**A logo with a palm tree

Description automatically generated**

**Project Title**

By

|  |  |
| --- | --- |
| Section No. |  |
| Student Name | University ID |
| Student Name | University ID |
| Student Name | University ID |
| Student Name | University ID |

Supervised by

***Name of the Professor/Dr/Mr …***

Submitted to King Khalid University in fulfilment of the requirements for the B.Sc. Degree in Computer Science

**Academic Year 2024-25(Semester I)**

**Department of Computer Science**

**College of Computer Science**

**King Khalid University**

Comments by Supervisor(s) / Examiner(s) Name signature of Supervisor and examiners

|  |  |  |
| --- | --- | --- |
| Name | Comments | Signature |
|  |  |  |
|  |  |  |
|  |  |  |

Abstract

The abstract is a concise and accurate summary of the research contained in the report. It describes the subject matter of the report, states the problem, the methods of investigation, the main findings and the general conclusions, and should not contain tables, graphs or illustrations. It must not exceed 350 words. There must be a single abstract for the entire project.

Your abstract should summarize the subject matter of the report, the problem being solved, the motivation for solving it, the approach taken, your findings and conclusions.

The abstract normally takes several revisions to achieve a good abstract.

Font size for text should be 10 –12 point and in Times New Roman.

Microsoft Word can give you an accurate word count of your abstract:

Declarations

Declarations regarding the originality etc of the thesis should go here…

Acknowledgements

Thanks to Allah SWT and you may acknowledge help from friends and colleagues, from your supervisor, staff, department and university, and from your parents and family.

Acknowledgment may extend for more than one page, but should be no longer than two pages.

Table of Contents

[Abstract 4](#_Toc125876355)

[Declarations 5](#_Toc125876356)

[Acknowledgements 6](#_Toc125876357)

[Table of Contents 7](#_Toc125876358)

[Table of Figures 9](#_Toc125876359)

[List of Tables 10](#_Toc125876360)

[Chapter 1: Introduction and Background 11](#_Toc125876361)

[1.1 Goal 11](#_Toc125876362)

[1.2 Motivation 11](#_Toc125876363)

[1.3 Signification 11](#_Toc125876364)

[1.4 Method 11](#_Toc125876365)

[1.5 Report Outline 11](#_Toc125876366)

[Chapter 2: LITERATURE REVIEW 12](#_Toc125876367)

[2.1 Subsection 1 12](#_Toc125876368)

[2.2 Subsection 2 12](#_Toc125876369)

[2.3 Problem Statement 12](#_Toc125876370)

[2.4 Research Questions(s) or Hypothesis(es) (optional) 12](#_Toc125876371)

[Chapter 3: PROJECT MANAGEMENT 13](#_Toc125876372)

[3.1 Approach 13](#_Toc125876373)

[3.2 Risk Management 13](#_Toc125876374)

[3.3 Project Plan 13](#_Toc125876375)

[Chapter 4:Requirements and Analysis 14](#_Toc125876376)

[4.1 Requirement Elicitation 14](#_Toc125876377)

[4.2 System Requirements 14](#_Toc125876378)

[4.3 Personas 14](#_Toc125876379)

[4.4 System Models 14](#_Toc125876380)

[Chapter 5: System Design 16](#_Toc125876381)

[5.1 Product Features 16](#_Toc125876382)

[5.2 User Interface 16](#_Toc125876383)

[5.3 Data Storage 16](#_Toc125876384)

[5.4 High Level Design 16](#_Toc125876385)

[5.5 Detailed Design 16](#_Toc125876386)

[5.6 Design Verification 16](#_Toc125876387)

[Chapter 6: PROJECT IMPLEMENTATION 17](#_Toc125876388)

6.1 [Coding 17](#_Toc125876389)

[6.2 Verification 17](#_Toc125876390)

[6.3 Validation 17](#_Toc125876391)

[Chapter 7: PROJECT TESTING (EVALUATION) 18](#_Toc125876392)

[7.1 Metrics 18](#_Toc125876393)

[7.2 Experimental Setup 18](#_Toc125876394)

[7.3 The Experiments 18](#_Toc125876395)

[7.3.1 Participants 18](#_Toc125876396)

[7.3.2 Treatments 18](#_Toc125876397)

[7.3.3 Procedures 18](#_Toc125876398)

[7.3.4 Data Analysis 18](#_Toc125876399)

[7.4 Results 18](#_Toc125876400)

[7.5 Validity 19](#_Toc125876401)

[7.6 Summary of Findings (Discussion) 19](#_Toc125876402)

