

**CS6P05 Project**

*Your Project Title goes here…..*

**Project Report**

**Final Submission**

Name: Add your Full name

ID Number: Your ID number

Date: Monday, 14 October 2019

First Supervisor: Add Your Supervisor’s Name Here

Second Supervisor: Add Your Second Marker’s Name Here

# Declaration

**Module:** CS6P05 **Deadline:** 3pm Monday 7th January 2019

**Module Leader:** Dr. Quan Dang **Student ID:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PLAGIARISM**

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed below. Please sign below to say that you have read and understand these extracts:

Student signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: 14/10/19

This header sheet should be attached to the work you submit. No work will be accepted without it.

Extracts from University *Regulations* onCheating, Plagiarism and Collusion

Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples…

1. Cheating: including taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; copying coursework.
2. Falsifying data in experimental results.
3. Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.
4. Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment.
5. Collusion to present joint work as the work solely of one individual.
6. Plagiarism, where the work or ideas of another are presented as the candidate's own.
7. Other conduct calculated to secure an advantage on assessment.

(viii) Assisting in any of the above.

Some notes on what this means for students:

1. Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.

2. Taking extracts from published sources *without attribution* is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. "e = mc2 (Einstein 1905)". A *references* section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution.

# Abstract

The Project abstract is a brief description of what your project is about for the general audience in order for them to understand the essence of your work and its benefits, and to decide whether it is of their interest and worth further reading.

You may use your project proposal contents in this section but try to avoid any diagrams.

The length of this section should be around 300-400 words, to the maximum length of one A4 size.

# Table of Contents

[Declaration 2](#_Toc21980982)

[Abstract 3](#_Toc21980983)

[Table of Contents 4](#_Toc21980984)

[List of Tables 6](#_Toc21980985)

[List of Figures 7](#_Toc21980986)

[Chapter 1: Introduction 8](#_Toc21980987)

[1.2 Project topic and rationale 8](#_Toc21980988)

[1.3 Project Aims and Objectives 8](#_Toc21980989)

[1.4 Methodology 8](#_Toc21980990)

[1.5 The report structure 8](#_Toc21980991)

[Chapter 2: Background Research 9](#_Toc21980992)

[2.1 Literature review of related work 9](#_Toc21980993)

[2.2 Critical evaluation of related products/solutions 9](#_Toc21980994)

[2.3 The scope of the project 9](#_Toc21980995)

[2.4 Review and justification of theories/models/development platforms/tools selected for use in the project 9](#_Toc21980996)

[Chapter 3: Requirements Analysis and Specification or Problem Analysis and Specification 10](#_Toc21980997)

[Chapter 4: Software Design or Research Methodology Design 11](#_Toc21980998)

[Chapter 5: Implementation and Testing or Research Results and Result Analysis 12](#_Toc21980999)

[5.1 Software Implementation or Research Results 12](#_Toc21981000)

[5.2 Software Testing or Analysis of the results 12](#_Toc21981001)

[Chapter 6: Evaluation of Results 13](#_Toc21981002)

[Chapter 7: Conclusions 14](#_Toc21981003)

[7.1 A summary of what has been achieved in the project 14](#_Toc21981004)

[7.2 Reflections and lessons learned 14](#_Toc21981005)

[7.3 Future work 14](#_Toc21981006)

[Appendices 15](#_Toc21981007)

[Appendix A: Project Management 15](#_Toc21981008)

[1.1 The original project plan from the Proposal 15](#_Toc21981009)

[1.2 Review of your project process 15](#_Toc21981010)

[1.3 Amendments to the original plan 15](#_Toc21981011)

[1.4 Lessons learned in project management 16](#_Toc21981012)

[Appendix B: Sample dataset / Survey responses 16](#_Toc21981013)

[References 17](#_Toc21981014)

[Bibliography 18](#_Toc21981015)

# List of Tables

**Table 1: The project WBS activities** 11

# List of Figures

**Figure 1: Literature Review Process, Dawson (2015), p.92** 9

**Figure 2: Planned Software Development Tasks based on the BRIDGE process, Jalloul (2012 ), p.51** 10

**Figure 3: The revised project Gantt chart** 14

# Introduction

This chapter should introduce what the project is about and the structure of the remainder of the report.

