ISRA UNIVERSITY

Software Engineering Documentation

A project submitted in partial fulfillment of the requirements for the degree of Bachelor in Software Engineering

By

Student's First Name Middle Initial Last Name (Student ID)
Student's First Name Middle Initial Last Name (Student ID)
Student's First Name Middle Initial Last Name (Student ID)

Supervised by:

Supervisor's First Name Middle Initial Last Name
Co-Supervisor's Name (if any)

Committee Member Names:

Committee Member's First Name Middle Initial Last Name
Committee Member's First Name Middle Initial Last Name

Month, Year

Acknowledgements

The acknowledgement is a statement of gratitude for assistance to accomplish the project. It may mention the names of the people the project members want to thank for their support in the project (usually parents, friends, instructors).

UNDERTAKING

Abstract

An abstract can be either descriptive or informative. A descriptive abstract summarizes the motivation, scope and methods used to attain the solution or findings. An informative on the other hand, is almost like the table of contents written in paragraph. It also includes the results, conclusions and recommendations. The abstract should not exceed 300 words and its contents are italicized[1].

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter comprises background of the project, the reasons for taking it, problems addressed by the project and expected outcomes. A good report starts with an introduction to the title of project. The necessary background information is provided to establish context of the project. The motivation and significance of the project should be highlighted. A crisp problem statement is followed by scope of the project along with any limitation or exclusions. The specific objectives to be achieved should be stated. A road-map or organization of report concludes the chapter.

1.1 Overview

In this section, you should write about the general review or summary of this project.

1.2 Project Motivation

In this section, you should write about the answer the following questions:

- Q1. What is the reasons behind your choice to develop this project?
- Q2. Why your project is important?
- Q3. What is the new idea that have been proposed by this project?

1.3 Problem Statement

write about the issues that have been addressed by this project and the conditions to be improved upon.

1.4 Project Aim and Objectives

Write about the overall purposes of this project, should be clearly and concisely defined. In this section you should answer the following questions:

- Q1. What is the goal that this project wants to achieve?
- Q2. How this project can achieve this goal?

1.5 Project Scope

Explains the boundaries (specified features and functions) of this project, establishes responsibilities for each team member and sets up procedures for how completed work will be verified and approved.

1.6 Project Software and Hardware Requirements

List the prerequisites software and hardware requirement of this project.

1.7 Project Limitations

you should clarify the limitations or parameters of the project and clearly identify any aspects that are not to be included.

1.8 Project Expected Output

Describe the desired results of the project.

1.9 Project Schedule

Listing of a project's milestones, activities, and deliverance's, with intended start and finish dates.

1.10 Project, product, and schedule risks

Describe the risk that the project takes longer than scheduled.

1.11 Gantt Chart

1.11.1 Analyze and Requirement

1.11.2 Screen Design and Data Base

1.11.3 Algorithms Design

1.11.4 Testing

1.11.5 Document

1.11.6 Graph

Chapter 2

Related Existing System (Optional)

2.1 Introduction

2.2 Existing Systems

2.3 Overall Problems of Existing Systems

2.4 Overall Solution Approach

Chapter 3

Requirement Engineering and Analysis

3.1 Elicitation

3.2 User Requirement

3.3 Analysis

3.3.1 Functional requirement

3.3.2 Non-Functional Requirement

3.3.3 System requirement

3.4 Stakeholders

List the individuals, groups, or organizations, who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of this project. And specify the type of each stockholder (e.g. Primary stakeholders, Secondary stakeholders, etc.).

3.5 Use Case Diagram

3.5.1 Use Case Section

Normal Flow for each use case including action, precondition, post-condition and other sections as you learnt in requirements engineering course.

3.5.2 Activity Diagram

3.5.3 Alternative flows

An alternate flow describes a scenario other than the normal flow **for each use case**.

3.6 Non-functional requirements

Specify the non-functional requirements of this project that can be divided into two main categories:

1. Execution qualities, such as safety, security and usability, which are observable during operation (at run time).

2. Evolution qualities, such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the system.

3.7 Constraints

List the conditions and restrictions of this project that must be satisfy.

Chapter 4

Architecture and Design

4.1 Overview

4.2 Software architecture

4.2.1 Logical view

Provide the software-architecture logical view for the major components as UML component diagram (or class diagram).

4.2.2 Process view

Provide the software-architecture process view for the major components as UML sequence diagram (or communication diagram).

4.2.3 Physical view

Provide the software-architecture physical view as UML deployment diagram.

4.2.4 Details of each component in a separate section.

4.3 Software design

4.3.1 UML sequence/communication diagram

Provide UML sequence/communication diagram for each use case scenario. You should show concurrent objects and the messages type (i.e. synchronous or asynchronous).

4.3.2 Class diagram

Provide class diagram to show classes' relationship, internal classes data, and methods. This should be based on the use case scenarios, problem description, and use case scenarios sequence/communication diagrams.

4.3.3 Normalization

1NF Normal Form

2NF Normal Form

3NF Normal Form

BCNF Normal Form

4NF Normal Form

4.3.4 ER diagram

Provide the ER diagram for your data structure in the database (if any).

4.3.5 State transition diagram

Provide state transition diagram of the system and if needed for some components/classes

4.4 User interface design

Provide snapshots for the graphical user interface screens of the system.

Chapter 5

Implementation Plan

This subsection of the Project Implementation Plan describes Solution in more details. Describes how the information system will be deployed, installed and transitioned into an operational system. It contains a brief description of the major tasks and components involved in the implementation, the overall resources needed to support the implementation effort (such as hardware, software. facilities, materials, and personnel), and any site-specific implementation requirements.

5.1 Description of Implementation

This section provides a list of programming languages, technologies, software and databases required to support the implementation. Identify them by name, code, or acronym. Identify which software is commercial off-the-shelf and which is State-specific. Identify any software used to facilitate the implementation process.

5.2 part of implementation if possible

Provide pieces of code for major tasks and components.

Chapter 6

Testing Plan

6.1 Black-box

Provide the black-box techniques that are used to test this project including test cases.

6.2 White-box

Provide the white-box techniques that are used to test this project including test cases (test case if code is available).

6.3 Testing automation

This section should provide:

1. The automation tools that have been used to control the execution of tests and the comparison of actual outcomes with predicted outcomes.
2. Decide what test cases to automate

Chapter 7

Conclusion and Results

The conclusion is a required part that closes the document with a brief summary of the study including the problems found and the proposed solution. Most importantly, it should recommend to the readers the benefits of pursuing the project based on the researcher's analysis.

References

[1] D. E. Knuth, "Literate programming," *The Computer Journal*, vol. 27, no. 2, pp. 97–111, 1984.