

Final Report — Template

COE 485: Senior Design Project (151)

1 Introduction

- Introduce the *general issue* with which the project deals, e.g. rising cost of health care, environment, etc.
- Why is it an issue? supporting evidence; statistics, news, articles, etc.
- Briefly describe how the project deals with the issue.
- Project Impact on society *locally* and *globally*:
 - Positive impact.
 - Possible negative impact, due to misuse or unaccounted for risks.

2 Problem Statement

The specific problem the project is trying to solve.

3 Background

- Related terminology, concepts, and technology.
- Existing solutions, e.g. products, research.

4 Requirements and Specifications

- Functional user requirements.
- Non-functional user requirements, e.g. response time, power consumption, cost, size.
- Technical specifications:
 - Derived from and maps to customer requirements.
 - Must be specific and testable. The product must meet all specifications.

5 System Design

Completely document the project design. Use graphical illustrations as much as you can.

5.1 Solution Concept

- General approach of solving the stated problem.
- Description of used/developed *algorithms*.
- Alternative approaches and algorithms, comparison, and selection criteria.
- Sub-function identification.

5.2 Architecture

- System architecture and components.
- Alternative architectures, comparison, and selection criteria.
- Hardware vs. software components.
- Functions of each component.

5.3 Component Design

For each hardware and software component:

- Custom vs. off-the-shelf, and justification for developing a custom component.
- Off-the-shelf components: alternatives, comparison, and selection criteria.
- Custom components:
 - Design and implementation, e.g. flow chart, state machine, pseudocode.
 - Component design alternatives, comparison, and selection criteria.

5.4 System Integration

- Standard vs. custom interfaces between components, and justification for developing custom interfaces.
- Specification of custom interfaces.
- Component interaction, e.g. sequence diagrams.
- Integration methodology: how the system is put together while ensuring it is working correctly.

5.5 Design Evolution

How and why the system design changed from the initial design to the final design.

6 Testing, Analysis, and Evaluation

- Testing methodology and results: how did you determine whether the system is operational and meets all the requirements and specifications?
- System *analysis* and *evaluation*: what attributes of your system did you study? how did you analyze and/or measure them? e.g. performance, efficiency, reliability, security – as applicable.

7 Issues

- Bugs and challenges. For each issue, list:
 - The issue.
 - Attempted, unsuccessful resolutions.
 - Final resolution: solution, workaround, issue ignored.
- Limitations and constraints of the design and/or the implementation.

8 Engineering Tools and Standards

- Tools: which of the available tools did you use, and why? e.g. simulators, emulators, boards, development environments, IDEs, debuggers, software frameworks.
- Standards: which of the relevant standards did you use, and why? e.g. communication protocols, storage formats, component interfaces, specification languages.

9 Teamwork

For each team member, list:

- Responsibilities: tasks managed by the team member.
- Contributions: tasks contributed to by the team member.
- Expertise: areas in which the team member is knowledgeable and often consulted.

10 Conclusion

- What was learned.
- What would you do differently in a similar project?
- Your own conclusions as a result of working on this project.