[Chapter 8: USER MANUAL (if applicable) 20](#_Toc125876403)

Chapter 9: [Conclusions and Future Work 21](#_Toc125876404)

[9.1 Limitations (i.e., limitations or problems with the study or outcomes) 21](#_Toc125876405)

[9.2 Future Work 21](#_Toc125876406)

[9.3 Summary 21](#_Toc125876407)

[REFERENCES 22](#_Toc125876408)

[APPENDICES 24](#_Toc125876409)

Table of Figures

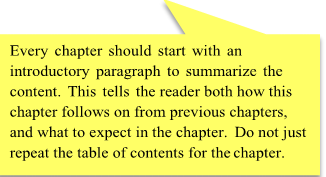
[Figure ‎2.1: Speech Quality Versus Bit Rate for the three Classes of Codecs [18] 12](#_Toc374812852)

List of Tables

[Table ‎3.1: The table caption goes here. 13](#_Toc374812858)

# Chapter 1: Introduction and Background

The introduction to the project report must outline the problem statement, hypotheses and/or goals of the project and provide sufficient information to enable a non-specialist reader to understand these. Say what it was about, give some brief background information (sufficient to ‘set the scene’) and list your project objectives. You should use your own words to provide a description of the problem domain, scope and why it is important. Your introduction should contain the elements listed below.



## Goal

State what to be achieved in the project

## Motivation

Describe why the problem is worth solving.

## Signification

State the significance of your project and why your project is important.

## Method

Summarize how the project was carried out

## Report Outline

Briefly outline the remainder of this report in a paragraph..

This thesis is organised as follows.

**Chapter 2**provides background information…

**Chapter 3**

**Chapter ‎4**

**Appendix I** introduces ….

**Appendix II** provides ….

# Chapter 2: LITERATURE REVIEW

This chapter should include the theoretical background on which the project is based. You should organize this in some way to show your understanding. Make good use of proper references to cite the papers, books, and other material you have read. Your literature review should show that you understand the area that your project work will be in, both from a technical viewpoint, and from a viewpoint of relevant research in the area.

Students should bare the following points in mind when preparing this chapter:

1. The background material should include descriptions and quotes of the major topics that are discussed in the report.

2. It is preferable to always link the background material with what is done in the project.

3. Students should avoid copying of materials that they quote from other sources, but try to summarize and paraphrase them.

4. All quoted materials should be cited appropriately.

5. You may assume that your readership has the level of knowledge of a good Information System student who has taken the same modules as you. Bear this in mind when writing about background technical information and do not present large amounts of information that such a reader would already know or that could be read in a standard textbook. Simply reference the textbook in your bibliography and keep the information you present specific to your own work.

Explain how any background material you present has been used in your project.



## Subsection 1

## Subsection 2

## Problem Statement

… Clearly state the problem that you are attempting to solve in your project

## Research Questions(s) or Hypothesis(es) (optional)

State your hypothesis or questions that you are trying to study in your project

# Chapter 3: PROJECT MANAGEMENT

Introductory paragraph – here where you discuss how you planned the project, and why you planned it the way you did



## Approach

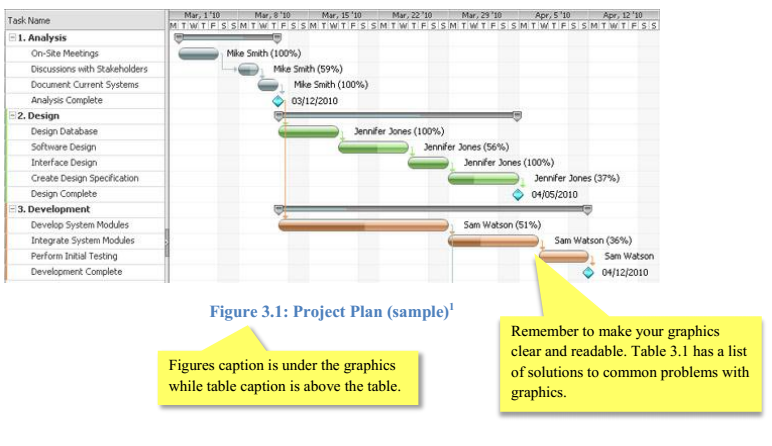
Discuss the approach you have taken in planning for your project (Methodology/Process Model) and the software process model which is also known as Software Development Life Cycle model. It introduces the selected model for the development process with justifications

## Risk Management

This section identifies the potential risks that may be faced during the project and the necessary plans to minimize them

## Project Plan

Show your project plan in the form of a Gantt Chart and a brief description, showing planned dates. If your task names are not self-explanatory, provide a table to explain the tasks.



