FACULTY OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA

THE SIMULATION OF PROJECT MANAGEMENT - QUALITY OF SOFTWARE DEVELOPED

By
HENG AI SHI
(WEK010092)

Supervisor: Assoc. Prof. Dr. Ow Siew Hock
Moderator: Ms. Su Moon Ting

A thesis presented to the Faculty Computer Science and Information Technology of University of Malaya in partial fulfillment of the requirement for the Degree of

BACHELOR OF COMPUTER SCIENCE
UNIVERSITY OF MALAYA
2003/2004

Abstract

The Simulation of Project Management is a stand-alone application that aims to prepare future project leaders by providing simulated project management experience and be used by students who are taking Project Management course to test their knowledge on this course and therefore increase their interest and skill in handling project in future.

Basically, the Simulation of Project Management will inform the results to the player after the simulation end from the following perspectives: project schedule, project cost, resource management and quality of the software developed by the project team members. The Simulation of Project Management is divided into 8 main modules which are: Overall Project Schedule, Team Organization, Manpower/Budget Schedule, Team Member Schedule, Project Leader Schedule, Module Design/Coding/Module Test Progress Table, Integration Test Progress Table and Simulation Result Screen.

This system was developed on Microsoft Windows XP Professional platform using Microsoft Visual Basic.NET with Macromedia Flash MX supported. Besides, the Database Management System (DBMS) for the Simulation of Project Management is Microsoft Access 2000, which is the most popular database nowadays.

Literature reviews on all topics involved have been conducted to determine the feasibility of the system. All these activities are aimed at achieving the best outcome, which is to create a comprehensive Simulation of Project Management to fulfill the needs and satisfaction of the player.

Acknowledgement

Firstly, I world like to express my gratitude to Computer Science & Information Technology Faculty (FSKTM), University of Malaya for giving me a chance to do thesis for my final year. FSKTM has provides valuable facilities and commitments to assist me to complete this thesis report successfully and smoothly.

My greatest thank-you and appreciation to Dr. Ow Siew Hock, my remarkable supervisor for whom I respect and salute to the utmost degree for providing me the golden opportunity to do my thesis under her privilege supervision. Your numerous and varied tasks well as guidance has given me plenty self-improvement.

Special thanks to Ms. Su Moon Ting, my thesis moderator for WXES3181 and WXES3182 taking time off attending my VIVA presentations and putting efforts in her valuable query and question to me.

My warmest thanks to Teoh Kean Boon, Tan Chee Hong and Tan Hai Liang, my thesis partner for the ideas and guidance occasionally on how to carry out the thesis. Thanks for sharing your knowledge on project scope, project objective, requirement specification and software design. I do enjoy and appreciate the time we had been together for discussion.

Finally, many thanks to all who have assisted me in during this thesis for your kind support.

Table of Contents

Chapter 1: Introduction

Chapter 2: Literature Review

Chapter 3: Methodology

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Project Background

The project title is Simulation of Project Management. This project is to develop software that simulates project management in software development. The simulation software will inform the results from the following perspectives:

a. Project schedule

b. Project cost

c. Resource management

d. quality of the software developed by the project team members

1.1.1 Project Definition

A project can be through of as the allocation of resources directed toward a specific objective following a planned, organized approach. Beside that, a project is set in time, budget and resource requirements. It is also set in the context of organization, a legal system, a political system, a technology structure, an economic system, and a social system. Figure 1.1 shows the components or parts of project. [Young, Trevor L, 1998]

At the top is a project objective. This is the overall purpose of the project. But objectives are vague. So we need a strategy to support the objective. The strategy identifies how we will achieve the objective of the project. The basic project documentation is the project plan. The project plan lives and breathes and changes as the project progresses or fails. The project plan must define the work to be done, the resources required, the methods to be followed in the project and its management and control, the tools to be used in support of the methods, and the schedule of work. The resources of the project support the structure of the project plan. [Young, Trevor L, 1998]

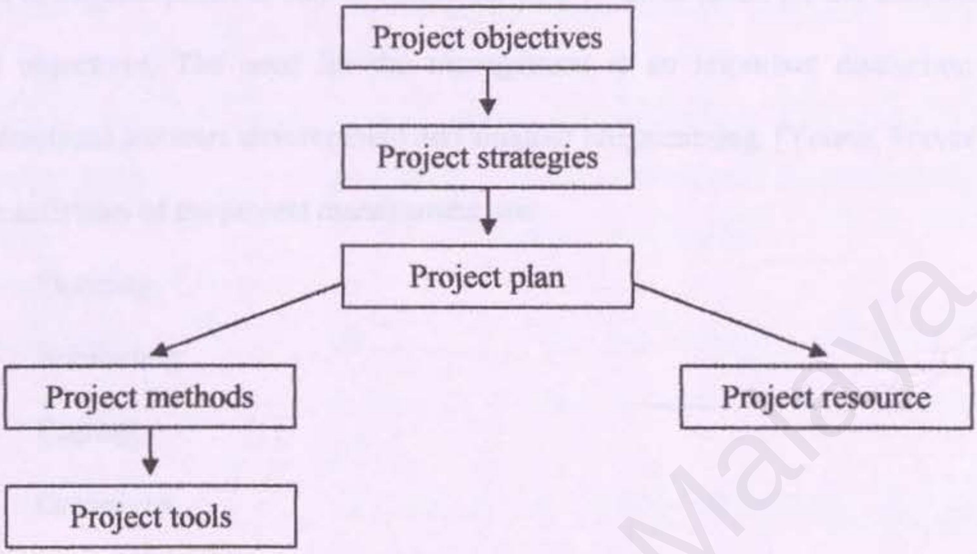


Figure 1.1 Components of project

Most projects can be characterized as:

- Having a specific and define purpose;
- Unique because it will never be repeated in quite the same way;
- Being focused on customer needs and expectations;
- Not being routine but including many routine-type tasks;
- Having defined constraints of time, cost and people available;
- Involving people in different departments and even sites;
- Involving many unknowns and hence many risks;
- Challenging traditional ways of working to introduce improvements;
- Providing an opportunity to learn new skills.

1.1.2 Project Management Definition

Project management can be defined as the actions taken to define the projects requirements, to plan the activities to accomplish the requirements and to implement the plan in the best possible way to ensure the best possible result for the intended purpose and objectives. The need for the management is an important distinction between professional software development and amateur programming. [Young, Trevor L, 1998]

The activities of the project management are:

- Planning
- Scheduling
- Costing
- Organized
- Controlling
- Monitoring
- Reporting

1.1.3 Project Leader

Substantial projects require more than one person to succeed. This group of people is commonly referred to as the project team. So, every project must consist of one project leader to manage the project team member. Being a project leader is a lot of work.

1.1.3.1 Project Leader Characteristics

The characteristics of the project leader will directly affect the outcome and quality of the project. Table 1.1 shows the characteristics of the project leader.

Table 1.1: Characteristics of project leader

Characteristics	Description
Communicator	- Needs to communicate well verbally and in writing within and outside of the project.
Generalist	- Able to see the big picture and then relate it to the current project situation.
Problem and conflict solver	- Able to identify and understand problems, place them in perspective, and then develop and implement solution.
People management	- Able to effectively manage people.
Experience	- Experience in several projects and able take this experience, integrate it, and then apply it to the current project.
Ambition	- The positive side ambition will lead the project leader to work hard and positively and also help him or her in dealing with people and getting the project completed.
Knowledge	- Able to acquire information about all aspects of the project quickly and ability to apply the knowledge in the project.
Perspective	- Able step back from the project and take an overall view.
Initiative and risk taking	- Able to show initiative and willing to take risks.
Able to take direction and suggestions	- Able to admit that there is a problem and accept responsibility. - Able to accept suggestions, hints, and criticism and turn it into positive action.

1.1.3.2 Project Leader Roles

- Define the project
- Prepare the project plan
- Present and sell the project and project plan
- Convey the purpose of the project
- Apply effective project management methods and tools
- Obtain project resources
- Coordinate the use of the project resources
- Interact with the customers and suppliers
- Prepare and review budgets
- Develop schedules
- Analyze issues
- Determine project status
- Delegate responsibilities
- Encourage people to maintain interest and motivation
- Regularly keep everyone informed of progress
- Promote an atmosphere supporting free and willing feedback
- Manage peer and senior groups to influence their support
- Manage third parties such as contractors and consultants
- Manage conflict in the team
- Help the team members to prioritize their workloads

1.1.4 Unforeseen Events of Project Management

During the project development, the project team especially project leader will encounter many unpredictable problems. Therefore, the project needs to do the risk management to avoid the risk or minimize the effect of the risk. The example of the unforeseen events will meet by the project leader, such as:

- Shortage of resources
- Expertise not available
- Weak performance of the team member
- Conflict in the team
- Customer changes the requirements
- Over budget
- Project completion can not meet the deadline

1.1.5 Quality of Software

For software products, there are four factors that can affect product quality. These are shown in Figure 1.2. The influence of each of these factors depends on the size and type of the project. For large project, the major problems are integration, project management and communications. There is usually a mix of abilities and experience in the team members. However, particularly skilled or talented individuals don't usually have a dominant effect over the lifetime of the project. A basic level of development technology is essential for information management. [Sommerville, I, 2001]

For small project, there are only a few team members, the quality of the development team is more important the development process used. If the team has a high level of ability and experience, the quality of the product is likely to be high. Where teams are small, good development technology is particularly important. [Sommerville, I, 2001]

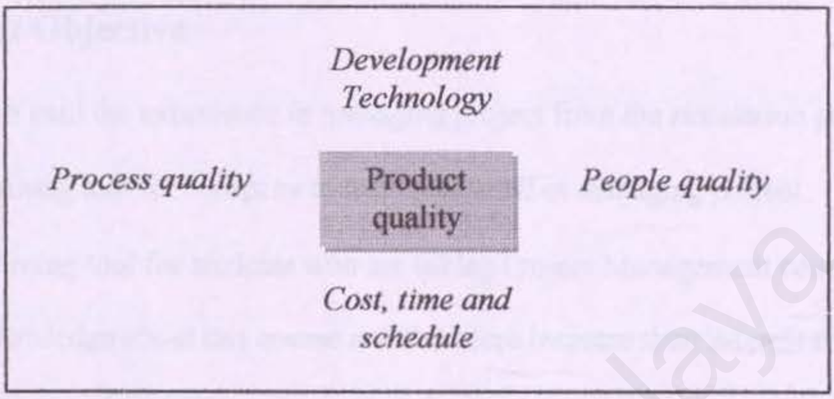


Figure 1.2: Principal software product quality factors

1.2 Project Definition

This project is Project Management Game simulation software. The participants will assume that they as the project leader in managing a simulated software development project. They will decide their own management techniques and gain experience in leading projects through interaction with the simulation software.

As the project leader in the simulation, must create a plan to develop the simulation system assign the project team tasks to execute this plan. Beside that, the project leader must review the progress of the project to make sure the project team's progress is on schedule and reschedule their work if necessary.

The simulation system must completed by specified deadline, under budget and be of acceptable quality. In addition, the project leader must communicate well the user and manager in the simulation to lead the project successfully.

1.3 Project Objective

- User can gain the experience in managing project from the simulation game.
- As a training tool for company to train their staff in managing project.
- As a learning tool for students who are taking Project Management course, test their knowledge about this course and therefore increase their interest and skill in handling project in future.
- Expose students to the real working environment such as, how they will manage the time, cost, people and problems of the project.
- After simulation, the user can understand about project leader's roles and responsibilities in a software development project.

1.4 Project Scope

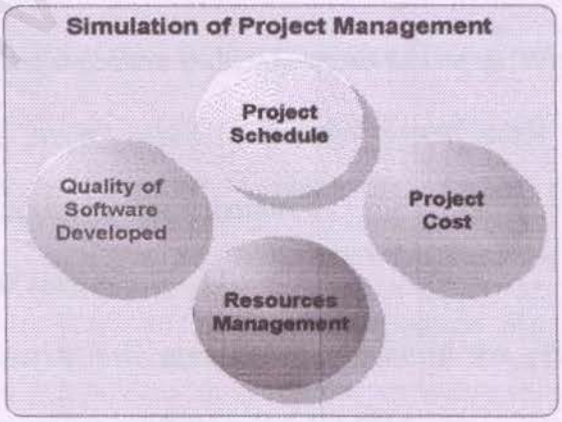

Simulation of Project Management
Project Schedule
Quality of Software Developed
Project Cost
Resources Management

Figure 1.3: Scope of Simulation of Project Management

This project has 4 major parts which shown on Figure 1.3, there are:

a. Project schedule

b. Project cost

c. Resource management

d. Quality of software developed

In the section below, is the explanation about the marks/scores allocation for the 4 major parts of project management.

a. Project Schedule

- User need to key in the project's name, start date and end date of the simulation.
- User can view the Gantt chart, pert chart and also the calendar.
- The simulation time will start to generate once the game be loaded and stopped when the player stop or pause the game.
- During the simulation game, the player may check the progress of the project at anytime once the game started.
- Incident issue:

  Top-level management suddenly wants to change requirements to shorten or extend the project schedule. So, the system needs to allow player to add budget to deal with this situation.
- Allocation marks:
  - Full marks will give for this part if the player can complete the simulation project before the deadline.

- For each one day's delay will deduct 5 marks for this part.
- If delay more than 10 days, the marks is 0.
- Table 1.2 shows the allocation mark for the project schedule performance.

Table 1.2: Allocation marks of project schedule

Performance	Marks
Complete before deadline	25 (full)
Delay each 1 day and less than 10 days	-5
Delay more than 10 days	0

b. Project Cost

- Simulation project's budget is under RM100,000.
- The total of effort cost for everyday of the simulation game's time will show at the progress check module.
- Incident issue:

  Display the warning message/complaint from top-level management or client when the project starts to over budget. This is to ensure the project to stay on the right track.
- Allocation marks:
  - Project complete under the budget is full marks (25%).
  - If over budget, over every RM1,000 of the budget will deduct 1 mark.
  - Table 1.3 shows the allocation mark for the project schedule performance.

Table 1.3: Allocation marks of project cost

Performance	Marks
Under budget	25 (full)
Over every RM1,000 of budget	-1

c. Resource Management

- Every staff has their own profile. Each staff has their own capability & knowledge in different field. The staff profile contains staff information; include attitude, skill, experience, salary & working efficiency.
- Player assigns the team member for the simulation project with refers the staff profile. The staff profile contains personal information of every staff.
- Progress of the project depends on the performance of the team member.
- The project leader (player) need to keep communicate with every member of the project team.
- Incident issue:

  Some of the workers who really have bad attitude want to quite in the mid way of the development project. So, it will test the player on how to handle in this circumstance by using the knowledge on project management.
- Allocation marks:
  - Table 1.4 shows the allocation mark for the resource management performance.

Table 1.4: Allocation marks of resource management

Performance	Marks
Every resource's problem occur	-5

d. Quality of Software Developed

- The quality of the system being developed for the simulation project is measured by the number of errors in the system.
- To develop a satisfactory system, it is necessary to reduce the number of errors to less than 0.5 errors per kilo step (1000 lines of code) by the end of development.
- Subsystem Quality Review
  - The project leader reviews the current documentation of the modules in a subsystem about the number of errors are found and corrected.
  - The "potential errors" can be created or corrected in module design (MD), coding (CD), and module test (MT) tasks.
  - The "potential errors" are only found and corrected in the module design review (MDR), coding review (CDR), and module test review (MTR) tasks and in the integration test (IT) process.
- Allocation marks:
  - The number of potential errors remaining per KLOC developed is equal or less than 0.5 errors will get full mark.
  - If the number of potential errors remaining per KLOC developed is more than 3 will get 0 marks.
  - Table 1.5 shows the allocation mark for the quality of software (* e = number of potential errors remaining per KLOC).

Table 1.5: Allocation marks of quality of software performance

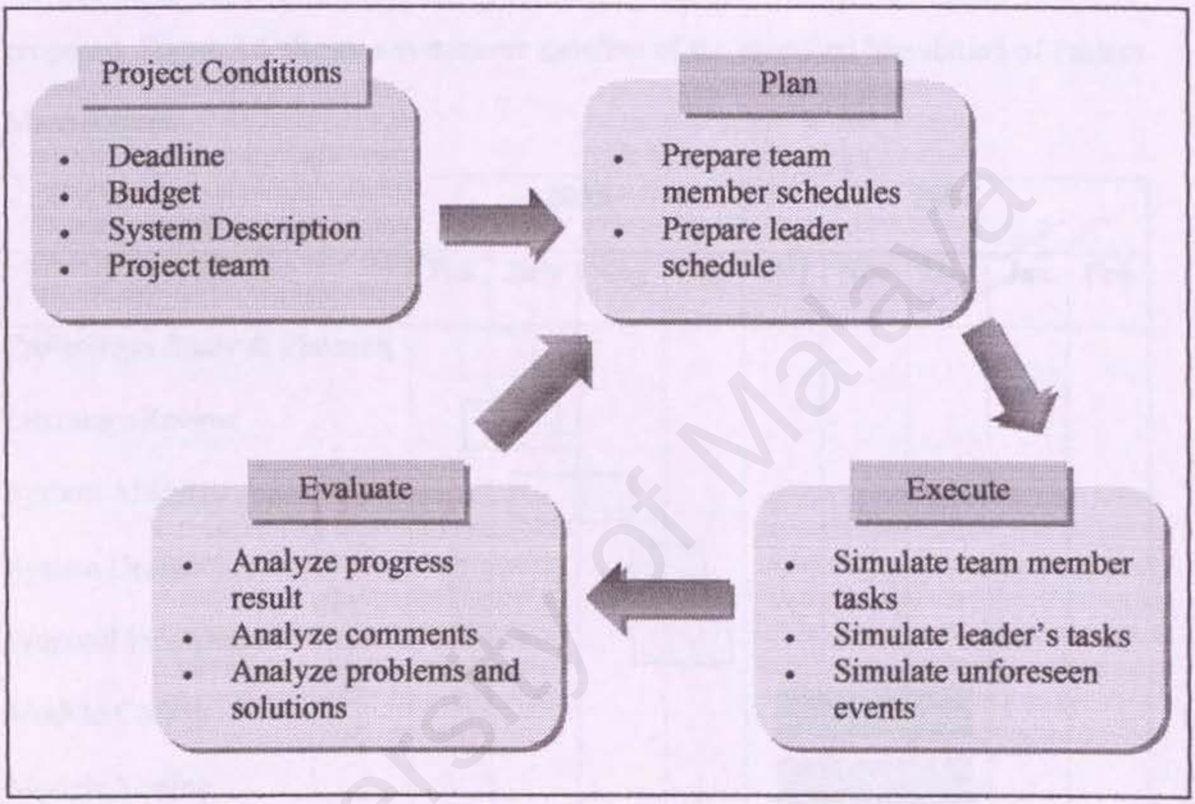
Performance	Marks
e <= 0.5	25
0.5 < e <= 1.0	20
1.0 < e <= 1.5	15
1.5 < e <= 2.0	10
2.0 < e <= 2.5	5
2.5 < e	0

1.5 Overview of Project Management Simulation

This project simulation has various aspects such as project schedule, project cost, resource management, quality of the software developed by the project team members and etc. In project schedule, deadline of the project simulation is fixes. If the completion of simulation project is delayed, score will be deducted. For the project cost part, the player will be given a budget on this project. The score of this part is depend on rang of between the project budget and the total effort costs that spend on the whole project simulation.