## Project topic and rationale

Provide an overview of the project topic you are working on, explain the project’s motivation, why it is interesting, useful and what the problem in hands and challenges are.

## Project Aims and Objectives

List the project aims and objectives, and a concise description where appropriate.

## Methodology

Outline how you carried out the project (e.g., by using a Unified Process for system development).

## The report structure

Give a description of how the remaining part of the report is organised and what each chapter is about.

# Background Research

This chapter provides a critical review of related work, and products with an aim to demonstrate a specialist understanding of the project topic, including its fundamentals and the state-of-the-art, i.e. the current developments and potential areas (i.e. “gaps”) for future development.

The review of the related work provides the context for the project work within the chosen topical area and relate the project to what is already known and available. As a result, the scope of the project can be determined.

Further, the chapter also reviews and describes relevant theories, methods and software/tools, which will be used in the project. Alternative options are discussed, and choices are justified.

It is very important to note that your review should not be simply a description of what others have published in the form of a set of summaries, but should take the form of a critical discussion, showing insight and an awareness of differing arguments, theories and approaches. It should be a synthesis and analysis of the relevant and up to date published work, relating at all times to your own purpose and rationale.

This chapter includes the following main contents.

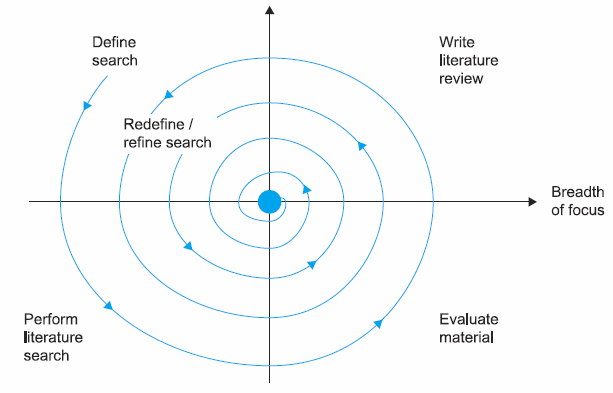
## Literature review of related work

## Critical evaluation of related products/solutions

## The scope of the project

## Review and justification of theories/models/development platforms/tools selected for use in the project

Below is an example of a figure with a reference, which will automatically be pickup by the Word generator of the List of Figures.



**Figure 1: Literature Review Process, Dawson (2015), p.92**

# Requirements Analysis and Specification or Problem Analysis and Specification

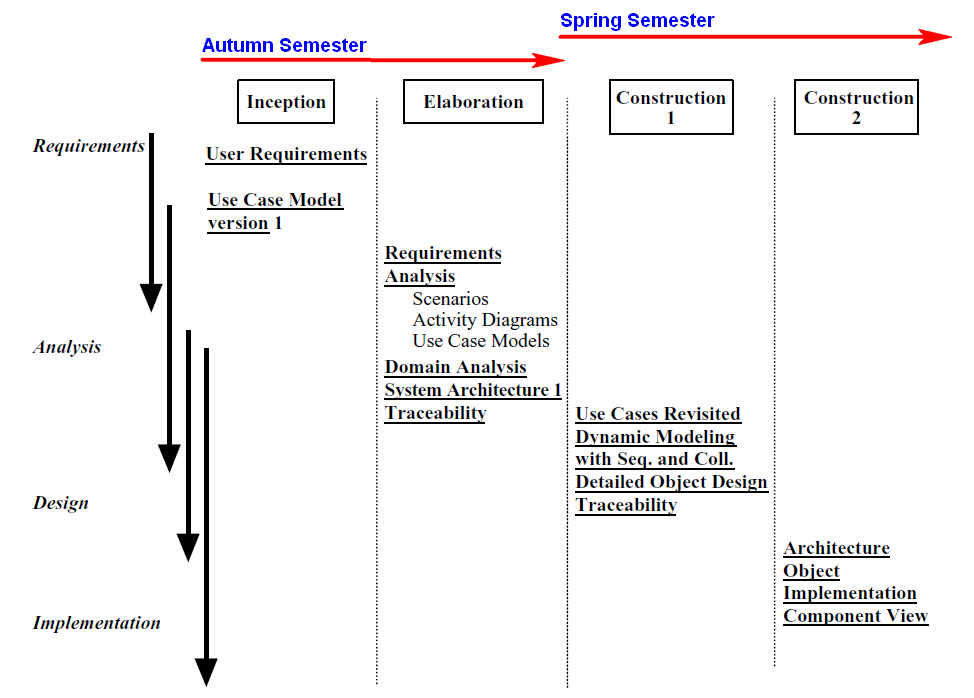
This chapter is eliciting **what** your system is required to do, not **how** it will do it.