# Chapter 4:Requirements and Analysis

Introductory Paragraph – this is where you describe and analyse the problem in detail This chapter should list the project requirements and the detailed analysis of such requirements. The following bullets emphasize the important aspects of this chapter:

1. If surveys and/or questionnaires were used in the requirements’ collection phase, then detailed analysis of the results should be discussed. Forms used and data samples should be included in one of the appendices and referenced in the chapter.

2. If interviews were conducted as part of the requirements’ collection phase, then sample questions and interviewee answers should be included in one of the appendices. Conclusions from such interviews should be outlined and discussed in this chapter (Chapter 4).

3. One or more of the analysis methodologies can be used in this chapter. Students should check with their supervisor on preferred methodologies (if any) to use.



## Requirement Elicitation

This section describes the methods of collecting the system requirement such as interviews, questionnaires, observations and the quantitative and qualitative data analysis.

Breakdown the problem into the different “user tasks” that the “user” needs to perform in order to “solve” their problem. Summarize the tasks here, perhaps describing a few key ones in more detail, and identifying the inputs and outputs for each. You might find it useful to include a summary “Use Case” diagram.

## System Requirements

This section describes the behavior and features of software applications which consists of both functional and non-functional requirements.

## Personas

// This section describes the characteristics of the system/application users.

## System Models

// This section describes different system models, which can be implemented either using the traditional approach or Object-Oriented (OO) approach as shown in Table 1

|  |  |
| --- | --- |
| Traditional Approach | OO Approach |
| * + 1. **Data Flow Diagram**   // This section describes the data flow diagram which explains how data is processed and transferred in a system. | **4.3.1 UML Use Case Diagram**  // This diagram describes the system’s boundaries and functionalities from the user point view. |
| * + 1. **Process Specification**   // This section describes the method used to document, analyze and explain the decision making logic and formulas used to create output data from process input data.  // This section may include also the description of the “Structured English” which details each step in the running process of the program. | **4.3.2 Process Specification**  // This section describes the use case descriptions and scenarios. In addition to,  UML sequence, state and activity diagrams for important processes should be included. |
| * + 1. **Entity Relationship Diagram**   // You illustrate here the information of a system’s entities and relationships between those entities. | **4.3.3 UML Class Diagram**  // This diagram describes the static structure of the system in term of classes. All relationships among classes should be represented. |

# Chapter 5: System Design

Introductory paragraph – this is where you describe your solution to the problem. This chapter should document the detailed design of the project. Discuss how you designed the software, identify different design options, and justify why you picked the one used.

Students are advised to select (with the approval of their supervisor) one of the known design methodologies. The chapter should also include tables and diagrams to clarify the design options related to the project.



## Product Features

Identify and summarize the key features. For a GUI these are the main commands that the user can give. For a networked system, these are the main protocol messages accepted. For an embedded system these are the main external events that the system must respond to.

## User Interface

For each product feature, state what each user interface component must do (state the inputs, actions, and outputs). Include a detailed design for each screen in the Appendix.

## Data Storage

Summarize what data must be placed in persistent store within the system, and what relationships must be maintained.

## High Level Design

Summarize the top level objects and their relationships (e.g. UML Class Diagrams)

## Detailed Design

Summarize the object contents (e.g. Javadoc specifications for Methods and Attributes)

## Design Verification

Summarize how your system design satisfies the user requirements. One way is to "walk through" a few key user tasks, using Sequence Diagrams to show how the combination of features allows the user to achieve the task.

Include full design details (see Project Design Specification) in the Appendix

# Chapter 6: PROJECT IMPLEMENTATION

This chapter describes the project implementation. Here where you discuss your choice of language, tools and platform.



## Coding

Where not obvious, discuss how you translated the detailed design into code (for example: how you decide to code association classes and two-way associations).

## Verification

Summarize how you tested the code (unit test and system test). Discuss the techniques you used to select test cases and test data. Summarize the test results, showing that the software satisfied the detailed design.

## Validation

Summarize how you showed that the solution satisfies the user requirements. This normally consists of manual execution of each user task.

Include full source code and testing details (see Software Testing Document and Software Validation Document) in the Appendix.