In resource management, the project leader assigns human resources for the tasks of the simulation project such as programmer, system analysis, designer and etc. For quality project, project leader can review the number of errors for each subsystem modules in subsystem quality review report. The quality of the software is depends on the number of errors found.

The overall process of the simulation is shown on Figure 1.3. After the new simulation game is started, the project leader needs to plan, execute and evaluate the simulation project. These three processes are iterative and will repeat until the project complete or when the project leader declare ending of the project.



Project Conditions
• Deadline
• Budget
• System Description
• Project team
Plan
• Prepare team member schedules
• Prepare leader schedule
Execute
• Simulate team member tasks
• Simulate leader's tasks
• Simulate unforeseen events
Evaluate
• Analyze progress result
• Analyze comments
• Analyze problems and solutions

Figure 1.4: Overview of the project management simulation

1.6 Project Schedule

A project schedule that consists of the whole development's essential as it acted as a time management and control to the developer. A carefully planned out project will achieve a systematic progress and ensure on-time delivery of the product. The draft of the proposed schedule is drawn in Gantt chart format for a good look of the timeline proposed. Figure 1.5 shows a systematic timeline of the proposed Simulation of Project Management.

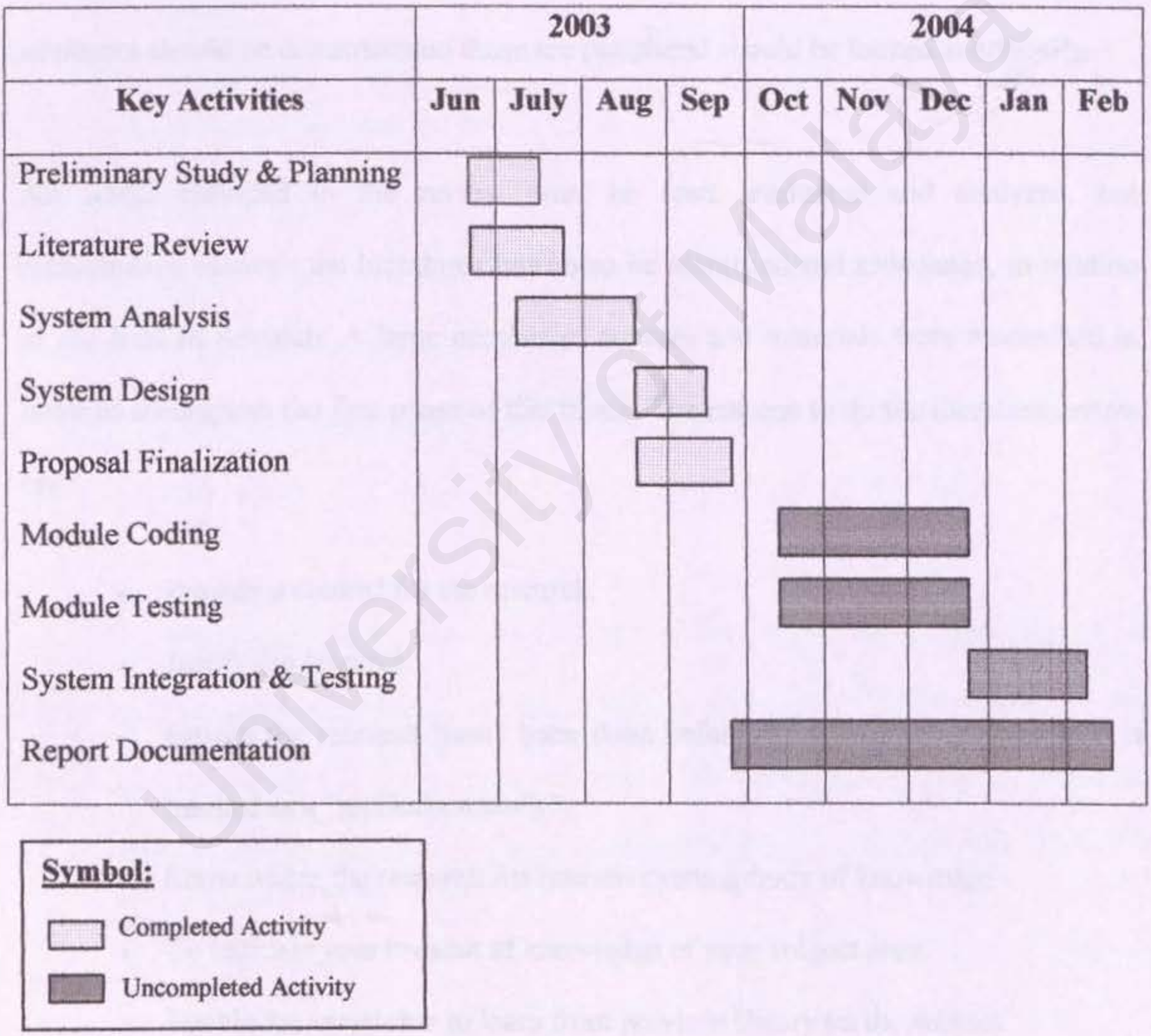


Figure 1.5: Project schedule for the Simulation of Project Management System

Chapter 2: Literature Review

2.1 Role of Literature Review

A literature review is an evaluative report of information found in the literature such as scholarly articles, books, thesis or dissertations, conference proceedings and other sources that are related to the selected area of study. The review should describe summaries, evaluate and clarify this literature. It should give a theoretical base for the research and help the author determine the nature of the research. Works that are irrelevant should be discarded and those are peripheral should be looked at critically.

All works included in the review must be read, evaluated and analyzed, but relationships between the literatures must also be identified and articulated, in relation to the field of research. A large number of sources and materials were researched in order to accomplish the first phase of this thesis. The reasons to do the literature review are:

- Provide a context for the research.
- Justify the research.
- Ensure the research hasn't been done before (or if it is repeated, that it is marked as a "replication study").
- Show where the research fits into the existing body of knowledge.
- To increase your breadth of knowledge of your subject area.
- Enable the researcher to learn from previous theory on the subject.
- Illustrate how the subject has been studied previously.

- Highlight flaws in previous research.
- Outline gaps in previous research.
- Show that the work is adding to the understanding and knowledge of the field.
- Help refine, refocus or even change the topic.
- To identify opposing views.
- To identify information, ideas and methods that may be relevant to your project.

2.2 Approach of Literature Review

2.2.1 Internet Surfing

The Internet is a powerful communications tool that allows global access to any form of information you desire. Today the Internet is used for research, commerce, business, marketing, education, and entertainment. It is known fact that the Internet constraint the fastest growing source of information and provide a great deal of information.

Various website were surfed to obtain resource on the subject matter, especially on the up-to-date information, which were not available on books. These website are very helpful especially need to learn something new within a short period of time. Numerous tools, applications and systems on project management were obtained from the World Wide Web.

Internet surfing not only provided sufficient information about the tools but also displayed a clear picture about the structure of the system and the techniques used. The

World Wide Web is filled with loads of information. So, it is needs to use search engines to find the information in a short time. Example of Search Engines as below:

- http://www.google.com
- http://www.about.com
- http://www.altavista.com
- http://www.excite.com
- http://www.go.com
- http://www.hotbot.com

To get the desired information or the relevant website, using the right key word is crucial. The wrong use of words may lead to unwanted and irrelevant information and cause unnecessary waste of time. From the Internet, I can get a lot of project management and thesis information. Beside that, I also got opportunities to use freeware web-base project management system in order to make me more clear and understand about the project management.

2.2.2 Reading

Various forms of materials were read and reviewed to obtain sufficient knowledge about the project. The sources included books, journals, previous thesis, documentation, magazines and newspaper. Academic Books focus on a particular topic and usually more detailed. They can provide a worth of information about a particular topic area and may be incorporated into graduate research, depending on the book, the topic, and other available literature.

The books like "How to be a better project leader" and "Project Leadership" that I got from the library gave me the knowledge about the project leader tasks, project leader characteristics, project leader roles and so on.

Documentation of project management application and system become the fundamental resource to understanding the structure, functions and the developing method of the preceding systems.

Previous thesis obtained from the library provided the guidelines to carrying out the project. Beside that, the magazines and newspapers on computer technology provide the latest news and updates on computer technology.

2.2.3 Supervisor Experience

Beside the Internet surfing and reading, supervisors advise also important to success the project. My supervisor, Assoc. Prof. Dr. Ow Siew Hock provided the relevant information and gave the guideline for me to carry out the project. Dr. Ow Siew Hock also showed the pervious project system to us in order to give us as references to build a good system.

2.2.4 Discussion

Useful advices have been given for each section meeting conduct with my group members. It is very helpful as a reminder when carrying out the system development process and also useful for error correctness.

2.3 Existing System Review

There are a variety of project management tools in use today. Some require a computer with supporting software, while others can be used manually. To obtain a good understanding on project management and project management tools, many systems, from stand-alone applications to web-based tools and freeware tools were review. Among the many applications and tools that were reviewed, three applications are explained in detailed in this section. The review of systems elaborates on the features, advantages, disadvantages, quality, support and compatibility. Below are the systems and tools reviewed:

- Fujitsu Project Management Simulation
- Ace Project
- Microsoft Project 2000

2.3.1 Fujitsu Project Management Simulation

This software is example of Simulation of Project Management which developed by three person Japanese. The participant (player) will assume the role of project leader in managing a simulation software development project. The player will decide his own management techniques and gain experience in leading project through interaction with the simulation software.

The Project Management Game Simulation software contains 3 different project models. The player will select one project models before start the simulation.

Table 2.1: Three different project models

	Project 1	Project 2	Project 3
Deadline	5/30	5/30	5/30
Budget	37 man-months ($274,550)	48 man-months ($353,770)	24 man-months ($180,710)
Size	20 ks	28 ks	12 ks
Team Members	8 members	10 members	5 members

In this simulation, the player will be responsible for managing the development of the system through its module design, coding, module test, and integration test phases. The Project Management Game contains 5 input screens as initial plan for the players to input their plan to develop the simulation project. [Project Management Simulation Participant's Handbook, 1988]

When the simulation stops, the player may modify the initial plan through the input screens. After modify the plan, the player can restart the simulation to execute the plan. The simulation will output messages concerning the actions of the simulation project leader, project team, manager and user. [Project Management Simulation Participant's Handbook, 1988]

At the end of the simulation, the simulation result screen evaluates the project leader's management of the project in term of:

- Meet the project deadline
- Staying under budget

- Building a quality system
- communicate with the user, manager and project team
- Manage and utilize the project team effectively

Table 2.2: Strangeness and weakness of Fujitsu Project Management Simulation

Strangeness	Weakness
▪ The simulation project management as a real world project management. ▪ Input screens are systematic and consistent. ▪ Information and data of the output screens are clear and simple.	▪ User interface not attractive and the graphic user interface design no compatible with WIMP (Windows, Icon, Menu and Pointing device). ▪ Don't have pointing device function to select item.

Overview of simulation screens

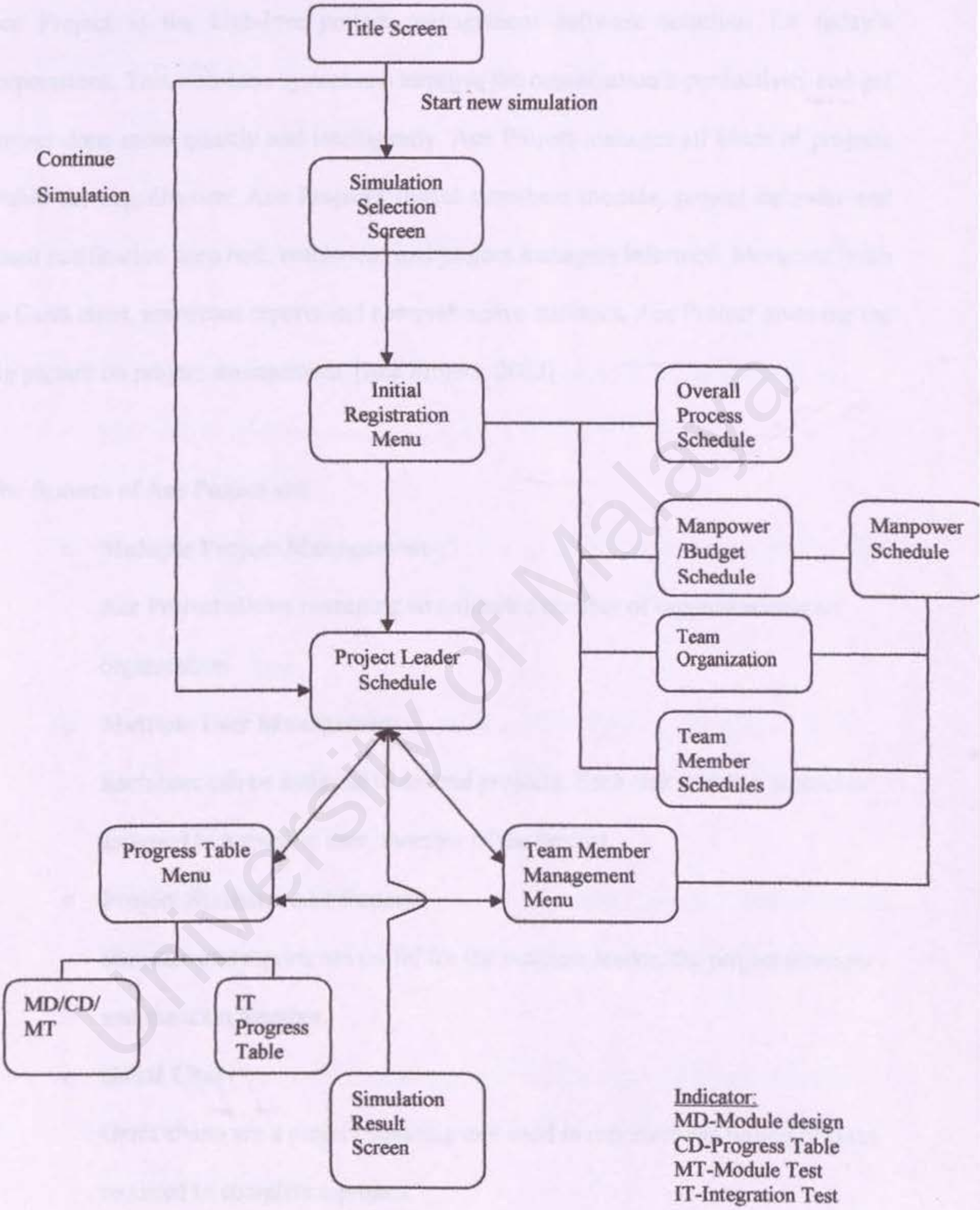


Figure 2.1: The Flow Chart of Fujitsu Project

2.3.2 Ace Project

Ace Project is the high-level project management software solutions for today's corporations. This web-base system can improve the organization's productivity and get project done more quickly and intelligently. Ace Project manages all kinds of projects within the organization. Ace Project's useful timesheet module, project calendar and email notification keep both employees and project managers informed. Moreover, with its Gantt chart, numerous reports and comprehensive statistics, Ace Project gives me the big picture on project management. [Ace Project, 2003]

The features of Ace Project are:

- **Multiple Project Management**

  Ace Project allows managing an unlimited number of projects within an organization.

- **Multiple User Management**

  Each user can be assigned to several projects. Each task within a project is assigned to a specific user, member of the project.

- **Project Statistics and Reports**

  Statistics and reports are useful for the business leader, the project manager and the team member.

- **Gantt Chart**

  Gantt charts are a project planning tool used to represent the timing of tasks required to complete a project.

- **Calendar**

Calendar shows task deadlines in a monthly format.

- **Employee Time Sheet Tracking and Approval**

  Every user fills out his/her time sheet on a weekly basis.

- **Security and Access Rights**

  Ace Project offers two security levels: Administrator and Normal. An administrator has all the rights. A normal user's access rights may vary.

- **Email Notification**

  Each user may decide to receive an email when he or she is assigned to a new task or when one of his or her tasks is modified.

- **File Attachments**

  This document management feature lets you attach files such as screenshots, images and text documents to tasks.

- **Users Preferences**

  Each user can specify his or her preferences, adapting Ace Project to his or her uses of the system.

- **Discussion Forum**

  This helps teams improve communications while leaving a record of those conversations.

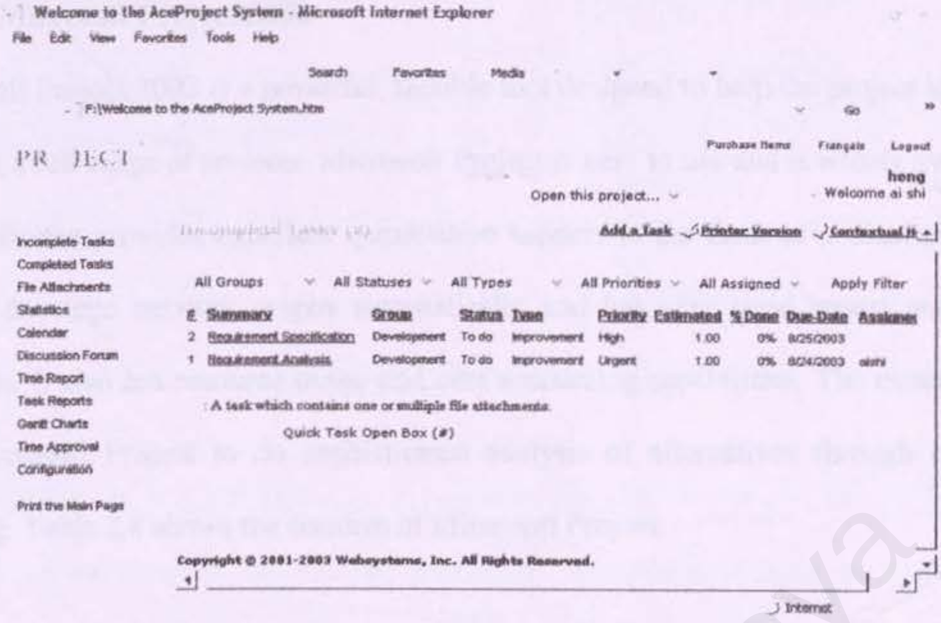
- **Message Center**

  This feature can send messages to a specific user or a whole project team.

