PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT

SITI NURHIDAYAH BT MOHAMAD KHAZALI

THESIS SUBMITTED TO IN FULLFILLMENT OF THE DEGREE OF COMPUTER SCIENCE IN SOFTWARE ENGINEERING

FACULTY OF COMPUTER SYSTEMS AND SOFTWARE ENGINEERING

2013/2014

UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS

JUDUL: PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT

SESI PENGAJIAN: 2012 / 2013

SAYA SITI NURHIDAYAH BT MOHAMAD KHAZALI

Mengaku membenarkan tesis/laporan PSM ini disimpan di Perpustakaan Universiti Malaysia Pahang dengan syarat-syarat kegunaan seperti berikut:

1. Tesis/Laporan adalah hakmilik Universiti Malaysia Pahang.
2. Perpustakaan Universiti Malaysia Pahang dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institut pengajian tinggi.
4. **Sila tandakan (√)

☐ SULIT (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) *

☐ TERHAD (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan) *

☐ TIDAK TERHAD

Disahkan Oleh

…………………………….   …………………………………

Alamat tetap: Penyelia:
18300 GUA MUSANG KELANTAN   AZLINA BINTI ZAINUDDIN

Tarikh: 12/12/2013   Tarikh: 12/12/2013

*Sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis/laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged.

Signature: ....

Student: Siti NurHidayah Bt Mohamad Khazali

Date: ....

SUPERVISOR DECLARATION

I hereby declare that I have read this thesis and in my opinion this thesis/report is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)

Signature : …………………………………….

Supervisor Name: Azlina Binti Zainuddin

Date : …………………………………….

ACKNOWLEDGMENTS

Alhamdulillah, thanks to ALLAH because with grant from Him I get finish my PSM project. In this project many experiences I was learnt. Many of people were involved in this project. Without their auxiliary and supported, I think this system will not do with prosperously.

Firstly, I would like to express my sincere appreciation to my project supervisor, Miss Azlina Binti Zainudddin for providing the guidelines with continued advice and feedback, motivation, full support, and constant attentions throughout the duration to finishing this PSM project. Secondly, I would also like to express my appreciations towards my entire friends with their helping, support morale or materials, valuable opinion and sharing ideas during the progress of this project. Their opinions and suggestions have helped realizing this project.

I am very grateful towards to all the lecturers in Faculty of Computer System & Software Engineering (FSKKP), University Malaysia Pahang (UMP) who have directly or indirectly influential and give supportive for this PSM project. Also not forget to any individual that have not mention here but has contributed to this project. To all of them, I only can say very much thank you for their help and support.

Finally, I would like to thank to my family for their understanding, encouragement and support, towards the completion of my project.

ABSTRACT

Final Year Project (FYP) is an integral part of University education, and an important one. It is the time that student work on a significant assignment in term of size and difficulties, and it is also the platform to equip them with practical skills to work on future projects in the industry. Despite for all of its good intentions, the approach for managing students' project is outdated, if not archaic. Key issues in FYP such as tracking project progress and accurate assessment are often difficult to manage. The objective of this paper is to present the process of Progress Monitoring System for Student Final Year Project (PMS). It is based on existing process regarding to solve a problem of monitoring FYP student. User requirement is the best way to implement in the future process. RAD model has been chosen for development PMS. So, the system can helps the online review documentation for student FYP in more efficient ways.

ABSTRAK

Projek Tahun Akhir (PSM) adalah merupakan sebahagian daripada pendidikan Universiti dan merupakan kajian yang penting. Ia merupakan tugasan akhir bagi pelajar dan merupakan platform untuk kemahiran praktikal pada masa akan datang dalam industri. Walaupun pendekatan menguruskan projek pelajar adalah lama, ia tidak kuno. Isu utama yang dibincangkan dalam FYP adalah proses pemantauan pelajar bagi menyiapkan tugasan yang diberikan. Jadi, objektif thesis ini adalah untuk membincangkan Sistem Pemantauan Kemajuan Pelajar Projek Tahun Akhir (PMS). Ia adalah berdasarkan kajian proses yang sedia ada untuk menyelesaikan masalah pemantauan pelajar FYP. Mengetahui keperluan pengguna adalah langkah terbaik untuk diimplement dalam proses yang baru.Model Rad digunakan dalam pembangunan PMS. Jadi, system ini boleh membantu para pelajar menyiapkan thesis mengikut piawai yang ditetapkan oleh pihak FYP dengan lebih cekap.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

1	FYP	Final Year Project
2	FSKKP	Faculty System Computer and software Engineering
3	PHP	Hypertext Pre Processor
4.	PMS	Progress Monitoring System for Student Final Year Project

PART 1

INTRODUCTION

1.0   Introduction

Project monitoring is a significant part of project management. All project activities should be carefully monitored while the project is being accomplished. This helps project manager to keep informed of work progress, and apply timely correctives. It is essential that project monitoring is simple and not time-consuming. Project monitoring software may really help to simplify this process. First, it is necessary to define project goals and plan the activities (Swapna, 2001).