# Chapter 7: **PROJECT TESTING (EVALUATION)**

Introductory paragraph - here where you describe how you test and evaluate your product/software and how the results meet your expectations or hypotheses. This chapter describes the components for an evaluation/test plan and the output data when conducting such test. An evaluation plan states the objectives of the evaluation, the questions that will be answered, the information that will be collected to answer these questions, and when collection of information will begin and end (timeframe). You should also summarize your main findings/results and evaluate what you have achieved and how you went about it. What is crucial is to have a critical self-evaluation of the extent to which you have achieved the things you set out to do.

Assess the extent to which you met your objectives. You will not be penalized for acknowledging that you failed to achieve everything you set out to do, and especially not the more advanced things, but you certainly would be criticized if you gave the impression of not having noticed that you had failed to meet an objective.



## Metrics

Identify and describe the metrics you used to evaluate your software.

## Experimental Setup

Describe the experimental setup for each metric, and how you obtained the measurements.

## The Experiments

Describe the inputs for each experiment

### Participants

Describe the demography of your sample.

### Treatments

Describe the tasks that are taken in your experiment

### Procedures

Describe how the experiment was done

### Data Analysis

Describe the metrics you used to analyze your data and which type of analysis you performed.

## Results

Summarize the output data, and the statistical or other techniques to deduce your results.

Summarize your results, including tables or graphs as appropriate with a brief description

of each. Where possible, compare your results with other products/systems.

## Validity

Identify any possible threats to the validity of your results, and discuss each.

## Summary of Findings (Discussion)

Summarize your results and findings and how they support your hypotheses. This should

include:

* Explanation of Outcomes
* Implications of Results
* Solution Review: Discuss how well your solution solves the problem, based on your results.
* Project Review: Discuss how well you addressed the project, and what you might do differently if you were to do it again. In particular, make sure to identify how you handled any problems that arose during the project.
* Key Skills: Identify key skills that you learnt during the project, and clearly describe how you applied these, and how you might apply them differently if you were to do a similar project.
* Summary of Discussion

Include detailed data and results in the Appendix

# Chapter 8: **USER MANUAL (if applicable)**

This chapter should pictorially (using screen shots) guide the user on how to use the finished system. The inclusion of this chapter is essential since it acts as a contract between developers (the students) and the customer (the supervisor and/or the actual customer) on what the finished system should look like. The screen shots (or diagrams) should be a rough sketch on what the students think the finished system will behave. The users' manual should include simple and easy to follow instructions on how to use all the functions that are outlined in the requirements list.

# Chapter 9: Conclusions and Future Work

Introductory paragraph, here where you summarize the project and your solution. This chapter summarizes the parts of the system that are completed and the parts that are still pending. This chapter may include the following:

* overall analysis and integration of the research and conclusions of the project in light of current research in the field
* conclusions regarding goals or hypotheses of the project that were presented in the Introduction, and the overall significance and contribution of the project
* comments on strengths and limitations of the project
* discussion of any potential applications of the findings
* an analysis of possible future research directions in the field drawing on the work of the project



## Limitations (i.e., limitations or problems with the study or outcomes)

Discuss any limitation or problems with the study or project.

## Future Work

Discuss any proposals for completion of the project, or for enhancements, or for re­design of your solution or software.

## Summary

Summarize your project in one or two paragraphs.

# REFERENCES

* 1. I. U. School of Informatics and Computing, "Thesis Formatting and Style," [Online]. Available: <https://soic.iupui.edu/files/Thesis_Formatting_and_Style_Document.pdf>. [Accessed 15 June 2015].
  2. U. o. B. C. Graduate and Postdoctoral Studies, "DISSERTATION & THESIS PREPARATION," [Online]. Available:

<https://www.grad.ubc.ca/sites/default/files/materials/thesis_review.pdf>. [Accessed 21 June 2015].

* 1. M. U. Department of Information System, "Final Year Project Handbook," 2012. [Online]. Available: <http://www.cs.nuim.ie/sites/default/files/FYProjectHandbook-> 2012-2013-final.pdf. [Accessed 5 July 2015].

Include a list of any source (books, journals, web pages etc.) that you cite in your report. You should also list any sources that you have used, even if not cited directly. Use the [numbered] convention. You should include here all references to materials that you obtained from outside sources. You should also list these references using some known citation standards, such as IEEE's style. For your references, produce one list, ordered by author surname (whether the material is drawn from books, journals, forums or blogs, or is a piece of software).

In Information System publications, a citation will normally take the form of a number in square brackets, such as [1]. An alternative form is to use author's name and a year in square brackets, such as [Brown 2010]. In your references, you must have a list of these, and then the full details for each so that the reader can look it up - you must include the publication title, author, year and publisher. Note that just a URL is not acceptable as the only content.