[Ace Project, 2003]

Table 2.3: Strangeness and weakness of Ace Project

Strangeness	Weakness
▪ A Universal System, Compatible with all Browsers. ▪ Free. ▪ Can access at anytime and anywhere. ▪ No software to install. ▪ All latest versions are available.	▪ Not available when server down. ▪ Own company's strategies of project management may know by the system developer.

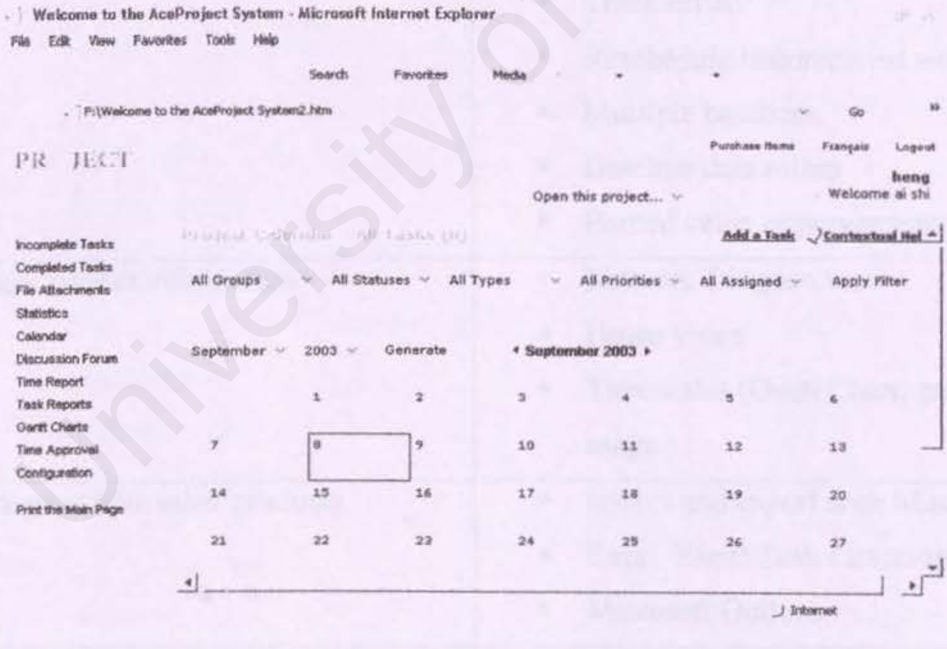


#	Summary	Group	Status	Type	Priority	Estimated	% Done	Due-Date	Assignee
2	Requirement Specification	Development	To do	Improvement	High	1.00	0%	8/25/2003	
1	Requirement Analysis	Development	To do	Improvement	Urgent	1.00	0%	8/24/2003	aishi

: A task which contains one or multiple file attachments.

Copyright © 2001-2003 Websystems, Inc. All Rights Reserved.

Figure 2.2: Ace Project - incomplete tasks schedule


Welcome to the AceProject System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Search Favorites Media
F:\Welcome to the AceProject System2.htm
Go
PR JECT
Purchase Items Français Logout
heng
Welcome ai shi
Open this project...
Project Calendar - All Tasks (0)
Add a Task Contextual Hel
Incomplete Tasks
Completed Tasks
File Attachments
Statistics
Calendar
Discussion Forum
Time Report
Task Reports
Gantt Charts
Time Approval
Configuration
Print the Main Page
All Groups All Statuses All Types All Priorities All Assigned Apply Filter
September 2003 Generate
◂ September 2003 ▸
Internet

	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27

Figure 2.3: Ace Project – project calendar

2.3.3 Microsoft Project 2002

Microsoft Project 2002 is a powerful, flexible tool designed to help the project leader to manage a full range of projects. Microsoft Project is easy to use and is widely available. This software provides excellent quantitative support in the form of critical path and PERT, develops network graphs automatically and has very good report generating facilities. It also has resource usage and cost accounting capabilities. The manager can use Microsoft Project to do sophisticated analysis of alternatives through resource leveling. Table 2.4 shows the features of Microsoft Project.

Table 2.4: Features of Microsoft Project 2002

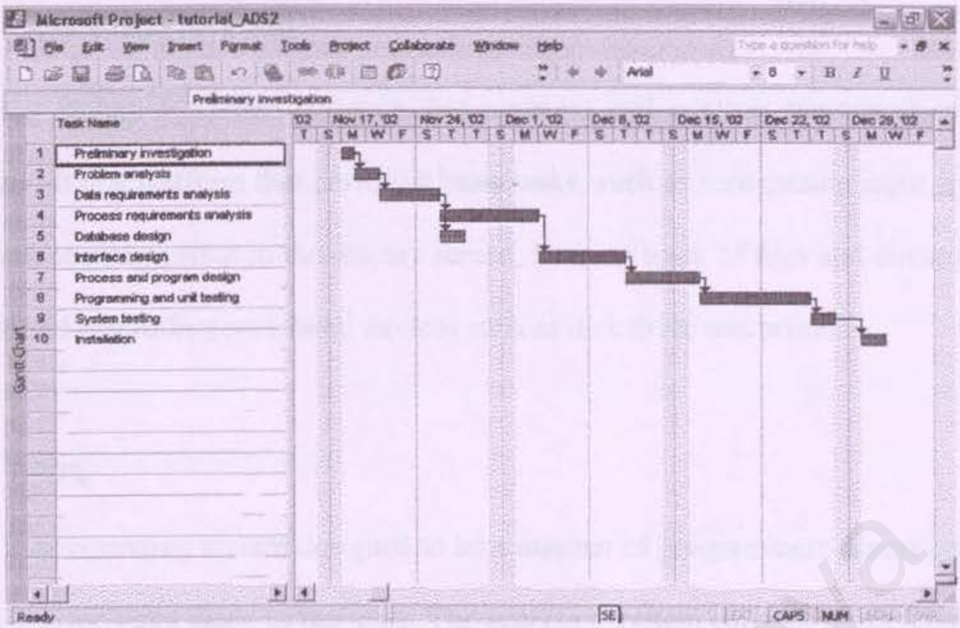
Task	Features
Schedule and track tasks	▪ Assign resources ▪ Track actual ▪ Reschedule uncompleted work ▪ Multiple baselines ▪ Baseline data rollup ▪ Earned value improvements
View project information	▪ Network Diagram view ▪ Usage views ▪ Timescales (Gantt Chart, graph and usage)
Integrate with other products	▪ Import and export with Microsoft ▪ Excel Excel Task List template ▪ Microsoft Outlook
Manage a collaborative project	▪ Simplified Timesheet ▪ Manager Transactions page ▪ Multiple managers

	▪ Task lists ▪ Resource comments
Manage documents and issues	▪ Document management ▪ Issue tracker
Resource management	▪ Resource pooling ▪ Resource leveling

Microsoft Project is a valuable project management tool. It provides one of the most flexible and user-friendly systems for generating critical path analysis. It has a variety of printing functions that user the ability to communicate this same information to other.

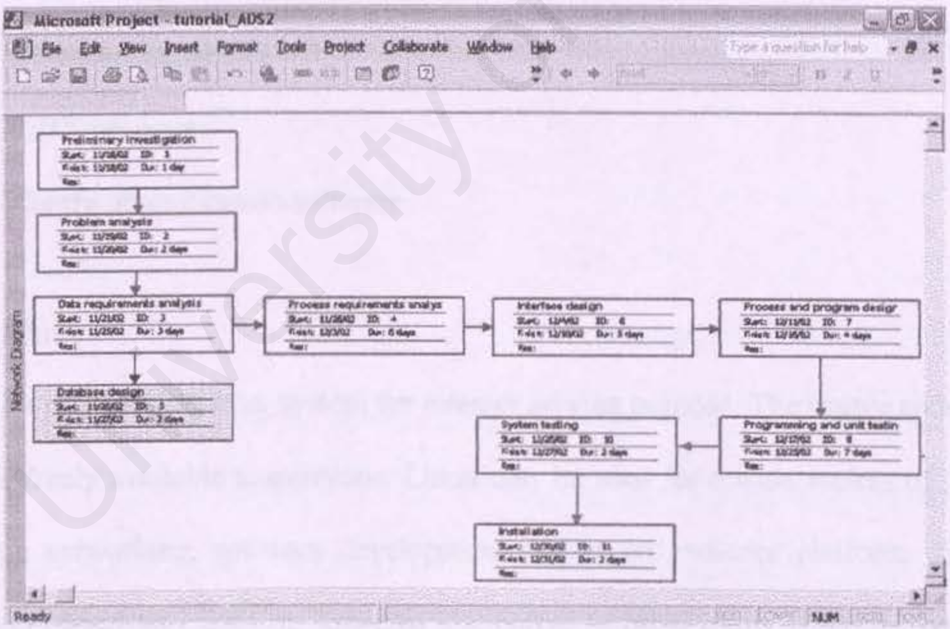
Table 2.5: Strangeness and weakness of Microsoft Project 2002

Strangeness	Weakness
▪ Just as in Microsoft Office 2002, only the items that users use most often are prominently featured on the menu. ▪ With the user-friendly interface, user can use it easily and comfortable ▪ Flexible analysis ▪ Easer reporting	▪ Need to install ▪ Need to purchase the software and license


Microsoft Project - tutorial_ADS2
File Edit View Insert Format Tools Project Collaborate Window Help
Type a question for help
Arial
8
Preliminary investigation
'02 Nov 17, '02 Nov 24, '02 Dec 1, '02 Dec 8, '02 Dec 15, '02 Dec 22, '02 Dec 29, '02
T S M W F S T T S M W F S T T S M W F S T T S M W F
Gantt Chart
Ready
SE
EXT CAPS NUM

	Task Name
1	Preliminary investigation
2	Problem analysis
3	Data requirements analysis
4	Process requirements analysis
5	Database design
6	Interface design
7	Process and program design
8	Programming and unit testing
9	System testing
10	Installation

Figure 2.4: Microsoft Project 2002 - Gantt chart


Microsoft Project - tutorial_ADS2
File Edit View Insert Format Tools Project Collaborate Window Help
Type a question for help
Network Diagram
Ready
NUM

Preliminary investigation
Start: 11/18/02 ID: 1
Finish: 11/18/02 Dur: 1 day
Res:

Problem analysis
Start: 11/19/02 ID: 2
Finish: 11/20/02 Dur: 2 days
Res:

Data requirements analysis
Start: 11/21/02 ID: 3
Finish: 11/25/02 Dur: 3 days
Res:

Process requirements analys
Start: 11/26/02 ID: 4
Finish: 12/3/02 Dur: 6 days
Res:

Interface design
Start: 12/4/02 ID: 6
Finish: 12/10/02 Dur: 5 days
Res:

Process and program desigr
Start: 12/11/02 ID: 7
Finish: 12/16/02 Dur: 4 days
Res:

Database design
Start: 11/26/02 ID: 5
Finish: 11/27/02 Dur: 2 days
Res:

Programming and unit testin
Start: 12/17/02 ID: 8
Finish: 12/25/02 Dur: 7 days
Res:

System testing
Start: 12/26/02 ID: 10
Finish: 12/27/02 Dur: 2 days
Res:

Installation
Start: 12/30/02 ID: 11
Finish: 12/31/02 Dur: 2 days
Res:

Figure 2.5: Microsoft Project 2002 - Network diagram (PERT Chart)

2.4 Operating System

Operating system is system software that manages and controls the activities of the computer. It is a platform that performs basic tasks, such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk and controlling peripheral devices such as disk drive and printer.

2.4.1 UNIX

UNIX is an operating system designed to let a number of programmers access computer at the same time and share its resource. The operating system controls all the commands from all of the keyboards and all of the data being generated, and permits each user to believe he or she is the only person working on the computer. The uniqueness of UNIX:

- Multitasking capability
- Multi-user capability
- UNIX programs
- Library of application software

2.4.2 Linux

Linux is a popular operating system for internet serving purpose. The source code of the Linux is freely available to everyone. Linux may be used for a wide variety of purpose including networking, software development and as an end-user platform. Linux is often considered an excellent, low-cost alternative to other more expensive operating system.

2.4.3 Windows 2000 Professional

Windows 2000 Professional was built on Windows NT technology with the user-friendly user interface. It can be used for running software applications, connect to internet and intranet sites, and access files, printer and network resources. Furthermore, it is the windows operating system for both businesses laptop and desktop system. Microsoft Windows 2000 Professional is known for its flexibility, manageability, reliability and its web capabilities.

2.4.4 Windows XP Professional

Windows XP Professional delivers new standard in reliability and performance of the windows operating system. It provides efficient and dependable computing experience. Windows XP Professional is built on Windows 2000 and Window NT workstation core software code, known as the NT kernel. It is more powerful, secure and stable than the previous version of windows operating system.

2.5 Database

The term database describes a collection of data organized in a manner that allows access, retrieval and use of that data. A database management system, such as Access, allows you to use a computer to create a database; add, change and delete data in database; sort the data in the database; retrieve data in the database; and create form and report using the data in the database.

2.5.1 My SQL

My SQL is an open source relational database management system. It is a small, compact, easy to use database server, ideal for small and medium sized applications. It is available on a variety of UNIX platforms, Linux, Windows NT/95/98/2000.

2.5.2 Microsoft SQL Server 2000

Microsoft SQL Server is a single process, multithreaded relation database server primarily intent for transactional processing. It is based on the client/server architecture, which divides processing into two components:

- Front-end or client component, that run on a local workstation
- Back-end or server component, which runs on a remote computer

2.5.3 Microsoft Access 2000

Microsoft Access 2000 is a windows-based database management system. It is a member of Microsoft Office 2000 family and it runs under Windows 95/98/NT/2000 operating system. Access is easily the world's most popular Relational Database Management System (RDBMS). It is powerful and get easy to use.

The database administrator can use Access to design and use database very quickly because of the very user-friendly user interface. Furthermore, tables, form, queries and reports can be generated just at a snap of a finger by using the set of wizards that come with this software. All this make Access an excellent all-in-one database tool for creating standalone database application.

2.6 Programming Language

2.6.1 C++

C++ as an extension of C was developed by Bjarne Stroustrup in the early 1980s at Bell Laboratories. C++ improves on many of C's features and provides object-oriented-programming (OOP) capabilities that hold great promise for increasing software productivity, quality and reusability. C++ is a superset of C, so programmers can use a C++ compiler to compile existing C programs, and then gradually evolve those programs to C++. C++ is a rich language. C++ programs are constructed with two major building blocks, functions and user-defined data types called classes. [Deitel & Deitel, 2001]

2.6.2 Java

Java generated interest in the business community because of the phenomenal interest in the World Wide Web. Java is now used to create Web pages with dynamic and interactive content, to develop large-scale enterprise applications, to enhance the functionality of World Wide Web servers, to provide applications for consumer devices and for many other purposes. [Deitel & Deitel, 2002]

Java programs consist of pieces called classes. Classes consist of pieces called methods that perform tasks and return information when they complete their tasks. Most Java programmers take advantage of rich collections of existing classes in Java class libraries. The class libraries are also known as the Java APIs (Application Programming Interfaces). One of the advantages of Java is many class libraries are supplied by independent software vendors (ISVs) and are available from the internet and World Wide Web as freeware or shareware. [Deitel & Deitel, 2002]

2.6.3 Visual Basic. Net

Microsoft Visual Basic.NET is the latest version of the Visual Basic tool set that enables the developers to address today's pressing application development issues effectively and efficiently. Visual Basic.NET enable developer to create Windows-based applications in less time, incorporate data access from a wider range of database scenarios, and create components with minimal code. [Product Overview for Visual Basic .Net 2003, 2003]

Developers can use Visual Basic.NET to build Windows-based applications that leverage the rich user interface features available in the Windows operating system. All the Rapid Application Development (RAD) tools that developers have come to expect can be found in Visual Basic.NET, including drag and drop design and code behind forms. The forms feature provides an easy way to create a graphical user interface. [Product Overview for Visual Basic .Net 2003, 2003]

Visual Basic.NET supports full object-oriented constructs to enable more componentized, reusable code. Language features include full implementation inheritance, encapsulation, and polymorphism. These features provide the availability for developers to reuse the code where enable more rapid development.

2.7 Multimedia Tools

The multimedia tools can help to create high impact user interfaces and environments. It is use to create the multimedia content likes animation, audio effects, and etc.

2.7.1 Macromedia Flash MX

Multimedia Flash was developed by Macromedia, Inc. It is a multimedia tool that enables developer to create multimedia content for internet content and application. The approachable environment includes powerful video, animation, audio, vector graphics, bitmap graphics and application development features, which allow designers and developers to create rich user interfaces, online advertising, product tours, e-learning courses and enterprise application front ends. [Macromedia, 2003]

2.7.2 Macromedia Director 8.5

Macromedia Director is an industry standard authoring tool for multimedia production. It is mainly designed for web application. It combines multimedia elements into portable movie and backs them up with Lingo, which is Director's own interactive scripting language. Lingo is a powerful scripting language. It enables a Director developer and the user to control any situation in the production. While adding features to Director that Lingo does not provide, we can obtain or create C modules called Xtras, which could communicate with Director. [Macromedia, 2003]

Furthermore, Director has a host of media editors to create, modify, import or edit graphics, sounds, text, video and interactivity to deliver the highest quality productions possible.

2.8 Relationship of Literature Review to Project

After literature review process, I get the information about the concept and environment of the project management. From the reading process, I can get the ideals and understand about project manager's roles, tasks, responsibilities, characteristics and the problems that will meet on the software development project.

The books like "How to be a better leader" and "Project Leadership" that I got from the library gave me the knowledge about the project leader tasks, project leader characteristics, project leader roles and so on.

The existing system reviews play a very important role as a guideline for how to define the story of the game and program structure. Especially, "Fujitsu Project Management Simulation" is the main reference and guideline for this project.

The main objective of the proposed system is to provide a user-friendly stand-alone application software system; the system needs to be simple and easy to use. The literature reviews has helped to decide on the most suitable time tracking method and also determine the important elements of a project management system.

2.9 Summary of Literature Review

The literature review need spend a lot of time to get the sources, read the information and do the analysis about the relevant topic. From the reading process, I can get the knowledge about the project management and project leader for a software development project. The literature on the existing system has revealed the pros and cons of the available project management systems. There are countless applications available, ranging from stand-alone applications to web-based systems and freeware tools. Most of the freeware tools are the trial version of an application. The users are encouraged to try the trial version and later purchase the actual system. To obtain the full amenities the user required purchasing the application with at sum of money. Therefore is not easy to get a freeware tool with all the necessary features. At the same time, there is not enough printed material about this system and contents of the existing printed materials are not updated.

