
SOFTWARE MAINTENANCE DOCUMENT

for

Encost Smart Graph Project

Version 1.0

Prepared by: **Student Name**
SoftFlux Engineer

SoftFlux

May 18, 2023

Contents

1	Introduction/Purpose	3
1.1	Purpose	3
1.2	Document Conventions	3
1.3	Intended Audience and Reading Suggestions	3
1.4	Project Scope	3
2	Specialized Requirements Specification	3
3	Maintenance	4
3.1	Maintenance Task #1	4
3.1.1	User Story	4
3.1.2	Problem/modification request	4
3.1.3	Problem/modification analysis	4
3.1.4	Acceptance/rejection	5
3.1.5	Modification implementation	5
3.1.6	Maintenance review/acceptance	5
3.2	Maintenance Task #2	5
3.2.1	User Story	5
3.2.2	Problem/modification request	5
3.2.3	Problem/modification analysis	5
3.2.4	Acceptance/rejection	5
3.2.5	Modification implementation	6
3.2.6	Maintenance review/acceptance	6
3.3	Maintenance Task #3	6
3.3.1	User Story	6
3.3.2	Problem/modification request	6
3.3.3	Problem/modification analysis	6
3.3.4	Acceptance/rejection	6
3.3.5	Modification implementation	7
3.3.6	Maintenance review/acceptance	7
4	CI/CD Pipeline	7
5	Conclusion	7

Revision History

Name	Date	Reason for Changes	Version

1 Introduction/Purpose

1.1 Purpose

What is the purpose of this document? Take a look at the SRS, SDS & testing documents as a guideline.

1.2 Document Conventions

Are any conventions, acronyms, specialised terms, etc. used in this document? If so, list them here, along with their meaning. Take a look at the SRS, SDS & testing documents as a guideline.

1.3 Intended Audience and Reading Suggestions

Who is the intended audience for this document? Take a look at the SRS, SDS & testing documents as a guideline.

1.4 Project Scope

What is the scope of the project? Take a look at the SRS, SDS & testing documents as a guideline.

2 Specialized Requirements Specification

Have you found any ambiguities, missing detail, missing features, etc. in the existing documents? Have you discussed them with your client (your lecturers) and received confirmation/clarification? If so, this information should go here.

3 Maintenance

Write a Product Backlog that covers all *High Priority* functional requirements in the SRS, based on the design laid out in the SDS, and the tests laid out in the Functional Test Plan.

3.1 Maintenance Task #1

Rename the *Maintenance Task #1* section header to more closely align with the specific maintenance task.

3.1.1 User Story

The maintenance user story as stated in the assignment specification

3.1.2 Problem/modification request

Reword the user story into a problem/modification request

3.1.3 Problem/modification analysis

Maintenance Category: State the maintenance category (preventative, perfective, corrective, or adaptive)

Impact Analysis

- **Verify the problem:** Explain the issue
- **Implementation options:** Explain how you might implement it
- **Effort:** (lines of code, additional learning)
- **Resources:** (time to make the change, time to run tests)

3.1.4 Acceptance/rejection

Are you accepting or rejecting the request. If rejecting, say why.

3.1.5 Modification implementation

If the maintenance task was accepted, then you are required to implement it. You should include evidence of your implemented solution in the document, whether this be a textual description, a diagram, a screenshot, or a combination of all three.

3.1.6 Maintenance review/acceptance

If the maintenance task was accepted, then you are required to test it. You should include evidence of your testing in the document, whether this be a description of manual tests (with screenshots) or a description of a unit test(s). If unit tests are developed they should be uploaded to GitLab.

3.2 Maintenance Task #2

Rename the *Maintenance Task #1* section header to more closely align with the specific maintenance task.

3.2.1 User Story

The maintenance user story as stated in the assignment specification

3.2.2 Problem/modification request

Reword the user story into a problem/modification request

3.2.3 Problem/modification analysis

Maintenance Category: State the maintenance category (preventative, perfective, corrective, or adaptive)

Impact Analysis

- **Verify the problem:** Explain the issue
- **Implementation options:** Explain how you might implement it
- **Effort:** (lines of code, additional learning)
- **Resources:** (time to make the change, time to run tests)

3.2.4 Acceptance/rejection

Are you accepting or rejecting the request. If rejecting, say why.

3.2.5 Modification implementation

If the maintenance task was accepted, then you are required to implement it. You should include evidence of your implemented solution in the document, whether this be a textual description, a diagram, a screenshot, or a combination of all three.

3.2.6 Maintenance review/acceptance

If the maintenance task was accepted, then you are required to test it. You should include evidence of your testing in the document, whether this be a description of manual tests (with screenshots) or a description of a unit test(s). If unit tests are developed they should be uploaded to GitLab.

3.3 Maintenance Task #3

Rename the *Maintenance Task #1* section header to more closely align with the specific maintenance task.

3.3.1 User Story

The maintenance user story as stated in the assignment specification

3.3.2 Problem/modification request

Reword the user story into a problem/modification request

3.3.3 Problem/modification analysis

Maintenance Category: State the maintenance category (preventative, perfective, corrective, or adaptive)

Impact Analysis

- **Verify the problem:** Explain the issue
- **Implementation options:** Explain how you might implement it
- **Effort:** (lines of code, additional learning)
- **Resources:** (time to make the change, time to run tests)

3.3.4 Acceptance/rejection

Are you accepting or rejecting the request. If rejecting, say why.

3.3.5 Modification implementation

If the maintenance task was accepted, then you are required to implement it. You should include evidence of your implemented solution in the document, whether this be a textual description, a diagram, a screenshot, or a combination of all three.

3.3.6 Maintenance review/acceptance

If the maintenance task was accepted, then you are required to test it. You should include evidence of your testing in the document, whether this be a description of manual tests (with screenshots) or a description of a unit test(s). If unit tests are developed they should be uploaded to GitLab.

4 CI/CD Pipeline

Include a screenshot and description of your CI/CD pipeline

5 Conclusion

Give a brief written summary of your maintenance and pipeline. This is a good time to go back through your document to make sure you haven't missed anything.