Software Engineering Report

SID:000000

**MOD003263**

**Submitted: December 2023**

Table of Contents

[1. Introduction 1](#_Toc149776694)

[2. Problem Definition Document 2](#_Toc149776695)

[3. Software Design 3](#_Toc149776696)

[3.1. Wireframe 3](#_Toc149776697)

[3.2. Database design 3](#_Toc149776698)

[3.3. Software design 3](#_Toc149776699)

[3.3.1. Use Case Diagram(s) 3](#_Toc149776700)

[3.3.2. Class Diagram 4](#_Toc149776701)

[3.3.3. Sequence Diagram 4](#_Toc149776702)

[4. Software Development Documentation 5](#_Toc149776703)

[4.1. User story 1. 5](#_Toc149776704)

[4.2. User story 2. 5](#_Toc149776705)

[5. Testing 6](#_Toc149776706)

[6. Plan of the Software Maintenance 7](#_Toc149776707)

[7. Conclusions 8](#_Toc149776708)

[8. References 9](#_Toc149776709)

# Introduction

Each chapter should start on a new page.

The main text should be Times New Roman 12pt and double-spaced.

Section sub-headings should be left aligned.

Throughout the project it is recommended never to go to deeper than three hierarchical subsection levels otherwise it can be confusing for the reader as to where one is in the overall structure of the project. So, for example a section 3.1.2 is acceptable, but say 2.3.4.6 is probably not. Try to keep to two levels wherever possible.

# Problem Definition Document

This part is common to all group members. Evidence of the author’s contribution has to be found in the logbook.

* What is the project aim?
* Resources required to complete the project. It might include a description of the team, the software used.
* List of User Stories

# Software Design

This chapter should include the design of the database and software.

Describe how the problem will be tackled (design)

## Wireframe

The wireframe is common to all group members.

The description of the wireframe is written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

## Database design

Describe the database’s structure

The database is common to all group members.

The description of the database is written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

## Software design

## Use Case Diagram(s)

To include the Use case diagram(s) and their description.

For the description please refer to the Fundamental of Design lecture notes.

The following link might help too. <https://computersciencesource.wordpress.com/2009/11/22/year-2-software-engineering-use-case-diagrams-descriptions/>

The Use Case diagram is common to all group members.

The description of the Use Case diagram is written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

## Class Diagram

The class diagram is built during the development of the user stories. For each user story the classes required are identified and then added to the class diagram.

The Class diagram is common to all group members.

The description of the Class diagram is written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

## Sequence Diagram

The sequence diagrams are built during the development of the user stories.

The Sequence diagrams are common to all group members.

The description of the Sequence diagrams is written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

# Software Development Documentation

This chapter is individually written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

This chapter contains the user stories or the parts of the user stories implemented by the author (NOT BY THE WHOLE TEAM).

Please check that the code you have written is readable:

* Use correct indentation
* Remove unnecessary whitespace
* Remove any unused commented-out code
* Remove old and redundant variable declarations
* Make sure all variables are declared
* Include brief descriptive comments for each section

Any piece of code coming from other resources has to be referenced.

Include a screen shot of the online repository showing the check-ins done by the author.

## User story 1.

Design (Class diagram, Sequence Diagram)

Implementation

Testing – the automated tests written by the author (if any)

## User story 2.

Design (Class diagram, Sequence Diagram)

Implementation

Testing – the automated tests written by the author (if any)

# Testing

This chapter is individually written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

Summarise the tests written by the author (their description is in chapter 4)

Record all the tests done manually even if they fail.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No** | **Test (Description)** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Plan of the Software Maintenance

This chapter is common to all group members.

# Conclusions

This chapter is individually written by the author (IS NOT COMMON TO ALL GROUP MEMBERS).

In Conclusions evaluate the application and reflect on what you have done/learnt. What else can be done to improve the application?

# References

Berndtsson, M., Hansson, J., Olsson, B. and Lundell, B., 2006. *Planning and Implementing your Final year project – with Success!*: A Guide for Students in Computer Science and Information Systems. Springer.

Bock, P., 2001. *Getting It Right. R&D Methods for Science and Engineering.* Academic Press.

Burke, R., 2004. *Project Management – Planning and Control Techniques.* Wiley.