Chapter 3: Methodology

3.1 Development Methodology

A methodology is a collection of procedures, techniques, tools and documentation aids, supported by a philosophy, which will help the system developers in their efforts to implement a new information system. A methodology will consist of phases and sub-phases, which will guide the system developers in their choice of techniques that might be appropriate at each stage of the project and also help manage, control and evaluate system project.

There are several process models in system development, such as:

- Waterfall Model
- Waterfall Model with prototyping
- V Model
- Spiral Model
- Rational Unified Process

3.2 Rational Unified Process Methodology

The Rational Unified Process or RUP is a software engineering process. It provides a disciplined approach to assigning tasks and responsibilities within a development organization. Its goal is to ensure the production of high quality software that meets the needs of its end users within a predictable schedule and budget. [Rational Unified Process: Best Practices for Software Development Teams, 2003]

The Rational Unified Process methodology is comprised of the ideas and experiences of industry leaders, partners, and literally thousands of real software projects, carefully synthesized into a practical set of best practices, workflows, and artifacts for iterative software development.

The process enhances team productivity and delivers software best practices via guidelines, templates and tool mentors for all critical software lifecycle activities. The best practices promoted by the RUP Methodology are including:

- Develop software iteratively
- Manage requirements
- Use component-based architectures
- Visually model software
- Verify software quality
- Control changes to software

RUP is a process product, designed like any software product, and integrated with the Rational suites of software development tools. RUP is based on an object-oriented design structure and uses Unified Modeling Language™ (UML) as the modelling language. [Rational Unified Process: Best Practices for Software Development Teams, 2003]

3.2.1 Process Overview

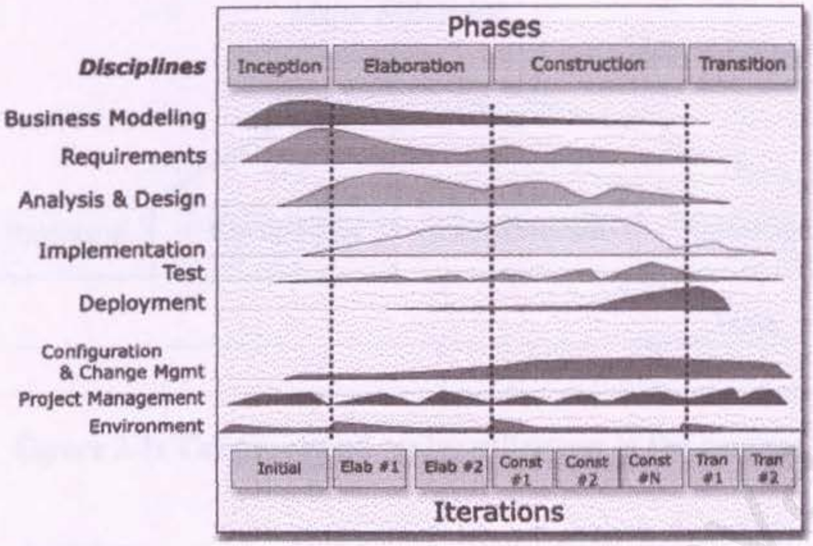


Figure 3.1: Two dimensions of RUP process

The RUP project structure is shown in two dimensions:

- The horizontal axis represents time and shows the lifecycle aspects of the process as it unfolds
- The vertical axis represents disciplines, which group activities logically by nature

The first dimension represents the dynamic aspect of the process as it is enacted and it is expressed in terms of phases, iterations, and milestones. The second dimension represents the static aspect of the process: how it is described in terms of process components, disciplines, activities, workflows, artifacts, and roles.

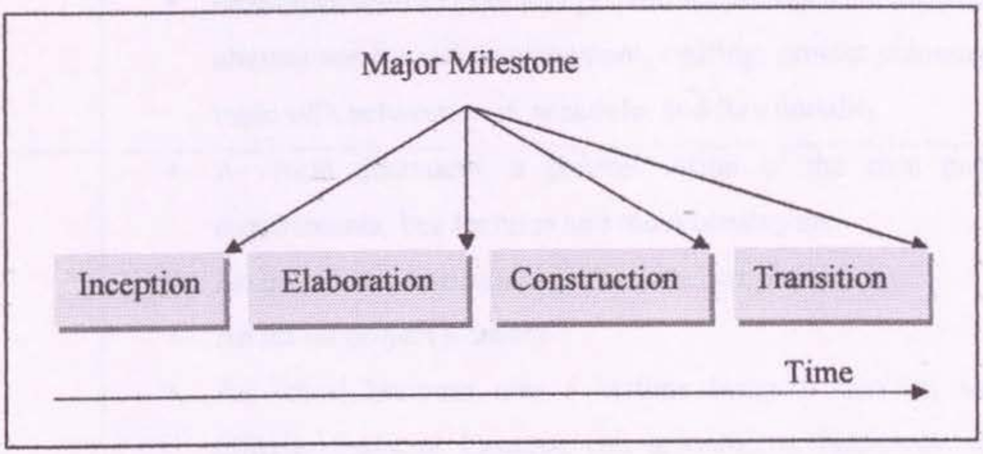


Figure 3.2: The phase and major milestone in the process

3.2.2 Phases of RUP

Rational Unified Process methodology divides a software development cycle into four phases: Inception, Elaboration, Construction, and Transition phases. Phases are separated by milestones. Each phase is sub-divided into one or more iterations. With the iterative approach, the emphasis of each workflow will vary throughout the lifecycle. [Rational Unified Process for outsourced software development, 2003]

3.2.2.1 Inception Phase

Table 3.1: Inception phase overview

Inception Phase	
Goal	Establishes the product feasibility and delimits the project scope
Activities	▪ all the stakeholders of the proposed system must understand what is needed and investigate the intended functionality of the system ▪ developer must identify all use-cases with identify all external entities (actors) which interact with the system and define the nature of this interaction at a high-level

	▪ developer need to plan and prepare a business case and evaluate alternatives for risk management, staffing, project planning, and trade-offs between cost, schedule, and functionality
Result	▪ A vision document: a general vision of the core project's requirements, key features and main constraints ▪ An initial use-case model (10%-20% complete) ▪ An initial project glossary ▪ An initial business case (include business context, success criteria, financial forecast, risk assessment, resources needed, and a project plan) ▪ An initial risk assessment ▪ Preliminary model ▪ User-interface prototypes
Major Milestone	Lifecycle Objective Milestone
Decision	whether to continue the project, cancel the project, or re-consider the project's scope

3.2.2.2 Elaboration Phase

Table 3.2: Elaboration phase overview

Elaboration Phase	
Goal	To obtain a complete and detailed vision of the system
Activities	▪ analyze the problem domain ▪ establish a sound architectural foundation ▪ develop the project plan ▪ eliminate the project's highest risk elements ▪ elaborate vision document ▪ understating of the most critical use-cases is established to drive the architecture and project planning decisions

	▪ specify use cases in detail (all use cases and actors have been identified, and most use-case descriptions have been developed) ▪ design the system architecture ▪ architectural decisions have to be made with an understanding of the whole system: its scope, major functionality and nonfunctional requirements such as performance requirements ▪ predictably determine the cost and schedule for the completion of the project
Result	▪ Use-case model (at least 80% complete). ▪ Supplementary requirements document (non-functional requirement and any application requirements that are not captured within the use-cases). ▪ Software architecture description. ▪ Executable architectural prototype. ▪ Revised risk list and a revised business case. ▪ Development plan for the overall project which include the coarse-grained project plan, showing iterations and evaluation criteria for all iteration. ▪ Updated development case specifying the process to be used. ▪ Preliminary version of user manual.
Major Milestone	Lifecycle Architecture Milestone
Decision	Whether to continue with the project, cancel the project, or go back and re-think the scope and architecture of the system (The project may be aborted or considerably re-thought if it fails to pass this milestone).

3.2.2.3 Construction Phase

Table 3.3: Construction phase overview

Construction Phase	
Goal	Build the system or product.
Activities	▪ all remaining components and features will be developed and integrated into the system ▪ all features will be tested thoroughly in order to ensure that all of the requirements and features agreed to earlier are correctly implemented in the final solution ▪ refine the architectural design into code ▪ refine the architectural baseline to a completed system ▪ manage the resources and control the operations at minimum costs, under schedules, and produce a high quality software that meet all the user needs
Result	▪ The executable software including configuration and installation. ▪ Testing model including testing result. ▪ Completed and updated architecture description. ▪ The user manuals. ▪ A description of the current release. ▪ Legal documents such as contracts, license documents, etc.
Major Milestone	Initial Operational Capability
Decision	Whether the software, the sites, and the users are ready to become operational without exposing the project to high risks. Transition may be postponed by one release if the project fails to meet this milestone.

3.2.2.4 Transition Phase

Table 3.4: Transition phase overview

Transition Phase	
Goal	Delivery of a system ready for exhaustive testing to ensure that the requirements have been met to the satisfaction of the stakeholders.
Activities	▪ software developers need to address the issues that usually arise that require them to develop new releases ▪ correct errors and problems ▪ finish the features that were postponed ▪ do beta testing to validate the system, parallel operation with legacy systems, and conversion of operational databases to work with the new system ▪ train user how to use the new system ▪ develop user-oriented documentation ▪ react to user feedback ▪ postmortem devoted to learning and recording lessons for future cycles
Result	▪ Completed and corrected product release including all models of the system. ▪ Manuals for all the users.
Major Milestone	Product Release Milestone
Decision	Whether or not to start another development cycle (if the objectives have been met).

3.2.3 Disciplines of RUP

RUP is comprised of nine core disciplines: six *core process* disciplines (business modelling, requirements, analysis & design, implementation, test, and deployment) and three *core supporting* disciplines (project management, configuration & change management, and environment). A discipline is a set of activities that is performed by one of the roles in a project and describes a meaningful sequence of activities that produce a useful result (usually an artifact). [Rational Unified Process: Best Practices for Software Development Teams, 2003]

Table 3.5: Nine core disciplines of RUP

Disciplines	Artifacts, roles and activities
Business Modelling	▪ understanding the structure and dynamics of the organization in which the system is to be deployed ▪ understanding the current problems in the target organization and identifying improvement potential ▪ ensuring customers, end users and developers have a common understanding of the target organization ▪ deriving system requirements to support the target organization
Requirements	▪ provide primary inputs (use case model and glossary) for analysis and design discipline ▪ produce use case model which validates at test discipline ▪ configuration & change management discipline provides change control mechanisms for the requirements discipline ▪ project management discipline plans the project and each iteration
Analysis & design	▪ transforming the requirements into a design of the system ▪ evolving a robust architecture for the system ▪ adapting the design to match the implementation environment, designing it for performance
Implementation	▪ defining the organization of the code, in terms of implementation

	subsystems organized in layers - implementing classes and objects in terms of components - testing developed components as units - integrating the results into an executable system
Test	- verifying the interaction between objects - verifying the proper integration of all the components of the software - verifying that all the requirements have been correctly implemented - identifying defects
Deployment	- providing a custom installation - providing a shrink wrap product offering - providing the software over internet
Configuration and change management	- supporting the development methods - maintaining the integrity of the product - ensuring the completeness and correctness of the configured product - providing a stable environment within which to develop the product - restricting changes to artifacts based on project policies - providing an audit trail on why, when and by whom any artifact was changed
Project management	- a framework for managing software-intensive projects - practical guidelines for planning, staffing, executing and monitoring projects - a framework for managing project risk
Environment	- designing, implementing and managing the project's required technical environments - defining the technical architectures for the development, system validation, testing and staging/release management environments

3.2.4 Advantages of Rational Unified Process Methodology

RUP has many advantages for use in modern software development:

a. The fact that it offers an iterative approach:

   i. risks are mitigated earlier

   ii. change is more manageable

   iii. there is a higher level of reuse

   iv. project teams learn as they go

   v. the finished product has a better overall quality

b. The fact that it is an architecture-centric process, this provides a solid basis for project management and leads to:

   i. a gain in intellectual control

   ii. frequent opportunities for reuse

c. The fact that it is use case driven:

   i. it is expressed from the perspective of users

   ii. it uses natural language and is thus easy to understand

   iii. it provides a high degree of traceability

   iv. it provides a simple way to decompose the requirements into chunks

   v. it offers a definition for the test cases

   vi. it aids in the planning of iterations

3.3 Reason of Choosing RUP Methodology

Rational Unified Process is selected as methodology to develop this Simulation of Project Management software. The reasons to choose this method are:

- RUP methodology is suitable for small project like Simulation of Project Management.
- The artifacts produced after each iteration is built upon by subsequent iteration thus allowing for more stable models and understanding of the system.
- Planning of each workflow before its commencement allows for more flexible scheduling of the life-cycle.

3.4 Summary

RUP will be used as the process methodology with modeling in the Unified Modeling Language (UML). The Rational Unified Process is easily customizable for the needs of this paper and provides a thorough approach to the software life cycle by allowing re-visitation of steps and continual improvement. The UML is also useful in projecting the system in numerous views and allowing for a better focus on the system as a whole.

Chapter 4: System Analysis

4.1 Requirement Analysis

The purpose of requirement analysis is to determine specifically what the system intended to do. Requirements analysis involves a thorough analysis of user information needs prior to systems design. Requirements analysis is the initial phase of systems development. Requirement analysis identifies the data information needed to automate some organizational task and to support achievement of organizational objectives. Many information system failures have been attributed to a lack of clear and specific information systems requirements.

4.1.1 Functional Requirement

Functional requirement is a description of activities and services a system must provide. Functional requirements are frequently identified in terms of inputs, outputs, processes, and stored data that are needed to satisfy the system improvement objectives.

4.1.1.1 Input Screens

The Simulation of Project Management contains 5 input screens for player to input the plan to develop the simulation project. These 5 input screens are:

- Overall Project Schedule
- Manpower/Budget Schedule
- Team Organization
- Team Member Schedule
- Project Leader Schedule

a) Team Organization

This is a module to build a project team for this simulation project. From this module, the player can:

- View staff attributes list
- Select staff for each position such as project analyst, programmer, designer, and testing engineer from the staff attributes list. (1 person for each position)

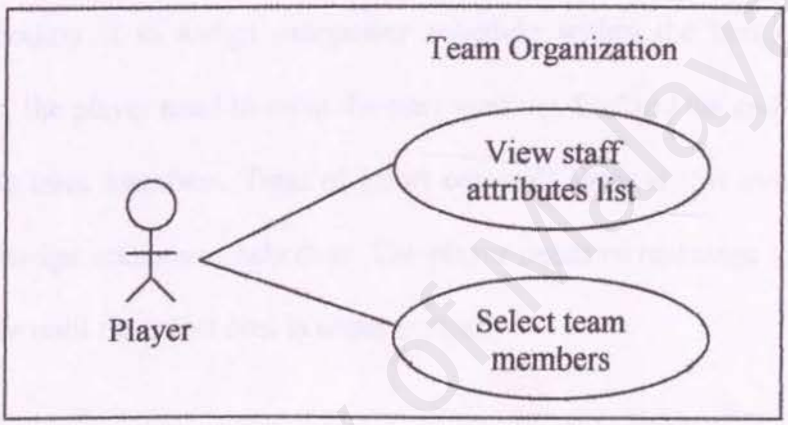


Figure 4.1: Team Organization use-case diagram

b) Overall Project Schedule

This module is to assign the duration for each process and the simulation project must complete within 60 days. From this module, the player must input the start and end dates for each process. This simulation project has 4 major processes:

1. Module Design (MD)
2. Coding (CD)
3. Module Testing (MT)
4. Integration Testing (IT)

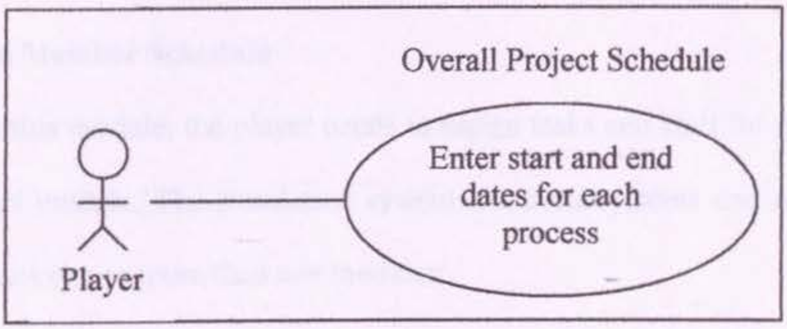


Figure 4.2: Overall project schedule use case diagram

c) Manpower/ Budget Schedule

This module is to assign manpower schedule within the budget. From this module, the player need to input the start working date and the end working date for each team members. Total of effort cost will show at this module after the player assign manpower schedule. The player needs to rearrange the manpower schedule until the effort cost is under budget.

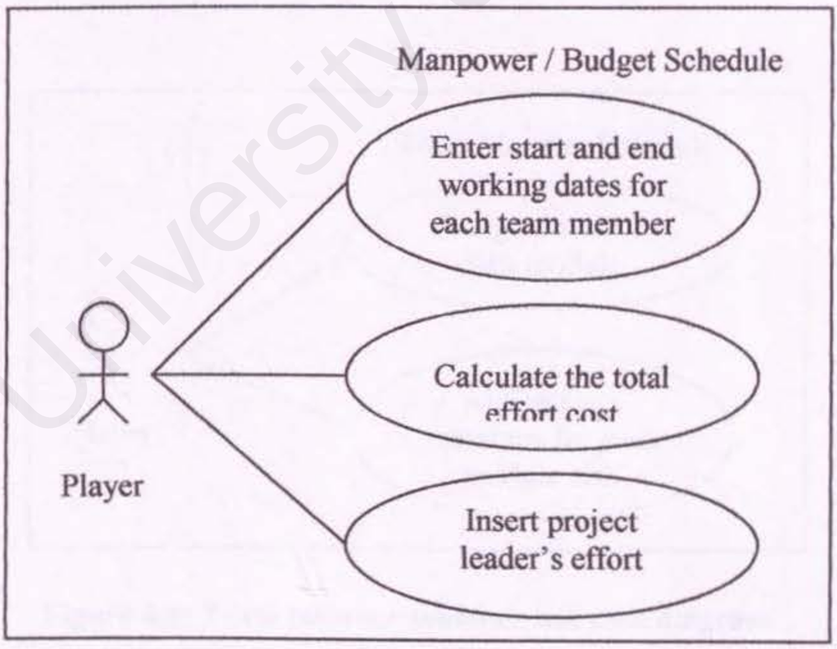

Manpower / Budget Schedule
Player
Enter start and end working dates for each team member
Calculate the total effort cost
Insert project leader's effort

Figure 4.3: Manpower/ budget Schedule use case diagram

d) Team Member Schedule

From this module, the player needs to assign tasks and staff for each simulation project module. The simulation system has 3 subsystems and each subsystem contains one or more than one modules.