If you develop a software, a structured approach should be used to determine the requirements of the system/ program or methodology being produced. There are several techniques particularly in the area of software design to assist in this process, and you are recommended to use a technique appropriate to your project, but which ensures that you fully analyse the system you are designing and fully understand its requirements.

This chapter should be concluded with a detailed and unambiguous specification of your **software** requirements (*functional* and *non-functional* requirements). Use cases technique is recommended for eliciting and representing the functional requirement.

**If your project is of an investigative type**, the result of the problem analysis should be a clear **problem statement** with a detailed problem description, research hypothesis, and **key research questions**.

Below is an example of a figure with a reference, which will automatically be pickup by the Word generator of the List of Figures.



**Figure 2: Planned Software Development Tasks based on the BRIDGE process, Jalloul (2012 ), p.51**

# Software Design or Research Methodology Design

The chapter describes expanded technical details of your software and/or research design that you have further elaborated your initial design, which was drafted in the interim report.

It is crucial that you should have got very clear ideas and the details of what you are going to implement in the next stage of your project process.

There should be a clear presentation of the methodological transitions from the requirement specification from the previous chapter to the **detailed** designof **your software system,** which will be implemented; **OR your investigation datasets, models, experiments**, which will be realised/conducted in the implementation stage.

The chapter’s contents and sections’ headings *may vary depending on the specifics of your project*.

|  |  |
| --- | --- |
| EITHER | OR |
| **For a project of the software development type, the following sections should be included.** | **For a project of the investigative type the contents below can be used as the starting points. Please discuss with your supervisor for further guidelines.**  *Recommended reading*: Section 5.5 Methodology Chapter, page 83 from *Breach M., 2008, Dissertation Writing for Engineers and Scientists.* |
| **Software Design**  4.1 User-Interface design  e.g. using mock-up screens  4.2 Database tables’ structure design  4.3 Main components of the software architecture  4.4 Detailed software class design  i.e. the public interfaces of your software classes, including class public properties and methods without implementation.  4.5 Use cases realisation  For each use case there should be an Activity diagram, or sequence diagram or a class collaboration diagram, which represent the dynamic (i.e. processing) aspect its execution. | **Research Methodology Design**  4.1 Application of chosen theories/methods  4.2 Description of the required sample datasets/experiments  4.3 Data analysis models  e.g. which statistical models, or neural network models etc.  4.4 How the results will be analysed |

# Implementation and Testing or Research Results and Result Analysis

## Software Implementation or Research Results

The contents of this section should cover the following: -

* Details of how you went about realising the specified design from the previous chapter. There should also be your consideration of alternative implementations and evidence of a justified reason for the implementation selected.

A specific, evidence-based description of the techniques, programming languages, software tools which were used, and how they were employed in the implementation and/or in the experiment and/or data analysis.

A detailed technical description of the implemented software and/or research results should be described in this section.

**If your project is of the investigative type** the recommended reading below would be useful.

*Recommended reading*: **Section 5.6 Results Chapter**, page 84 from *Breach M., 2008, Dissertation Writing for Engineers and Scientists.*

## Software Testing or Analysis of the results

This section details how you set about proving:

(a) that the system/program/methodology works and

(b) that the implemented product/research results meet the requirement specification as set out at in Chapter 3 of the report.

