, you might want to put in who helped you achieve your result(s).

1.2 Report outline

Describe here the structure of the report. This means for each chapter you describe very briefly the main issues contained in the chapter.

2 Problem description

Describe the problem you are dealing with (1/2 – 1 page) based on the purpose and motivation from the introduction and on the review of literature. If possible, include scenarios related to the problem description. This means that you build a scenario where your solution interacts with its environment. Describe the problem **you** are solving and distinguish it from related problems⁶.

Answer the following question:

- What exactly is the problem you are solving?
- State possible research questions related to the problem
- What is your motivation to solving the problem?
- What are the high-level requirements for the solution; are there any requirements to the solution method or to the results?
- How do you intend to solve the problem, (what is your solution strategy?)

⁶ For example you might want to model a protocol that is solving a security issue. Then your problem is the modelling, not the security issue.

3 Background

First a comment on the following text in the template: It is highly academic and somewhat abstract. With a very simple practical problem and a simple solution, you do not necessarily have to write much about state-of-the-art. Look through the text we provide here, and use what you find relevant and reasonable. There is no need to overdo. Discuss this with your teacher or supervisor.

Basically, this chapter is only about performing a review of the literature, and explaining where and how your problem comes in. What is literature in this context? With the wide span of student problems to be covered by this document template, it can be journal papers, conference papers, documents on the web, company product specifications or handbooks and even private correspondence (if correctly referenced).

Describe the state-of-the-art in the area concerned, and how your project intends to enhance the state-of-the-art in that area. Make sure that your description is connected to the problem described in the previous chapter(s). You may introduce further subsections if needed, (like for instance **3.1 Existing tools**.)

Provide references to **all** external sources you used in the report, such that you do not have problems with incorrect or illegal use of sources (see also [1]).

Also describe the use of standards in the project, if any.

Finally, introduce all terms that are central to your work (remember to reference the definitions properly). Indicate that short definitions are available in the appendix glossary.

This chapter answers the following questions:

- What is the main field of interest, the domain or environment where your problem and the solution fits in?
- How did others solve the problem you have to solve, if you know a solution, or how have others solved related problems? What is the current status?
- Are there standards related to this problem?
- Are there preferred and/or known ways to solve the problem?
- What is the outline of your way of solving the problem?

4 Solution

The structure of your report in this section will vary depending on the approach you have chosen and the nature of your project (see also [2]).

In this template we assume you have a **design** problem to solve with the four parts:

- *requirements,*
- *design specification,*
- *implementation, and*
- *test/validation.*

If you instead have a **theory** problem, the four parts are:

- *definition of the objects of study,*
- *theorem about their relationships,*
- *proof of the relationships, and*
- *interpretation of results.*

Finally, for **modelling** tasks, the parts are:

- *establishment of hypothesis,*
- *forming a model and predicting a result,*
- *experiment and data collection*
- *result analysis.*

Then, in Chapter 4, you will provide answers to the following questions:

- Which approach did you choose and why?
- What did you achieve with this approach (result/solution)?
- How does this solve the problem?
- What were your main design principles/intentions?
- How does your design relate to the high-level requirements from the problem statement?

Remember that the following four subsections are written for a design report. If your project has a different nature you would have different subsections (see above).

1.3 Requirements

Present the requirements for the solution (prototype) or function under development. Make use of established methods such as UML.

1.4 Design Specification

Provide a design specification of your prototype. Use diagrams to present what is to be implemented. Discuss the software tools that are to be used during the implementation. Often, it is useful at this stage also to design for testing and testability. How do you want to test in order to validate your solution? Can you build into the design, features which help testing specific (critical) parts of or the whole solution? What input data would you need for testing? If you can, define some test cases which cover all the properties (the functionality or the performance) you want to include in the solution.

1.5 Implementation

Present your prototype as an implementation of your design (see section 1.4), including code excerpts of interesting aspects of the implementation. Do not put the complete code in here.

1.6 Validation and Testing

This is about testing your solution to prove or at least show (experimentally) that it actually solves the whole problem or as much as possible of the problem. Describe all tests and show how they cover your test cases, and test the requirements of section 1.3. If some of the requirements are not fulfilled, report on it. Do not change the original requirements list! This section is supposed to present the objective evaluation of your work (test results)

5 Discussion

First some introductory comments: Discuss your findings and the way you arrived there. Basically the discussion compares the problem and its requirements, with your solution or result. Does your solution deliver the wanted solutions to the problem? You can discuss the solution versus given requirements, but you can also discuss the solution in general terms, like advantages or disadvantages (pros and cons), and also in terms of computing efficiency, energy efficiency, cost efficiency etc. This means that you have or find some relevant or (specified) discussion criteria, and perform the discussion of the solution against this set of criteria. (*Terminology: One criterion, more criteria*)

Your may also discuss your solution and its relation to possible alternative solutions.

In contrast to the testing in the previous section, this section gives an interpretation and an evaluation of the results.

Discuss the solution or the project outcomes: do they fit the initial project purpose and problem, what was achieved, in which points you failed and why, what are the future paths of development, etc. You may also give a high-level summary of the evaluation results.

Also discuss briefly alternatives you have not taken and relative advantages and disadvantages with them.

Discuss and evaluate your decisions concerning technology, architecture, testing, and experiments. Discuss the relevance and reliability of the results and evaluate the methods used. Do also discuss your project management and how it helped you to achieve the results.

You may want to divide this section into subsections.

In short, you answer the following questions:

- What are your results?
- Why did you choose the solution presented?
- What other alternatives did you see?
- How well does your solution solve the problem or satisfy the requirements? Does it solve the problem completely and in all aspects, or have you found only a partial solution?
- What did you learn from the project?
- What would you have done differently if you were to start again?
- What would you propose as future work in this area?

6 Conclusion

Very briefly outline problem, solution, and the properties of the solution, and finally make a statement that you have actually solved the problem or proved or disproved the hypothesis. That statement is your basic conclusion. Then explain how the results make a difference. Tie together examples and results with the points made in the report.

Answer the following question:

- What was the problem? (2-4 sentences)
- What is the solution you worked on? (2-4 sentences)
- What was the outcome or result of your effort? (2-4 sentences)
- How does your solution make a difference to existing solutions? What are the new benefits for users or problem owners?
- What would you propose as next steps or future work? (2-4 sentences)

Appendices

Appendix 1 Glossary & Abbreviations

Short definitions of terms used and references to further relevant glossary sources

Appendix 2 Project Management (optional)

Include here a description of the project and how you organised it.

Answer the following questions:

- What is the project title, and who was the problem owner (external company)?
- Who were the project participants?
- How did you organize the project as a team, was there a project leader, did you have separate roles in the project? How did you distribute the work internally in the project: who did what? Did you cooperate with other teams?
- Which milestones did you have? What dates?
- If you have a cost budget, present it.
- Is your project part of a larger project? What is the relation between the two?

Appendix 3 References

Follow the IEEE guidelines for references

see: http://www.ece.uiuc.edu/pubs/ref_guides/ieee.html

- [1] HiA, Bruk av kilder i skriftlige arbeider ved Høgskolen i Agder, August 2006, <http://www.hia.no/stud/eksam/kilder.htm>
- [2] Peter J. Denning, Douglas E. Comer, David Gries, Michael C. Mulder, Allen Tucker, A. Joe Turner, and Paul R. Young, Computing as a Discipline, Communications of the ACM, Volume 32, Number 1, 1989