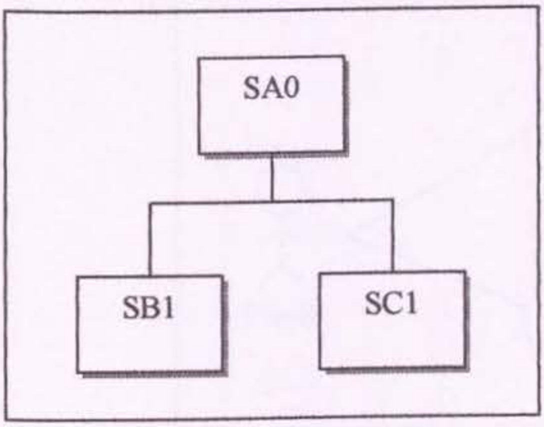


Figure 4.4: Subsystems of simulation system

Table 4.1: Scale of modules

Subsystem	Module Name
SA0	A01
	A02
SB1	B11
SC1	C11
	C12
	C13

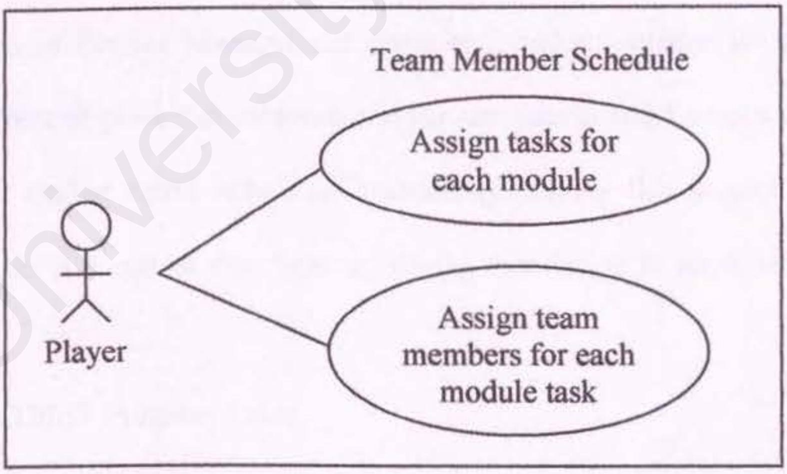

Team Member Schedule
Assign tasks for each module
Assign team members for each module task
Player

Figure 4.5: Team member schedule use case diagram

e) Project Leader Schedule

This module is to assign project leader tasks for this simulation project. The player can choose any task from the task menu and insert into task schedule sheet.

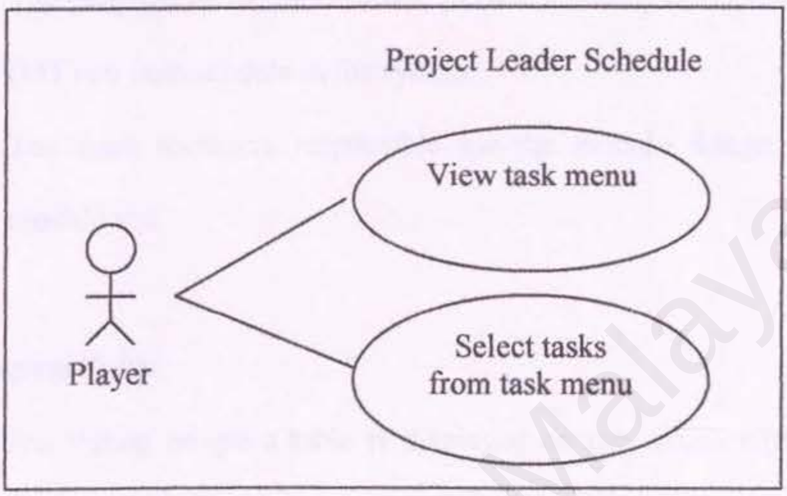


Figure 4.6: Project leader schedule use case diagram

4.1.1.2 Output Screens

The Simulation of Project Management contains 2 output screens for the player to check the progress of project development in the simulation and 1 output screen for the player to view his/her result only after completely playing this project management simulation or declare end of development during simulation at anytime. These three screens are:

- MD/CD/MT Progress Table
- IT Progress Table
- Simulation Result Screen

a) MD/ CD/ MT Progress Table

This progress table is displayed when the player needs to check the project progress. This progress table just can review and can not make any changes. The MD/CD/MT Progress Table display:

- The progress of the module design (MD), coding (CD), and module test (MT) on each module in the system.
- The team members responsible for the module design, coding, and module test.

b) IT Progress Table

Integration testing progress table is displayed on the screen when the player wants to review integration testing progress. The IT Progress Table displays:

- The number of test items are created and tested during the integration testing.
- The number of errors found through the testing process.
- The number of these errors solved through the integration test debugging task.

c) Simulation Result Screen

This screen is displayed when at the end of development. This screen displays the score of the game. The Simulation Result Screen evaluates the project leader's management of the project in terms of

- meeting the deadline (schedule)
- allocate the right person in the right place (resource)

- staying under budget (cost)
- building a quality system (software quality)

4.1.1.3 Subsystem Quality Review

In the quality review, the project leader review the current documentation of the modules in a subsystem about the number of errors found and corrected at the tasks module design, module design review, coding, coding review, module test and module test review. According to this information, the project leader can determine the quality of a subsystem.

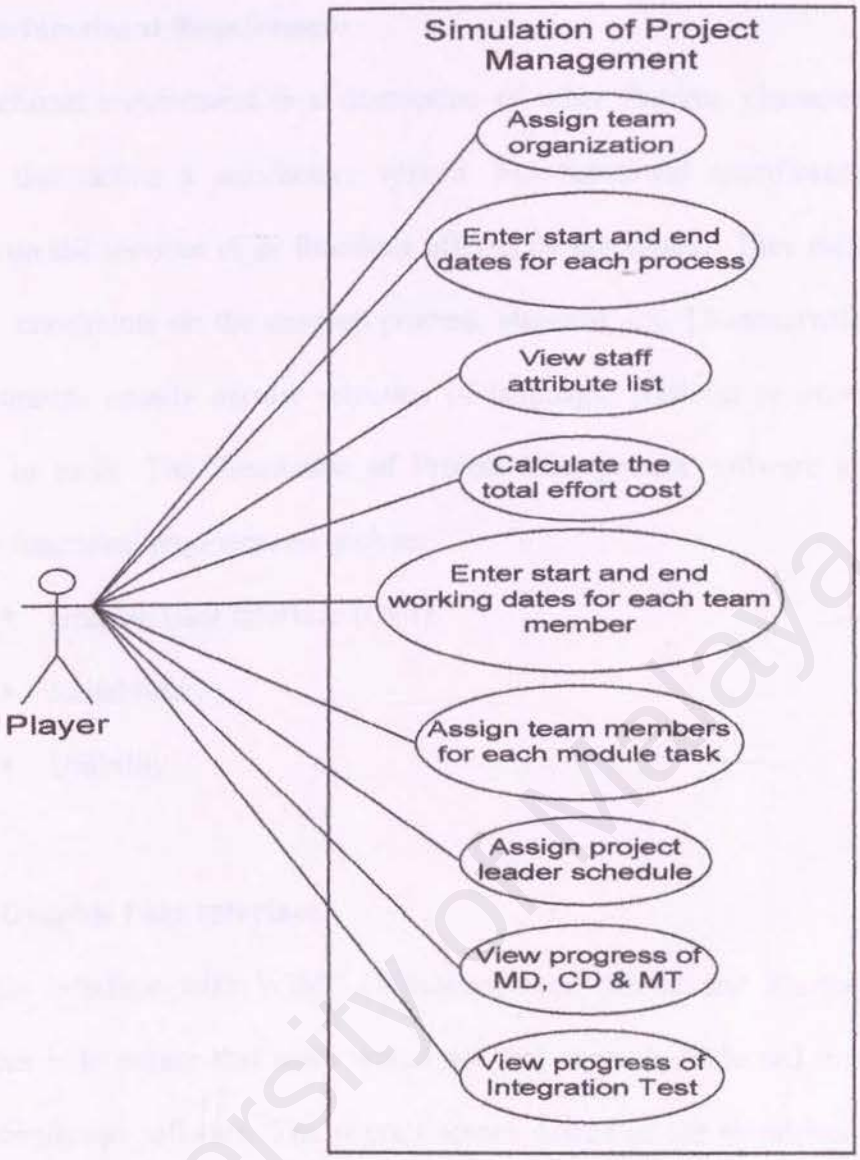

Simulation of Project Management
Assign team organization
Enter start and end dates for each process
View staff attribute list
Calculate the total effort cost
Enter start and end working dates for each team member
Assign team members for each module task
Assign project leader schedule
View progress of MD, CD & MT
View progress of Integration Test
Player

Figure 4.7: Overall software use case diagram

4.1.2 Non-functional Requirement

A non-functional requirement is a description of other features, characteristics, and constraints that define a satisfactory system. Non-functional specifications are the constraints on the services of or functions offered by the system. They include timing constraints, constraints on the develop process, standard, etc. [Sommerwille, I, 2001] These constraints usually narrow selection of language, platform or implementation techniques or tools. The Simulation of Project Management software must ensure certain non-functional requirements such as:

- Graphic User Interface (GUI)
- Reliability
- Usability

4.1.2.1 Graphic User Interface

User-friendly interface with WIMP (Windows, Icon, Menu and Pointing Device) features. This is to ensure that users would not feel uncomfortable and irritated when using the simulation software. The overall screen design of the simulation should be smart and consistent. While notification massage should be accurate, concise, clear and simple.

4.1.2.2 Reliability

The software must show system reliability so that programming task will be executed effectively and functions as intended with required precision. As a result, the software can be trusted in performing its functions and operations. Thorough testing should also be carried out to ensure the software reliability while performing continuous testing to further improve the overall trustworthiness.

4.1.2.3 Usability

The system should be ready to use and should not contain any weaknesses in the run time environment.

4.2 Software Development Tools and Technologies

4.2.1 Development Platform

❖ Windows XP Professional

Windows XP Professional is chosen as the development platform. The main reason choosing Windows XP as the development operating system because Windows XP can support Microsoft Visual Basic .Net 2003 program. Beside that, Windows XP is more powerful, secure and stable than the previous version of Windows operating system. In addition, Windows XP also has many advantages, such as:

- Superior operating system technology
- Ability to recover work
- System memory protection

4.2.2 Programming Tool

❖ Microsoft Visual Basic .Net 2003

Microsoft Visual Basic .Net 2003 is chosen as the software tool to build the Simulation of Project Management software. The main reason for choosing this software is because we want to use a more advance tool to build and develop this product. There are many advantages of Microsoft Visual Basic .Net such as:

- Features such as automatic control anchoring and docking eliminate the need for complex form-resize code.
- Side-by-side versioning enables multiple versions of the same components to exist on the same machine.
- No complicated setup scripts or interactions with the system registry are required.
- A powerful new integrated emulator also enables developers to build solutions without requiring hardware.

4.2.3 Database Management System Consideration

❖ Microsoft Access

Microsoft Access in the relational database management system (RDBMS) will be used as the system database. The reasons of Microsoft Access have been chosen are as follows:

a) Microsoft Access is more suitable for the medium and small scale application if compare with the others as Project Management Simulation Game is only a small scale application.

b) Provide easy menu driven interface.

c) Can be easily access by application developed by Visual Basic.NET by using ActiveX Data Object.NET (ADO.NET).

d) Easily to maintain as most of the people have Microsoft Access that is included in the Microsoft Office package.

4.2.4 Multimedia Tool

❖ Macromedia Flash MX

Macromedia Flash is selected as the multimedia tools. The Macromedia products should produce the multimedia features required by the system. There are enabling to create beautiful, compact and resizable animation and graphics. Furthermore, their capabilities to import by the Visual Basic by using Shockwave Flash Control for Multimedia Flash are very important.

4.3 Software Requirement

The software requirements are requirements that have to do with the software needs that must be fulfilled in order to develop or run the system.

Table 4.2: Software requirement for developer and user

Stakeholder	Minimum Requirement
Developer	❖ Microsoft Window XP Professional ❖ Microsoft Visual Studio .Net ❖ Microsoft Access ❖ Macromedia Flash MX ❖ Adobe Photoshop 7.0
User	❖ Microsoft Windows 98/Me/2000/XP ❖ Microsoft Access 2000

4.4 Hardware Requirement

The developers should have the below hardware which show at the table 4.3 in order to make sure that the developing process going smoothly. Beside that, the users system also should fulfill the requirement which show at the same table.

Table 4.3: Hardware requirement for developer and user

Stakeholder	Minimum Requirement
Developer	❖ IBM Compatible PC. ❖ Intel Pentium III 450 MHz or above. ❖ 128 MB SDRAM or above. ❖ Others standard computer peripherals such as scanner, speaker, microphone, CD-RW Burner, and etc.
User	❖ IBM Compatible PC. ❖ Intel Pentium II 300 MHz or above. ❖ 64 MB SDRAM or above. ❖ Minimum 100 MB free hard disk space. ❖ Others standard computer peripherals such as CD-ROM, speaker and mouse.

4.5 Summary

The analysis of the requirements will serve as the input for the design phase in chapter 5 (next chapter). The chapter 5 is system design.

Chapter 5: System Design

5.1 Architectural Design

The architectural design process is concerned with establishing a basic structural framework for a system. It involves identifying the major components of the system and the communications between these components. The output of the architectural design process is an architectural design document. This consists of a number of graphical representations of the system models along with associated descriptive text.

Because of RUP is the methodology to develop this Simulation of Project Management software and RUP is based on an object-oriented design structure, therefore UML is used to model the system.

5.2 Object-oriented Design

Object-oriented design is concerned with developing an object-oriented model of a software system to implement the identified requirements. The objects in an object-oriented design are related to the solution to the problem that is being solved. There are two types of design models that should normally be produced to describe an object-oriented design. These are:

- Static model – describe the static structure of the system in terms of the system object classes and their relationships.
- Dynamic model – describe the dynamic structure of the system and that show the interactions between the system objects (not the object classes). These

interactions may include the sequence of service requests made by objects and the way in which the state of the system is related to these object interaction.

5.2.1 Class Diagrams

A class diagram is a diagram that shows a set of classes, interfaces, and collaborations and their relationships. Class diagrams are used to model the static design view of a system and show the interactions between structural elements found in the system. In UML, a class is presented as a rectangle.

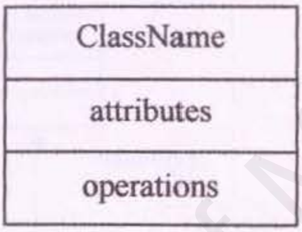


Figure 5.1: Notation of class

Every class must have a name to distinguish it from other classes. According to figure 5.1, ClassName is the name of the class. An attribute is a named property of a class that describes a range of values that instances of the property may hold. A class may have a number of attributes or no attributes at all. An operation is the implementation of a service that can be requested from any object of the class to affect behavior. At the same time, a class also may have a number of operations or no operations at all. The classes diagram for this Simulation of Project management software shown on Figure 5.2.

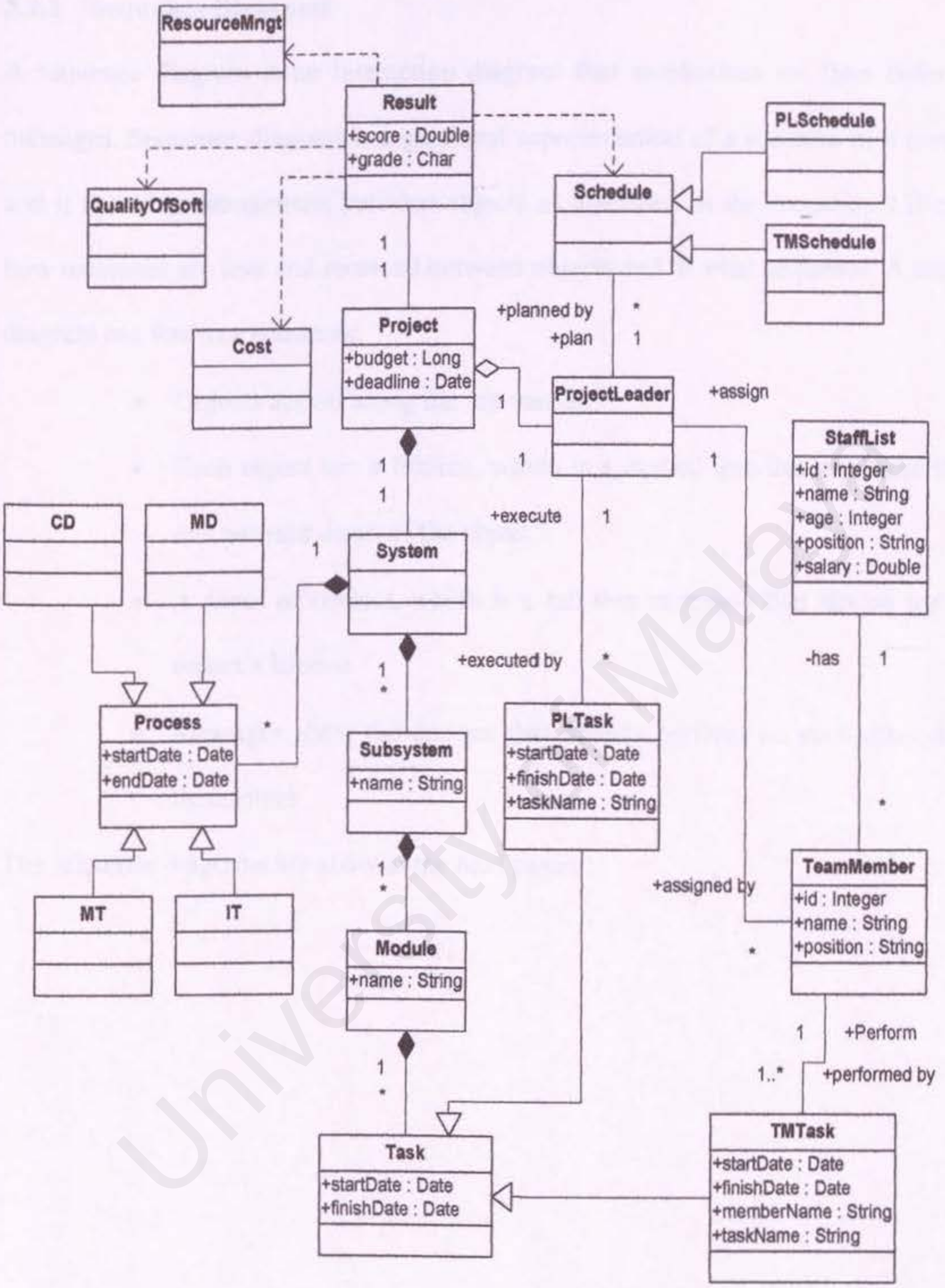


Figure 5.2: Class diagram of Simulation of Project Management

5.2.2 Sequence Diagrams

A sequence diagram is an interaction diagram that emphasizes the time ordering of messages. Sequence diagram is a graphical representation of a scenario of a user case, and it describes interactions between objects as described in the scenario. It illustrates how messages are sent and received between objects and in what sequence. A sequence diagram has four key elements:

- Objects appear along the top margin
- Each object has a lifeline, which is a dashed line that represent the life and perhaps death of the object
- A focus of control, which is a tall thin rectangle that sits on top of an object's lifeline
- Messages show the actions that objects perform on each other and on themselves

The sequence diagrams are show at the next pages.