Both above points are important and must be clearly and objectively demonstrated.

**If your project is of the investigative type** the recommended reading below would be useful.

*Recommended reading*: **Section 5.7 Analysis Chapter**, page 86 from *Breach M., 2008, Dissertation Writing for Engineers and Scientists.*

**If you developed a software,** there must be a test plan and test results. The **test plan** should provide a clear explanation of how the system was tested, and evidence which proves the correlation between the final implementation and the initial specification. The **presentation of the test results** should include evidence of testing sessions using e.g. computer screen shots, evidence of thoughtful consideration being given to the results of any tests undertaken, and evidence for how much of the solution was implemented and working correctly.

For software testing please read Chapter 17: Testing from the book “*Software Development with UML”* by Lunn. The book is accessible from the live Reading List on the CS6P05 Weblearn space.

# Evaluation of Results

This chapter is to assess whether the project output and results have achieved the project’s aim and objectives, or have solved the stated research problem, putting the project *in the wider context* of the chosen topical area of the project. This could be done by a comparison of your product & results with closely related products and/or related work on the same topic / subject area.

If not, then what the remaining or further work could be.

# Conclusions

In this chapter you should review what you have achieved. Comment on how much of what you set out to achieve you have done. You should review anything which didn’t work and anything you set out to achieve but didn’t highlighting the reasons for that. Finally, you should consider what, if anything, you would do differently if you were to start the project again, and how the project might be taken further by a developer/researcher building on your work.

## A summary of what has been achieved in the project

Unique features of your software product, or your research findings; and its significance.

## Reflections and lessons learned

Reflection upon personal development, legal, social, ethical and professional issues (**LSEP**).

## Future work

Work that enhances your software product or research results

New directions for future work, which emerge from your achieved results.

# Appendices

*Appendices are for including specific data sheets of a component of the project work that may not readily be available and its inclusion in this report is necessary, such as List of Data, Images, Program code listing, screen shots etc.*

*ONLY include them in the report if it is needed for the reader to understand the discussion in the report.*

## Appendix A: Project Management

### 1.1 The original project plan from the Proposal

Below is an example of a table with a reference, which will automatically be pickup by the Word generator of the List of Tables

**Table 1: The project WBS activities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Activities** | **Estimate Duration** | **Activity Description** |
| 1 | Literature Search | 2 weeks | Search, skim-read and filter out 5-7 relevant authoritative published sources on the project topic. |
| 2 | Literature review | 3 weeks | Scan-read, critical review of the selected publications. |
| xxx | ~~~~ | ~~~~~~ | ~~~~~ |
| zzz | Finalise the report | 2 weeks | Finalise, spell-check, format the report and get it proof-read. |
|  | ***Total duration*** | ***27 weeks*** |  |

### 1.2 Review of your project process

Ahead/On target/Behind the schedule

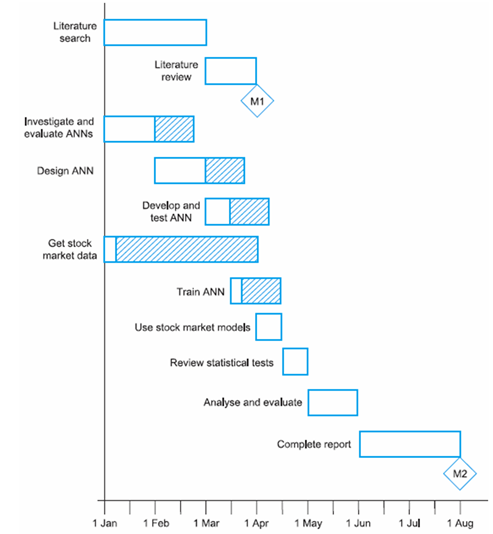
### 1.3 Amendments to the original plan

###### 1.3.1 Remedial actions should be included if you’re behind the schedule

The student should look at how far they have got, and where they said they would be according to their project plan. Are they ahead of schedule, on target, or behind. If they are behind, they must produce a remedial action plan to enable them to get back on target and still bring the project in on time.