Book: Author(s). Book title. Location: Publishing company, year, pp.

Example: W.K. Chen. Linear Networks and Systems. Belmont, CA: Wadsworth, 1993, pp. 123-35.

Article in a Journal: Author(s). "Article title". Journal title, vol., pp, date.

Example: G. Pevere. "Infrared Nation." The International Journal of Infrared Design, vol. 33, pp. 56-99, Jan.1979.

Articles from Conference Proceedings (published): Author(s). "Article title." Conference proceedings, year, pp.

Example: D.B. Payne and H.G. Gunhold. "Digital sundials and broadband technology," in Proc. IOOC-ECOC, 1986, pp. 557-998.

Electronic Journal: Author. (year, month). "Article title." Journal title. [Type of medium]. Vol. (issue), pages. Available: site/path/file [date accessed].

Example: A. Paul. (1987, Oct.). "Electrical properties of flying machines." Flying Machines. [On-line]. 38(1), pp.778-998. Available: [www.flyingmachjourn/properties/fly.edu](http://www.flyingmachjourn/properties/fly.edu) [Dec. 1, 2003].

World Wide Web: Author(s)\*. "Title." [Online]. Available: complete URL, date updated\* [date accessed].

Example: M. Duncan. "Engineering Concepts on Ice." [Online]. Available: [www.iceengg.edu/staff.html](http://www.iceengg.edu/staff.html), Oct. 25, 2000 [Nov. 29, 2003].

**Special Note on Plagiarism**

**Presenting the work of others as your own is not acceptable.**

**A reader must be able to clearly distinguish**

**between your own work and the work of others.**

**If you use the work of others, either as a direct copy, or**

**summarizing or using their ideas, you must (a) cite their work at**

**every place where you make use of it, and (b) include a full**

**reference to their work**

**All projects will be submitted to “Blackboard” to assist the**

**Department in ensuring that your work is original.**

# APPENDICES

The appendices to your report provide supporting evidence of the quality and quantity of the work you have done. All appendices in the project must be presented together after the concluding chapter. Page numbering for appendices continues on from the body of the project, in sequence.

This chapter can include one or more appendices. Each appendix should talk about one single subject that is discussed lightly within the report chapters.

Appendices must be limited to supporting material genuinely supplementary to the main argument of the project. Appendices must only include material that is referred to in the body of the project.

The following are appropriate for inclusion in the appendices:

* additional details of methodology and/or data
* diagrams of specialized equipment developed
* copies of questionnaires or surveys used in the research
* Source Code (including all scripts, makefiles, XML, HTML, etc.)
* If possible, include a compiled executable with running instructions
* Reading reports
* Interim Progress Reports
* Project Plan Document
* User Requirement Document (Development Project) or Theoretical Development (Research Project)
* Product/System Design Document
* Software High-Level Design
* Detailed Software Designs (for example, if using Java, include the Javadoc output files here)
* Software Test Documents (tests & test results)
* Software Evaluation Results
* Any additional outputs that you created during the project

Without this supporting evidence it is possible that the examiner will take the view that you have not done everything you claim to have done. However, the appendices are only there to back up the claims made in your report. Examiners can only be expected to look at those parts of the appendices you draw their attention to in the main body of the report. They are not obliged to read the appendices in detail, though they may do so. If you think it is important to draw the markers' attention to a particular document, or a part of a document, tell them where to find (don't just say "the code for this is in appendix 3", give a page number, and/or other information that makes it clear how to find it). All program code written by you must be presented in the appendices. Do not include code that is machine generated, or that comes from a different author, unless it is necessary in order for the reader to understand the work you have done. If you do include code that you did not write yourself, it is your responsibility to make clear which parts of the program are your own and which parts are not. If you present automatically generated code, or the code of another programmer, as if it were your own, you may be accused of plagiarism. Do not include copies of any web pages that you have referred to, unless it is absolutely necessary for the reader to see them in order to make your point: just put the citation details in your bibliography. Samples of the work that is presented in the appendices may be included in the body of your report in order to illuminate a particular point or for discussion purposes.

**1. Appendix A: Project Plan**

**2.** Appendix B: User Requirements Specification**3. Appendix C: Project Design Specification**

**4. Appendix D: Software Testing Document**

**5. Appendix E: Software Validation Document**

**6. Appendix F: Experimental Data and Results Document**

**7. Appendix G: POSTER**

# This is an outline report structure. Many projects will vary from this. Consult with your project supervisor to create the most appropriate structure for your report*.*