Sequence diagram: Assign Team Organization

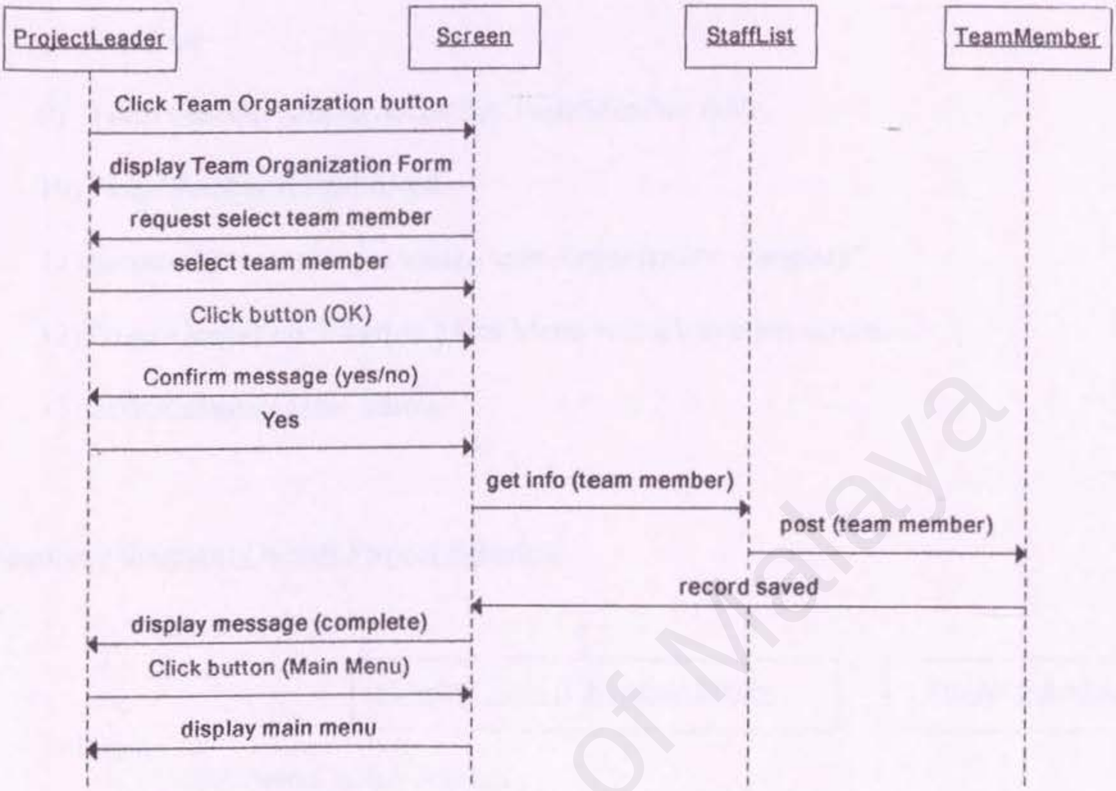


Figure 5.3: Sequence diagram of assign team organization

Explanation:

1) Project leader click Team Organization button from Main Screen.
2) Screen display Team Organization form.
3) Screen shows message to request project leader select team member.
4) Project leader select team member from staff attribute list which display on the screen.
5) Project leader click OK button after select the project team member.
6) Screen show message to confirm with the project leader.

7) Project leader click yes button.

8) System gets team member info which selected by the project leader from StaffList.

9) Team member info saves on the TeamMember table.

10) TeamMember record seved.

11) Screen shows message "assign team organization complete".

12) Project leader click button Main Menu to back to main screen.

13) Screen display Main Menu.

Sequence diagram: Overall Project Schedule

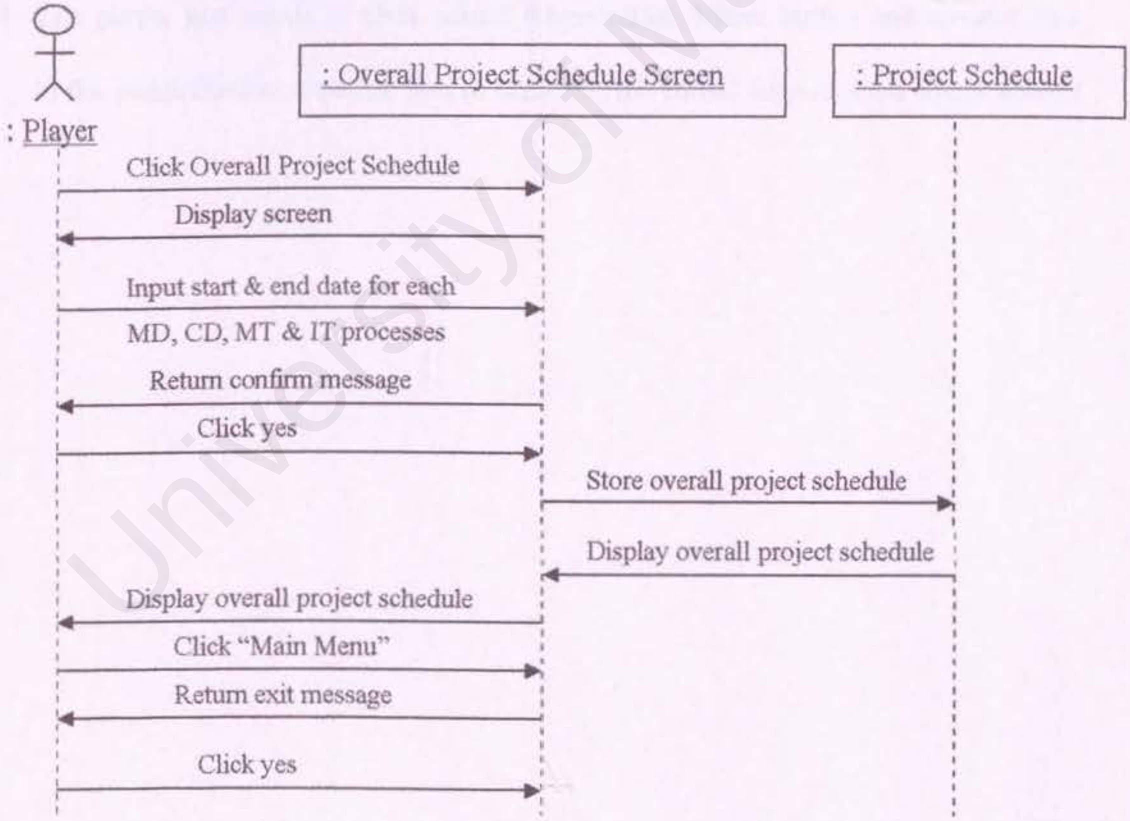

: Player
: Overall Project Schedule Screen
: Project Schedule
Click Overall Project Schedule
Display screen
Input start & end date for each
MD, CD, MT & IT processes
Return confirm message
Click yes
Store overall project schedule
Display overall project schedule
Display overall project schedule
Click "Main Menu"
Return exit message
Click yes

Figure 5.4: Sequence diagram of assign Overall Project Schedule

Explanation:

1) Player clicks the Overall Process Schedule button in the main screen in order to enter into the Overall Process Schedule input screen.
2) After that, the player has to input the start & end date for Module Design, Coding, Module Testing & Integration Testing processes into the table on the screen & click OK button.
3) Then, the system will display a confirmation message box to the player.
4) After the player had click "Yes" button, the system will store the overall process schedule into the database and display the estimated overall process schedule in a Gantt chart format to the player.
5) The player just needs to click Initial Registration Menu button and answer Yes to the confirmation message box to return to the Initial Registration Menu screen

Sequence diagram: Assign Manpower/Budget Schedule

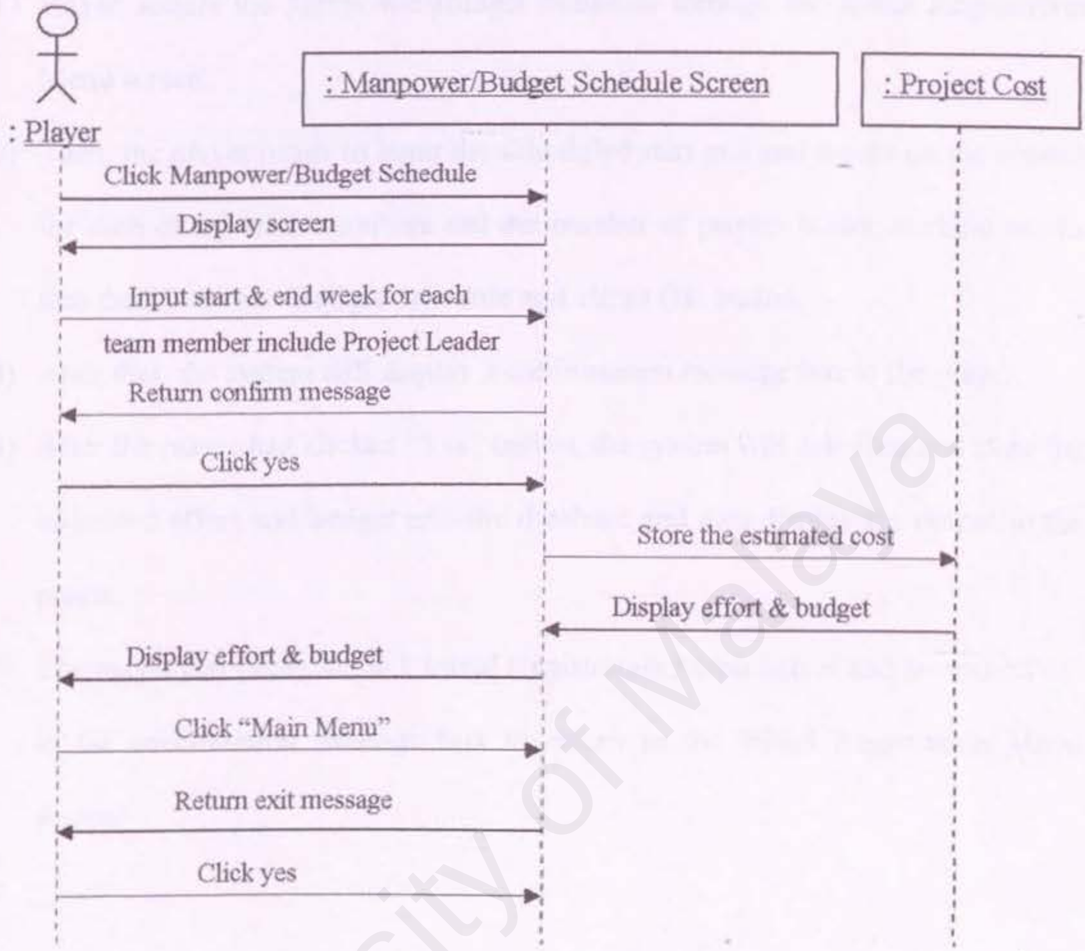


Figure 5.5: Sequence diagram of assign Manpower/Budget Schedule

Explanation:

1) Player selects the Manpower/Budget Schedule through the Initial Registration Menu screen.
2) Then, the player needs to input the scheduled start and end weeks on the project for each of the team members and the number of project leader working weeks into the Manpower/Budget Schedule and clicks OK button.
3) After that, the system will display a confirmation message box to the player.
4) After the player had clicked "Yes" button, the system will calculate and store the estimated effort and budget into the database and also display the output to the player.
5) The player just needs to click Initial Registration Menu button and answer "Yes" to the confirmation message box to return to the Initial Registration Menu screen.

Sequence diagram: Assign Team Member Schedule

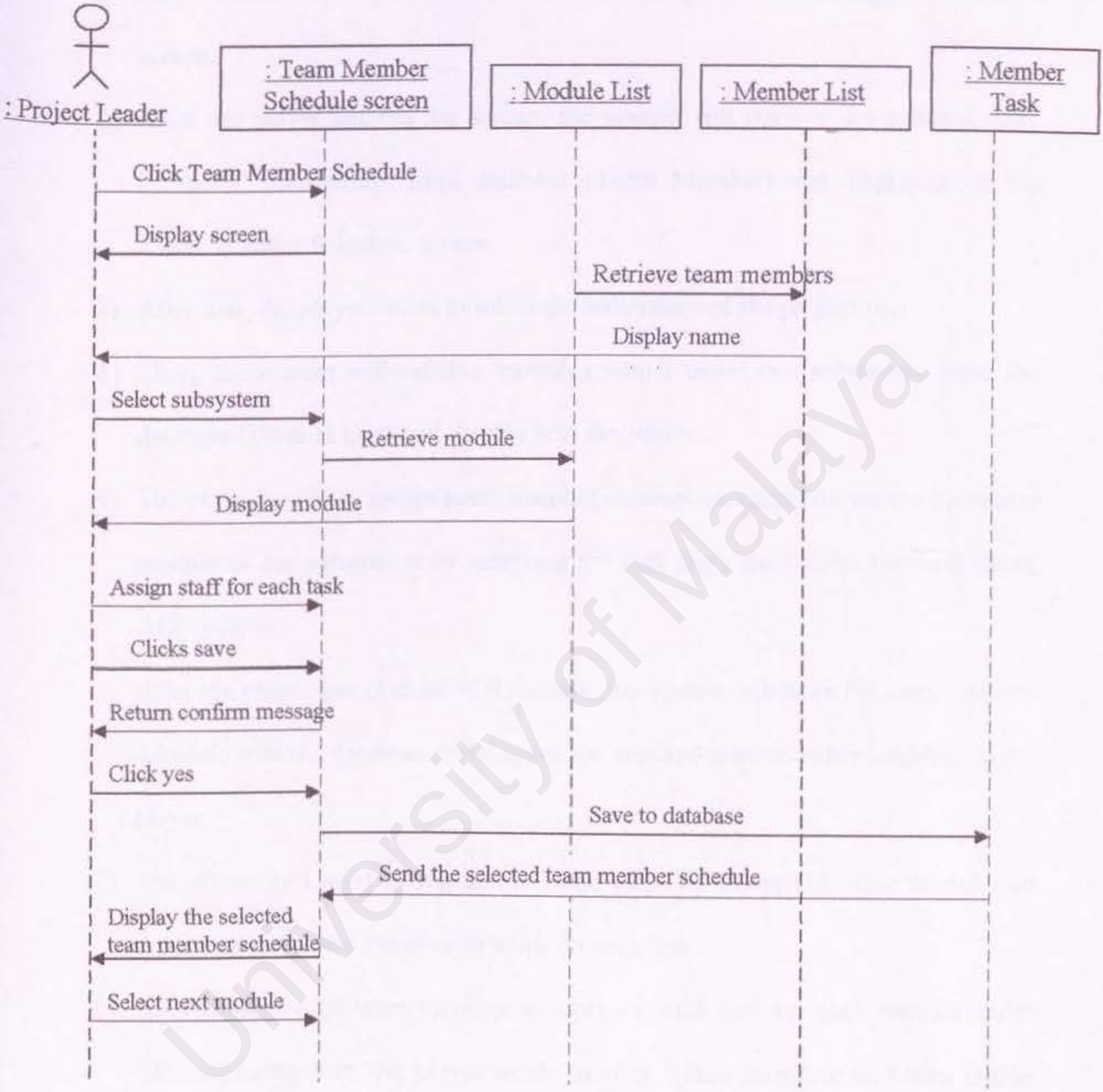
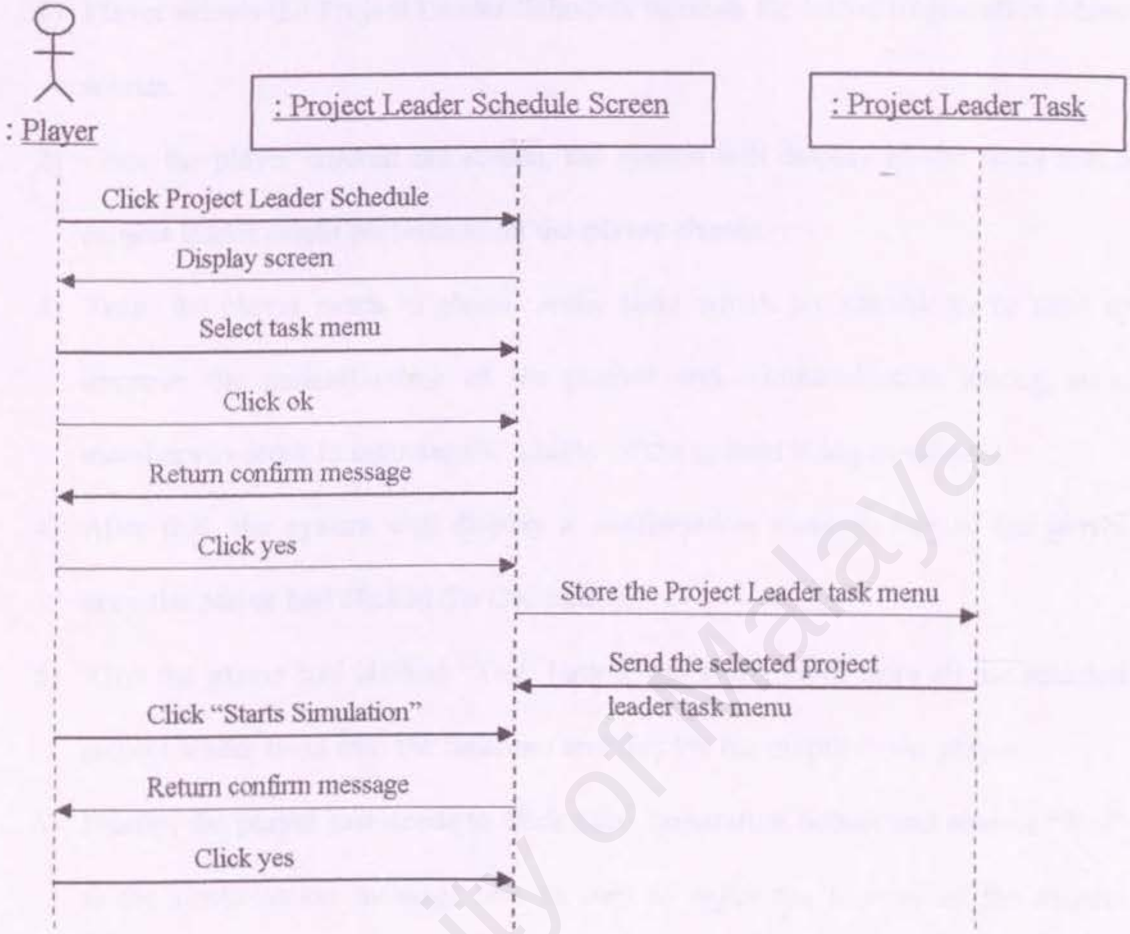


Figure 5.6: Sequence diagram of assign Team Member Schedule

Explanation:

1) Player selects the Team Member Schedule through the Initial Registration Menu screen.
2) Once the player entered the screen, the system will retrieve the selected team members' information from database (Team Member) and display it on the Team Member Schedule screen.
3) After that, the player needs to select the subsystem of the project first.
4) Then, the system will retrieve modules which under that subsystem from the database (Module List) and display it to the player.
5) The player needs to assign team member to work on each task on the particular module of the subsystem by selecting the task from the combo box and clicks ADD button.
6) After the player had clicked ADD button, the system will store the team member schedule into the database and display the selected team member schedule to the player.
7) The player just need to repeat the same steps by select the other modules in order to assign team member to work on each task.
8) After finish assign team member to work on each task for each module under different subsystem, the player needs to click Initial Registration Menu button and answer "Yes" to the confirmation message box to return to the Initial Registration Menu screen.