A revised project plan with a revised Gantt Chart should be provided.

Below is an example of a figure with a reference, which will automatically be pickup by the Word generator of the List of Figures.



**Figure 3: The revised project Gantt chart**

### 1.4 Lessons learned in project management

## Appendix B: Sample dataset / Survey responses

# References

***Notes***

* *All references used should be referred to from within in the body text and listed here.*
* *Harvard style is recommended;*

Author’s Surname, Initials. (Year of publication). Title. Edition (if not the first). Place of publication: Publisher.

For example: -

Adair, J., (1988). Effective time management: How to save time and spend it wisely, London: Pan Books.

Dawson, C. (2015). Projects in Computing and Information Systems - A Student Guide, 3rd Edition. Harlow: Pearson Education Ltd.

Jalloul, G (2012). UML by Example. Cambridge: Cambridge University Press.

Fisher, R., Ury, W. and Patton, B. (1991) Getting to yes: Negotiating an agreement without giving in, 2nd edition, London: Century Business.

*Book with an editor*Danaher, P. (ed.) (1998). Beyond the ferris wheel, Rockhampton: CQU Press.

*Books with an anonymous or unknown author*The University Encyclopedia (1985). London: Roydon.

*Journal article*Muller, V. (1994). ‘Trapped in the body: Transsexualism, the law, sexual identity’, The Australian Feminist Law Journal, vol. 3, August, pp. 103-107.

*World Wide Web page*Young, C. (2001). English Heritage position statement on the Valletta Convention, [Online], Available: http://www.archaeol.freeuk.com/EHPostionStatement.htm [24 Aug 2001].

# Bibliography

***Note****: The items listed in this section are relevant publications you read and found useful but have not been cited or referred in the report’s body text.*

*Harvard style is recommended.*

Author’s Surname, Initials. (Year of publication). Title. Edition (if not the first). Place of publication: Publisher.

*Examples:*

Bennett, McRobb, Farmer. (2010) Object-Oriented systems analysis and design using UML, 4th edition. London: McGraw-Hill.

Breach, M. (2008). Dissertation Writing for Engineers and Scientists. Harlow: Pearson Education Ltd.

Connolly, T., Begg, C. (2014) Database Systems- A Practical Approach to Design, Implementation, and Management (6th ed.). Upper Saddle River, N.J: Pearson.

Dawson, C. (2015). Projects in Computing and Information Systems - A Student Guide, 3rd Edition. Harlow: Pearson Education Ltd.

De Oliveira , J., Bruchet M. (2017). Learning ASP.NET Core 2.0. Birmingham: Packt Publishing.

Hughes, B. (2012). Project Management for IT-related Projects. London: BCS Publications.

Jacobson, Booch and Rumbaugh. (1999) The Unified Software Development Process. Upper Saddle River, NJ: Addison-Wesley.

Jalloul, G (2012). UML by Example. Cambridge: Cambridge University Press.

Müller M. (2018). Practical JSF in Java EE 8 : Web Applications in Java for the Enterprise. New York City: Apress.

Pfleeger, C. P. et al. (2015). Security in Computing, 5th edition. Harlow: Pearson.

Rothman, D. (2018). Artificial Intelligence By Example: Develop machine intelligence from scratch using real artificial intelligence use cases. Birmingham: Packt Publishing.

Sommerville, I. (2016). Software Engineering, 10th Edition. Harlow: Pearson

Stallings, W. and Brown, L. (2018) Computer Security: Principles and Practice, Global Edition, 4/E. Harlow: Pearson Education.

Turban, E., Sharda, R., Delen, D., 2010, Decision Support and Business Intelligence Systems, 9th edition. Upper Saddle River, N.J: Pearson.

Welling L. and Thomson, L. (2017). PHP and MySQL Web Development, 5/E. Upper Saddle River, NJ: Addison-Wesley.