

Bachelor project SOFT

Jonas Hein

11-03-2021

1 The project

The bachelor project is a project where the student investigates a software-related problem, propose a solution, and does some implementation to demonstrate the solution. The student should strive to include elements from the courses passed during the programme (Large Systems Development, Databases, Testing, System Integration).

The project may be done in groups of up to 4 people. Having a larger group should increase the project scope, as it raises the general expectations of the project.

The project covers 15 ECTS points, which is roughly equivalent to

$$15 \cdot 27.4 \text{ hours} = 412.5 \text{ hours}$$

2 Report

Size of Report The maximum page count for the bachelor report is given by the formula below.

$$\text{maxPageCount} = 40 + 20 \cdot \text{numberOfStudents}$$

Reports shorter than 23 will usually be experienced as short, although there is no official minimum size.

Language The report can be written in either Danish or English.

Contents The report should contain a thorough description of the work that has been done during the bachelor project, as well as an evaluation and reflection on the work.

3 Report components

The instructors recommend the following sections to be in the report. The components of the report do not have to be presented in exactly the same order but are recommended to be present.

Project title Project title Cover page: Project title, your name and student identification, school and supervisor.

Abstract Four sentences: Problem statement, further elaborated in Introduction. It can be seen as an advertisement for the rest of the report. The final abstract is written in the end of the project

- What is the problem?
- Why is the problem interesting?
- What is the solution?
- What are the implications of the solution?

Introduction Project summary:

- Motivation — Why this topic is important nowadays.
- Project objectives — what is the expected result.
- Project tasks — what is to be done for achievement of the objectives - steps in the work, such as getting acquainted with the state-of-the-art and trends in the area, selecting a development methodology, creating a design, selecting development tools and environments, programming, testing, implementation and evaluation, as appropriate.
- Scope of the project — what is not an objective or a task.
- Brief description of the other chapters that follow — one paragraph per each.

State of the art and trends Survey of current technologies and similar solutions in this area – to show that you are well informed and step on the available achievements: For example, you can present some relevant knowledge gathered from both previously studied modules and external sources, organized in a structured form. Here you can also describe and compare alternative development methodologies, technologies, environments, frameworks, or programming 2 languages that could have been used, and to explain and argue your choice. Inside the text, a citation or a reference to the used sources (books, articles, web pages) is required. The sources can be listed either underscore, or in an Appendix.

Requirements specification and solution design Formulation of project foundation – a solid ground of your solution: This part reflects the design process, or the methodology you have followed. Analysis – who are the users, use cases and scenarios, intended user experiences. The analysis lead to specifications – formal functional and non-functional requirement to the application. Specifications lead to requirements: you consider how to design the solution – system architecture, data models, visual interface, control, operability, algorithms, integration and other components, as appropriate. Use graphics for visual presentation of your concepts as much as possible.

Solution development and implementation Presentation of the application, test procedures, deployment and maintenance environments: This part presents the product in technical terms. Implementation and test environment — test strategies, test plan, test suites, demo and screen captures, usability evaluation, etc. Units of code, packages, deployment, supported interfaces, algorithms, input and output, supported file formats, frameworks, servers and clients, etc.

Conclusions Brief summary: What has been done and the benefits of it. Recommendation for future extensions and upgrades. Reflection on the work and the product.

Appendices Reference of information sources - as many, as possible, listed according a standard (see [https://iee-dataport.org/sites/default/files/analysis/27/IEEEguideand/or user manual](https://iee-dataport.org/sites/default/files/analysis/27/IEEEguideand/or%20user%20manual.pdf), if appropriate. Source code of main components, if appropriate.

Final works Project and documentation completed. Visual (power point) presentation and a demo prepared. It is recommended to discuss the draft report with the supervisor before the final!

References

- [1] <https://datsoftlyngby.github.io/soft2020fall/resources/bbe51cf2-bachelorProject.pdf>.