Sequence diagram: Assign Project Leader Schedule



: Player
: Project Leader Schedule Screen
: Project Leader Task
Click Project Leader Schedule
Display screen
Select task menu
Click ok
Return confirm message
Click yes
Store the Project Leader task menu
Send the selected project leader task menu
Click "Starts Simulation"
Return confirm message
Click yes

Figure 5.7: Sequence diagram of assign Project Leader Schedule

Explanation:

1) Player selects the Project Leader Schedule through the Initial Registration Menu screen.
2) Once the player entered the screen, the system will display all the tasks that a project leader might perform to let the player choose.
3) Then, the player needs to choose some tasks which are suitable to be used to improve the understanding of the project and communication among team members in order to increase the quality of the system being developed.
4) After that, the system will display a confirmation message box to the player once the player had clicked the OK button.
5) After the player had clicked "Yes" button, the system will store all the selected project leader tasks into the database and display the output to the player.
6) Finally, the player just needs to click Start Simulation button and answer "Yes" to the confirmation message box to start to enjoy the journey of the Project Management Simulation!

5.2.2.1 State Diagrams

A state diagram shows an object's state machine. It is used to model the dynamic behavior of a particular object of a system. It illustrate an object's life cycle – the various states that an object can assume and the events that cause the object to transition from one state to another state. Figure 5.3shows the state diagram of the Cost class.

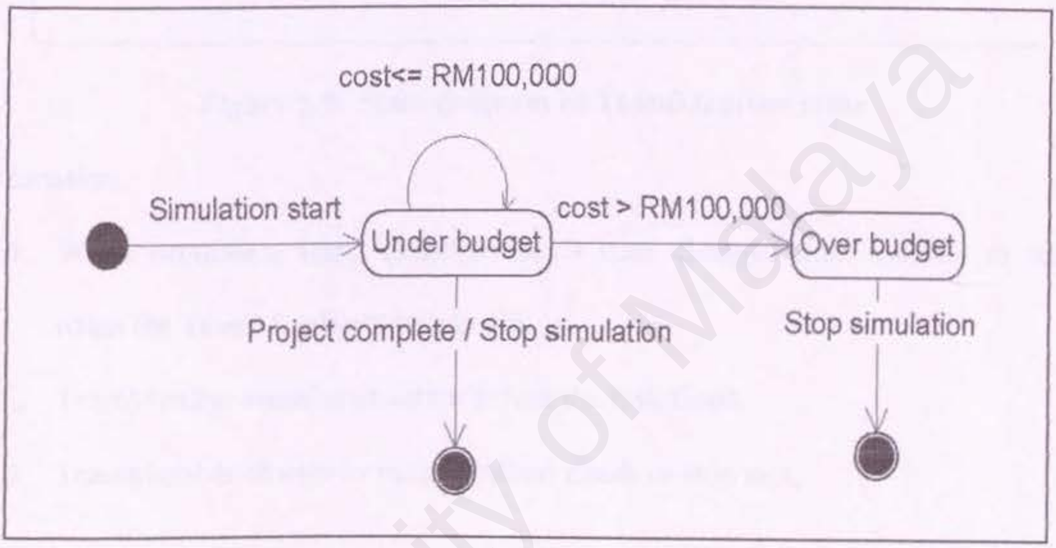


Figure 5.8: State diagram of Cost class

Explanation:

1. When new simulation start, the project cost is under budget.
2. If the project cost is equal or less than RM100,000, the state of project cost is under budget.
3. If the project cost more than RM100,000, the state of project cost is over budget.

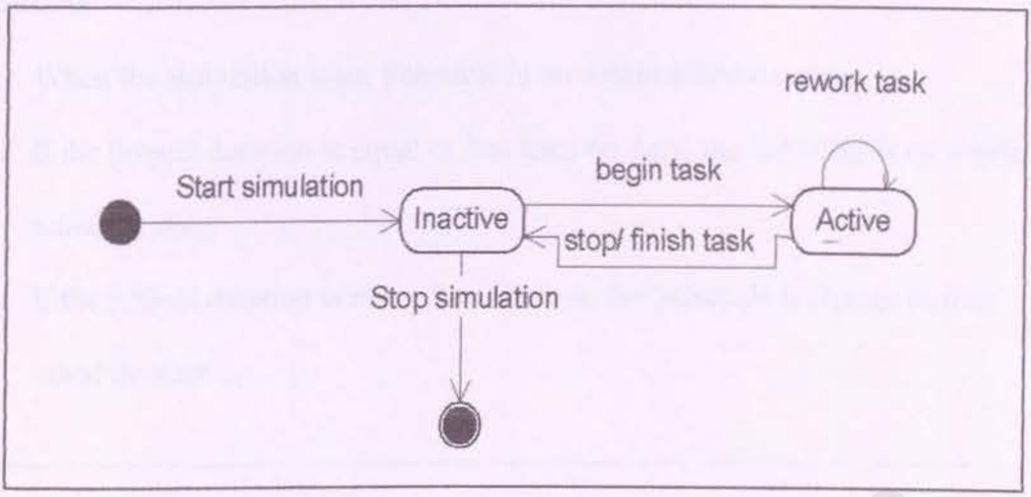


Figure 5.9: State diagram of TeamMember class

Explanation:

1. When simulation start, TeamMember's state changes from inactive to active when the TeamMember's task begin.
2. TeamMember remains at active before the task finish.
3. TeamMember change to inactive when finish or stop task.

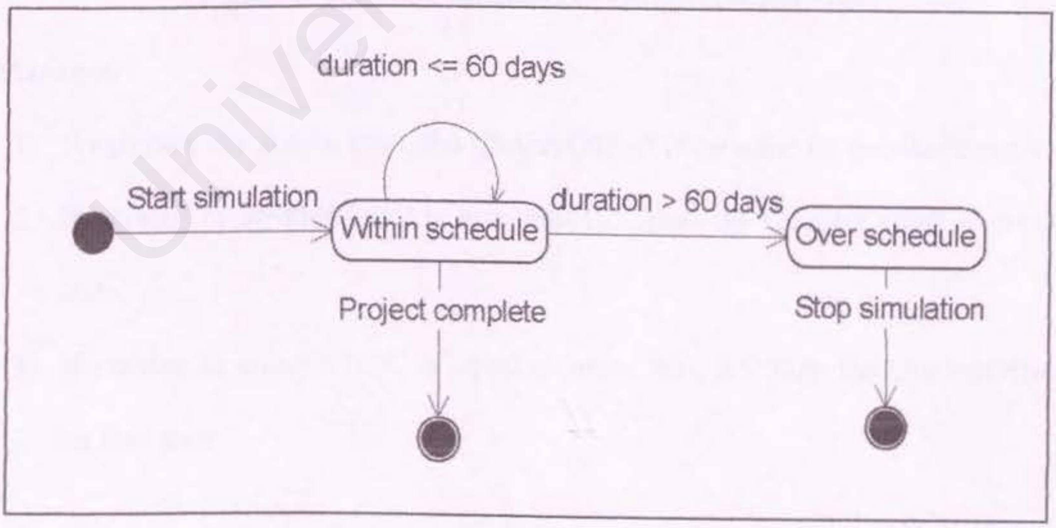


Figure 5.10: State diagram of Schedule

Explanation:

1. When the simulation start, Schedule is on within schedule state.
2. If the project duration is equal or less than 60 days, the Schedule is on within schedule state.
3. If the project duration is more than 60 days, the Schedule is change to over schedule state.

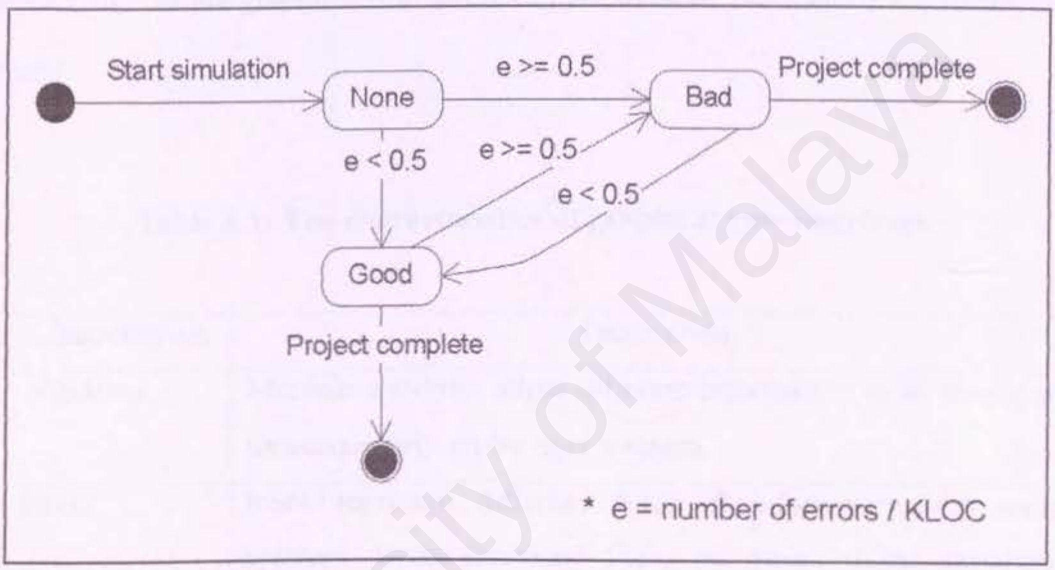


Figure 5.11: State diagram of QualityOfSoft class

Explanation:

1. When new simulation start, the QualityOfSoft is on none performance state.
2. If number of errors/KLOC is less than 0.5, than the QualityOfSoft is on Good state.
3. If number of errors/KLOC is equal or more than 0.5, than the QualityOfSoft is on Bad state.

5.3 User Interface Design

User interface design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventual presentation of desired outputs and inputs. Most of today's user interfaces are graphical. The basic structure of the graphical user interface (GUI) is provided within either the computer operating system or in the internet browser. The characteristics of the graphical user interfaces are show on the Table 5.1. [Sommerville, I, 2001]

Table 5.1: The characteristics of graphical user interfaces

Characteristic	Description
Windows	Multiple windows allow different information to be displayed simultaneously on the user's screen.
Icons	Icons represent different types of information. On some systems, icons represent files; on others, icons represent process.
Menus	Commands are selected from a menu rather than typed in a command language.
Pointing	A pointing device such as mouse is used for selecting choices from a menu or indicating items of interest in a window.
Graphical	Graphical elements can be mixed with text on the same display.

5.3.1 Main Page

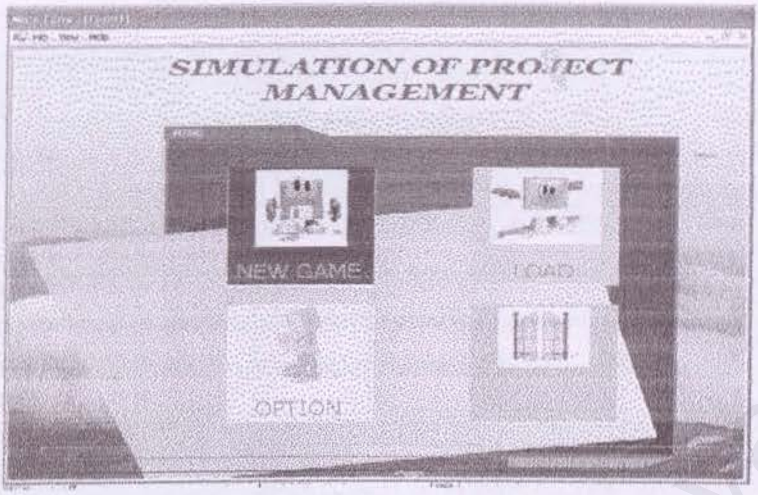
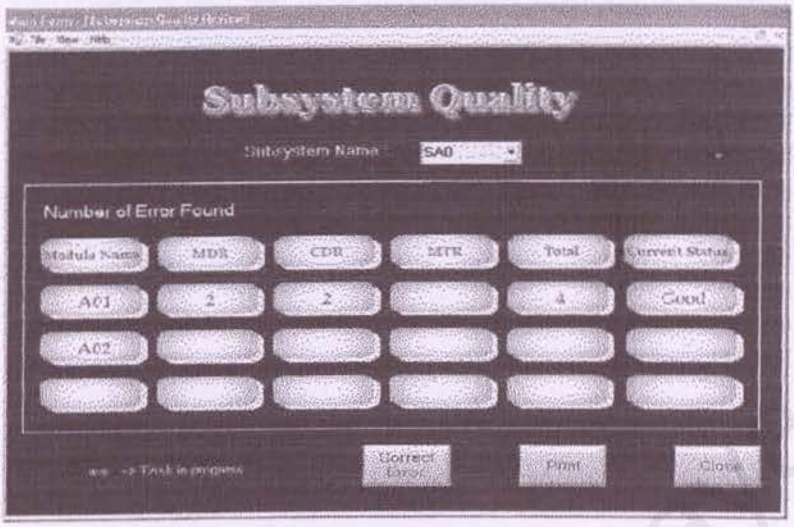


Figure 5.12: Graphical user interface for main page

This main page has four labels; there are "New Game", "Load", "Option" and "Exit". Users click the "New Game" label to access the player information form, which allow new player to register before start the simulation game. Users click the "Load" label is to continue the previous simulation. "Exit" label is to quit the simulation.

5.3.2 Subsystem Quality Review Form



Subsystem Quality

Subsystem Name: SA0

Number of Error Found

Modula Name	MDR	CDR	MTR	Total	Current Status
A01	2	2		4	Good
A02					

Figure 5.13: Subsystem Quality Review form

This page is a subsystem quality review form. The combo box at the top-center is easier the player to select the desired subsystem for review. The numbers in labels under the "MDR", "CDR", "MTR" and "Current Status" is the number of errors found and current status of quality for each module in the selected subsystem. Table 5.2 shows the description of the 3 buttons which at the bottom page.

Table 5.2: Button descriptions

Button	Description
Correct Error	Player clicks this button to correct the error.
Print	Player clicks this button to print the subsystem quality report.
Close	Player clicks this button to close this form.

5.4 Summary

These design specification will use to implement the software of Simulation of Project Management. Thus, the software is hoped to be implemented according to the system design in order to produce a system that is high quality and fulfill the system requirement.

Chapter 6: System Implementation

6.1 Introduction

During the system implementation phase, the major tasks are coding and debugging. Beside write the program source code, there are also need to solved the problems and errors found in the coding stage. Programming language characteristics and coding style will affect the system quality maintainability.

Coding step translates a detail design representation of software into a programming language realization. In a software project, the requirements analysis, system design and implementation phases do not have a clear boundary. Each phase tend to overlap one another. This phase at time involves some modifications to the previous design.

6.2 Database Development

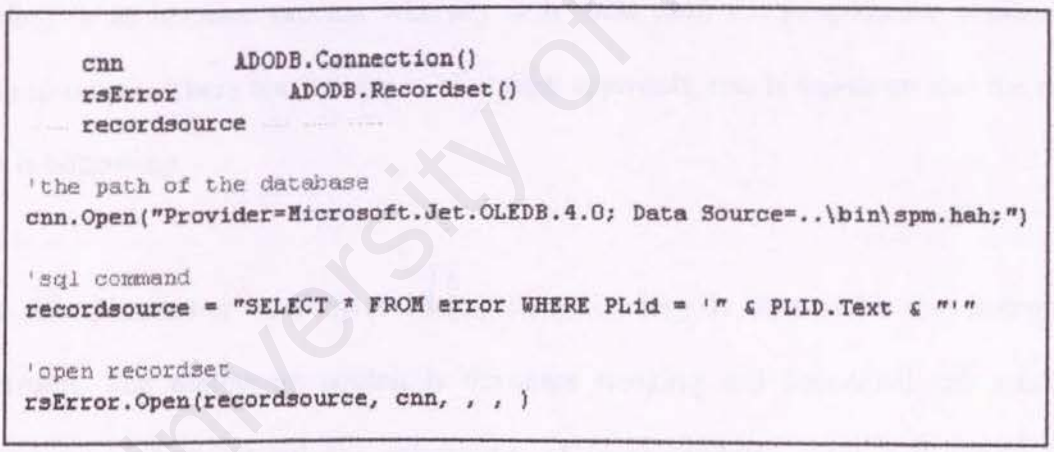
Firstly, build a system's database based on the logical data model for the system created during the system design phase. The database is developed by Microsoft Access 2000. The database name for this database which created for this system is called SPM. The entire tables then are created by specifying all the fields for each table and the field properties. A primary key is allocated for each table in the database.

6.2.1 Database connection

All communications with a database take place through an open connection. Therefore, before any data can be access from the database, a connection with the database must be opened. The ActiveX Data Object (ADO) model defines the base objects that developer work with to perform specific actions in developer ultimate request to access data. The base objects, such as Connection, Recordset can be created independently of one another and are related in a hierarchical sense.

Connection Object represents the actual session established with the database. There are two methods; Open() and Close() that used to open and close the database connection. Recordset Object represents any group of records whether it is the result of the query or the entire contents of a table. This is the object use for almost all data access.

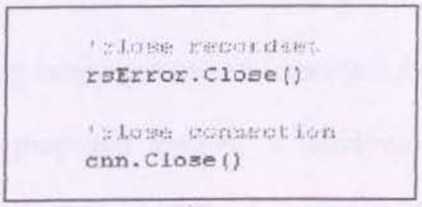
The code to create the connection to database for the Simulation of Project Management is shown as below:



```
    cnn          ADODB.Connection()
    rsError          ADODB.Recordset()
    recordsource

'the path of the database
cnn.Open("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=..\bin\spm.hah;")

'sql command
recordsource = "SELECT * FROM error WHERE PLid = '" & PLID.Text & "'"

'open recordset
rsError.Open(recordsource, cnn, , , )
```

Figure 6.1: Source code of database connection

Once have finished with a connection, the connection and the recordset need to close in order to free-associated system resources. The codes are shown as below:



```
'close recordset
rsError.Close()

'close connection
cnn.Close()
```

Figure 6.2: Source code of close database connection and record set

6.3 Application Development

The Simulation of Project Management application development involves code generation that translates all the algorithms into Microsoft Visual Basic.NET programming language instructions.

6.3.1 Coding Methodology

Coding is an iterative process whereby it is done until the programmer obtains the desired output. There are two types of coding approach; one is top-down and the other one is bottom-up.

For this Simulation of Project Management, coding is done with the bottom-up approach. The bottom-up coding is develops working and functional sub modules before proceeding toward the integration of large modules. This will continue to progress upward like wise until the overall system is modeled. The advantages of this approach are: testing can be carried out on some of the functions as soon as it is completed, and critical functions can be coded first in order to do the efficiency test.

6.3.2 Coding Tools

The riches of programming languages are an important factor to produce a high quality system. To develop the proposed system, a windows-based GUI (Graphical User Interface) programming tool from scratch was needed. Therefore, Visual Basic .NET was used. All the source codes are kept in files in a directory which created using VB.NET.

6.3.3 Coding Style & Practice

Coding style is an important attribute of source code and it determines the intelligibility of a program. With a clear and systematic coding style, it helps the programmer to see the codes clearer and easier. It will also help the programmer in maintaining and also debugging the application.

There are many rules in defining a good coding practice. For Simulation of Project Management, a lot of the precaution steps have been taken and implemented during the coding process. A good coding practice is mainly focusing on the system consistency, maintainability and readability. The followings show some of the coding practices for Simulation of Project Management.

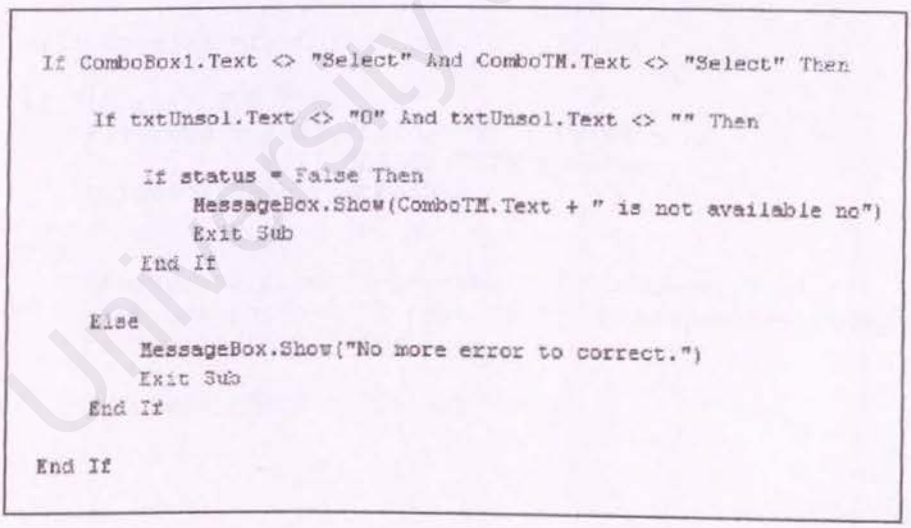
i. Meaningful Variables Names

- Choosing the meaningful variable names can help developer to easily trace flaws in programs and also enable references to be done easily. It can also help to reflect their usage as well as meaning.

- The standard of variable names is use the camel case. This is by starting the variable name with a lower-case letter and continuing in lower case, except for the first letter of each concatenated word, which should be in upper case.
- For example, "numError" which means the variable contains the number of errors.

ii. Formatting to Enhance Understanding

- A code that is written without proper formatting will function as work as a formatted code. But, this can make the developer difficult to find out the cause of error.
- The format of statements like the indentation and spacing of statements can reflect the basic control structure of the coding.
- A sample of a formatted code is shown as below:



```
If ComboBox1.Text <> "Select" And ComboTM.Text <> "Select" Then

    If txtUnsol.Text <> "0" And txtUnsol.Text <> "" Then

        If status = False Then
            MessageBox.Show(ComboTM.Text + " is not available no")
            Exit Sub
        End If

    Else
        MessageBox.Show("No more error to correct.")
        Exit Sub
    End If

End If
```

Figure 6.3: Formatted code

6.4 Coding example

The coding shown in below is takes from "frmSolITerror.vb" to perform the functions of display the selected team member and display the number of error found in the Integration Test task.

```
Dim cnn As New ADODB.Connection()
Dim rsIT As New ADODB.Recordset()
Dim recordsource As String

txtUnsol.Text = ""
lblStatus.Text = ""
lblStandar.Text = ""

'display selected team member function
frmCorrectTask.DisplayTeamMember()

'open connection
cnn.Open("Provider=Microsoft.Jet.OLEDB.4.0; Data _
          Source=..\bin\spm.hah;")

recordsource = "SELECT IT, ITB FROM ITerror " & _
               "WHERE Plid = '" & PLID.Text & "' " & _
               "AND subName = '" & ComboBox1.Text & "' "

rsIT.Open(recordsource, cnn, , , )

If Not rsIT.EOF Then
    ErrorNum = rsIT.Fields("IT").Value - 
               rsIT.Fields("ITB").Value
    txtUnsol.Text = ErrorNum

Else
    MessageBox.Show("Subsystem " + ComboBox1.Text + " _
    have not perform IT task.", "", MessageBoxButtons.OK, _
    MessageBoxIcon.Information)

    ComboBox1.Text = "Select"

End If
```

Figure 6.4: Source code from "Correct Integration Test Error" form

6.5 Summary

System implementation is a process that translates a detail design representation of software into a programming language realization. Coding convention such as program labeling, naming, conversion, comments, and indention should be adhered too. Codes should be easy to understand, easily revised or corrected and readable to facilitate maintenance.

The codes should be able to handle user error by responding appropriately, perhaps with a diagnostic error message and system failure should not result. A good programming practice enhances the program readability, understandability and cleanliness. The source code also should simple, straightforward manner, precede a comment describing the purpose of the functions and indent statement inside the bodies of event procedures.

The whole system requirements including software development tools and hardware specification are part of the important key to make the development of the system successful.

Chapter 7: Testing

7.1 Introduction

The test workflow in RUP is involves a set of activities that validate and verify the developed system, whether the system meet the system requirements or not. During the developing system, on-going testing has been carried out. Validation and verification are actually two different activities. Validation refers to a set of activities that ensuring the software have been built traceable to user requirement. Verification refers to the set of activities that ensure the software is correctly implemented.

The purposes of testing are to:

- Plan the test required in iteration, including integration test and system test. Integration tests are required for every build within the iteration, where as system tests are required only at the end of the iteration.
- Design and implement the tests by creating test cases that specify what to test, creating test procedures that specify how to perform the tests and creating executable test components to automate the test if possible.
- Perform the various tests and systematically handle the result of each test.

There are four basic concepts related to software testing:

i. Error detection

   - o Involves identifying errors either by inspection, walkthroughs or other type of error detection approach.

ii. Error removal

   - o Involves debugging and removing identified error.

iii. Error tracking

- Involves finding and correcting the cause of the error.

iv. Regression testing

- Involves testing of changed and affected components.

Figure 7.1 shows the testing steps that perform in the testing of Simulation of Project Management program.

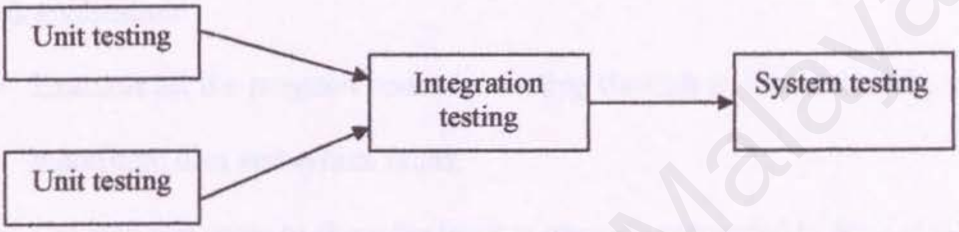


Figure 7.1: Testing steps

7.2 Testing strategy

Testing strategies in this Simulation of Project Management are involved unit testing, integration testing and system testing. The objective of unit testing is to ensure all coding of the function implement the design correctly. The integration testing is done on the groups of integrated modules to verify that the system components work together as described in the system and program design specification. Finally, system testing verifies that overall system is functioning properly and system performance and objective are achieve.

7.2.1 Unit testing

Unit testing is a process of test the individual component such as the procedure, function or the object class for particular unit in the system without affects other system components. Unit testing is similar as to find faults in the components and to ensure that all the components within the system are operated correctly as what the system user desire. During implement phase that the system is in progress with the source-code generation, the unit testing was carried out. There were several steps being carried out for this application:

- Examine all the program codes by reading through it. Consecutively, try to spot algorithm, data and syntax faults.
- Develop test cases to show the input is properly converted to desired output.
- Boundary conditions are tested to make sure the functions run at boundaries established for limiting or restricting process.
- Use the "break point" function in the Visual Basic .Net application to find out the cause of the error when error detected.
- Test all errors handing paths.

7.2.2 Integration testing

This process is performed after all the objects, components and individual sub modules have passed local unit tests. This is the process of verifying that the system components work together as described in the system and program design specifications. System with integrated sub modules and modules must go through integration testing to ensure valid linking and dynamic relationship establishes between modules of the whole system and between sub modules contained in all individual modules is no different. This is a systematic technique for constructing the program structure while at the same time conducting testing to uncover errors associated with interfacing.

Bottom up testing approach has been used to test the integration of all components and modules. Each component at the lowest level of the system hierarchy was tested individually. After that, the next components to be tested were those that all the previously tested components combine with the modules at the higher levels of the system.

7.2.3 System testing

System testing is a series of different tests which primary purpose is to fully exercise the computer-based system. It is designed to reveal bugs not possible to individual components or to interaction between components and modules. System has been tested thoroughly to ensure that the simulation run smoothly. There were involved functional testing and performance testing for the system testing of the Simulation of Project Management.

7.2.3.1 Functional testing

System testing begins with function testing which is based on the functional requirements. A function test checks that the integrated system perform its function as specified in the system analysis. In Simulation of Project Management, functional testing will be carried out on five input forms and four output forms.

Table 7.1: Functional testing evaluation

Aspects	Tested	Evaluated
Overall Project Schedule		
Display Gantt chart according to the inserted data.	√	Good
Can insert new data and update existing data.	√	Good
Data of "start day" and "end day" control by the interdependency function.	√	Good
Team Organization		
Display all staff profile information	√	Good
Click "next" button to view the next staff information and click "pervious" button to view pervious staff information.	√	Good
Can review and select team member from the staff profile	√	Good
Can remove the selected team member from combo box.	√	Good
Manpower/ Budget Schedule		
Display selected team member into combo box.	√	Good
Display budget according to the position into bar chart.	√	Good
Display each team member assigned working day.	√	Good
Calculate the total of estimated budget.	√	Good
Team Member Schedule		
Assign team member for each module.	√	Good
Assign start day and end day for each task.	√	Good
Project Leader Schedule		
Add and remove task from the project leader schedule table.	√	Good
Display 60 working days and task name in the project leader schedule table.	√	Good
Enable to add working day when simulation days exceed 60 days.	√	Good
Subsystem Quality Review		
Display updated record.	√	Good
Print report.	√	Good
MD/CD/MT Progress Table		
Display updated record.	√	Good
Print report.	√	Good

IT Progress Table		
Display updated record.	√	Good
Print report.	√	Good
Simulation Result		
Display all the simulation result	√	Good
Print report	√	Good

7.2.3.2 Performance testing

Performance testing compares the integrated components with the non-functional system requirement. The non-functional that have been stated during system analysis and design phases will be tested one by one with all function. Thus, the performance test will be focus on the aspect of graphic user interface (GUI), reliability and usability.

Table 7.2: Performance testing evaluation

Non-functional aspects	Tested	Evaluated
Attractiveness	√	Fair
Readability of text	√	Good
Form layout / design	√	Poor
Error handle completely	√	Fair
Run application successfully	√	Fair

7.3 Summary

Through the testing stage, there were a few errors being found in the newly developed Simulation of Project Management program. The errors were mostly from the programming coding and the functional design of the system. Those errors caused the system run incorrectly and also invoked bugs and problems to the other functions of the system. Therefore, testing is one of the most important stages in developing of Simulation of Project Management.

There are many techniques that can be used to test the code components. The goal of testing in Simulation of Project Management is to find faults, not to prove correctness. Indeed, the absence of faults does not guarantee correctness.

A complete system is the system that is without any fault and can be function as well as expected by the system user.

Chapter 8: System Evaluation

8.1 Introduction

Evaluation is the ultimate phase of developing a system and an important phase before delivery the system to the end users. Evaluation was related to user environment, attitudes, information priorities and several other concerns that are to be considered before effectiveness can be concluded. During the system evaluation phase, the system's effectiveness and efficiency was evaluated. This chapter analyzes the system and identifies its strengths and limitations and also explains the problem encountered and the solutions for the system.

8.2 Problem Encountered & Solutions

- **Unable to decide on the development tools**

Choosing the right and most suitable hardware and software was indeed a very big dilemma. The operating system and programming tool that was chosen during the system analysis phase in the previous semester was Microsoft Windows XP Professional and Visual Basic.NET respectively. This was because they are newer technologies with better features and they were more flexible especially to develop a windows-based application like the Simulation of Project Management.

- **Inadequate experience in database design**

Due to insufficient experience in creating a good database, the database design had a lot of changes before the final product was completed.

- **Unfamiliar with certain development tools**

My inadequacy with the Visual Basic.NET programming tools was quite a drawback at first. Thus, need to spend a lot of time to learn the tool itself before building the system.

- **Slow processing time**

Microsoft Visual Basic.NET is a graphical-oriented programming language. It required more memory to compile and execute the application.

8.3 System Strengths

- **Simple and user friendly interface with a consistent looks and feel**

The Simulation of Project Management interface is simple and consistent. The user can familiarize themselves with the application within minutes.

- **User and password validation**

This system is a password-protected application for the player. Therefore the security feature ensures that unauthorized players are prohibited from playing the simulation.

- **Display process messages**

There are a lot processes between the system and its database. Therefore, it is important to inform player that every action taken will display messages to notice the player. For example: "Check progress stop" message will be displayed for player knowledge when the player need to perform the check progress task.

- **Reliable system with effective error recovery**

To avoid run time error, this system is developed with error handling. Error message will be displayed when system encounters exceptions and it will not terminate suddenly. For instance, if the player inserts a wrong username and password, the system notifies the player with a message "Invalid username and password".

- **Tutorial as a guideline for player**

The tutorial menu provides the guidance for the user to know how to use the system. Simple explanation on its features helps new players to learn and use the application instantly.

8.4 System Limitation

- **Non-portability**

This system only can run on Windows operating system.

8.5 Future Enhancements

- **User Interface**

The user interface could be given a face-lift or more 3D animation to give the simulation a much more attractive appearance and professional-look.

- **More Functions**

The Simulation of Project Management program could be added more functions to make this application become more advance. For instance, a function to display a calendar for the player to view the estimated schedule instead of the continuous working days would be an advantage.

- **Provide Resources**

A help file that containing links to project management web sites would benefit the player. It would help them to learn more about project management and become more confident to score high mark for the Simulation of Project Management.

8.6 Knowledge and Experienced Gained

This project had provided a great deal for knowledge and experience to me and gave me great opportunity to learn how to apply the knowledge and self-improvement. During the system development, I learn and use the programming tools and application such as Visual Basic. NET, Microsoft Access, Ulead PhotoImpact and Macromedia Flash MX. Beside that, I also manage to identify my weaknesses in same application development technologies and improve myself in whatever that was lacking to accomplish the desired system successfully.

8.7 Conclusion

The Simulation of Project Management is a fully stand-alone application and is a simple simulation program. Basically, the project has accomplished the listed prerequisites and had achieved all of the objectives of the project. The Simulation of Project Management is stable and had provided tutorial guideline to the user. Therefore, the player should not find any difficulty in using this system.

References

Ace Project. *http://www.aceproject.com.* Access date: August 2, 2003. Websystems, Inc.

Deitel & Deitel. 2001. *C How To Program.* 3rd Edition. New Jersey, Prentice Hall, Inc.

Deitel & Deitel. 2002. *Java How To Program.* 5th Edition. New Jersey, Prentice Hall, Inc.

Macromedia. *http://www.macromedia.com.* Last updated: Jul 23, 2003. Macromedia Incorporation.

Product Overview for Visual Basic .Net 2003.

http://msdn.microsoft.com/vbasic/productinfo/overview/default.aspx.

Accessed date: August 8, 2003. Microsoft Corporation.

Project Management Simulation Participant's Handbook. 1988. Fujitsu Corporation.

Rational Unified Process: Best Practices for Software Development Teams

http://www.augustana.ab.ca/~mohrj/courses/2000.winter/csc220/papers/rup_best_practices/rup_bestpractices.html. Accessed date: Aug 6, 2003. Rational Software Corporation.

Rational Unified Process for outsourced software development: *http://www.everlastingtec.com/m_rup.php.* Accessed date: August 8, 2003. Everlasting Technologies Inc.

Sommerville, I. 2001. *Software Engineering.* 6th Edition. Harlow, Addison Wesley, Ltd.

Young, Trevor L. 1998. *How to be a better -- project manager.* New Delhi, Kogan Page.

Bibliography

Bennet P. Lientz, Kathryn P. Rea. 1998. *Project Management for the 21st Century*. San Diego, Academic Press.

David L.Olson. 2000. *Introduction to Information Systems Project Management*. New York, Irwin/McGraw-Hill.

Jeffrey L. Whitten, Lonnie D. Bentley, Kevin C. Dittman. 2001. *Systems Analysis and Design Methods. 5th Edition.* Singapore, Irwin/McGraw-Hill.

Microsoft Visual Studio .Net. *http://msdn.microsoft.com/vstudio.* Accessed date: August 8, 2003. Microsoft Corporation.

Rational Unify Process (RUP), A Software Process Knowledge Base. *http://ref.cern.ch/CERN/CNL/2002/001/SDT_RUP*. Last Updated: March 28, 2002. CERN 2002 -- European Organization for Nuclear Research.

The Rational Unified Process *http://www.menloinstitute.com/freestuff/whitepapers/Rational%20Unified%20Process.pdf.* Accessed date: August 21, 2003.