Nowadays, project monitoring will implemented and will use for final year project's student. The purpose of this implemented because to achieve the goal of project undergraduate's subject in university. This way, it helps to track student if they are have problem to finish their project. The step for approach student totally different for each supervisor. So, the best way to keep track student is through web based system. Before this, data and information for FYP is recorded as manually. Staffs need to record all data and information and keep all information in files. Recorded as manually will make the data and information lost. This is not consistent in order to find old record or data. It may cause waste of time when want to search any information.

The Progress Monitoring System for Student Final Year Project (PSM) is computer software which helps student contacts of University Malaysia Pahang (UMP) under Faculty System Computer and Software Engineering (FSKKP) gain better control of their project planning and implementation through keeping student connected with supervisor, regardless of where student are located. Supervisor in FSKKP can monitor every detail, simply, and easily. However the system is about managing projects from remote destinations. So that, this system helps student to complete projects, keep within budget, stay on track, and collaborate with supervisor.

Therefore, the Progress Monitoring system for Student FYP allows its student to easily update project problems as they arise. Those update project problem, actually they are will get the feedback from supervisor. So, Supervisor will evaluate the progress and assign marks to student based on rubric. This system also can helps student streamline the project management process, helping to keep student on track and providing user with reports and real-time data so that their project success is assured.

The software will be used to develop this system is Macromedia Dreamweaver. The language used is PHP. For the database we will use MySQL, Apache and web server XAMP.

1.1   Problem Statement

Nowadays, every third year of the graduate under FSKKP, each third year student has to be assigned a project which she/he has to work on it and submit it till the end of the year. During the year, the student has also to submit various products that show the progress in his/her workings. The report each student has to submit are two (Research Report and Full Report). What the instructor (supervisor) has to do is to assess these products by completing special assessment forms regarding each product.

Based on observation from Andrew Hadiyonto, student has lack experience and insight. They often fail to estimate the time required to complete their task (Andrew Hadiyonto, ISATE 2011). They also failed to understand the whole process in completing the task. They are unable to determine accurately the current status of their project. Besides that, student may not have time to meet their supervisor. So that, many of them are often behind schedule without realizing it and can't achieve the goal of FYP.

Normally, during supervision phase, students have to organize meeting with their supervisor to show their weekly process on the project. Log book is compulsory to record all meeting between student and supervisor . Thus, it is difficult to manage students. All students will miss communication in order to achieve the good product. It is not easy to keep student connected with supervisor. One of the supervisor's tasks in final year project is to track each student's progress. One of the reason students fail to

meet their supervisor because they are busy with their schedule and their time are limited. In addition, supervisor has one or more students to manage and hard to supervisor divided time to meet students. As a result, they cannot do corrections on their works to be submitted and will get lower marks for their project.

Besides that, review process is currently done manually. It is difficult and requires a long time to find files and information stored. All handle in manual activities. On evaluation phase also, evaluator will give marks to students based on their presentation. All marks will recorded in form provided. Once, the form is missing or damage, the mark will be lost. This process requires a lot of man power and there is also no privilege on student's information. After recording the marks, all evaluators are responsible to key in the mark in excel format and email it to PSM/PTA coordinator. So for maximum effectiveness, the review process needs undergo a formal review by using computerize system. So with this system, it will help users as it makes the software project smoother and easier.

1.1.1 Objectives

This project embarks the following objectives:

1. To complete the final year project using systematic approach.

2. Completion with full repository and complete requirement following the timeline.

3. Evaluate the progress and assign marks based on rubric

1.2   Existing System

Nowadays, many supervisors take an interest in the ways in which effective communication systems can facilitate contact between supervisors and their students. The ease of use of electronic mail likes Instant Messaging and Facebook for communication and of resource access via the World Wide Web means that students can complete a research having never left their home environment or having never met their supervisor.

There are some existing systems that use web-based application to manage their system:

1. Managing Student Final Year Projects with Redmine

2. Clarizen's Project Management Software

3. The Design and Implementation of Online Management System for Undergraduates' Thesis (Project)
4. Nanyang Technological University Final Year Project Portal
5. Web-Based Evaluation System for Online Courses and Learning Management Systems
6. Online Document Management system for Academic Institutes

Table 1.1: Comparison Existing System

Existing System	Respondent	Software/ Technique/ Platform	Result
Managing Student Final Year Projects with Redmine	University FYP undergraduate students	Web development, Ms Access or Ms SQL	The system provides all the guidance and improvement for student final year project.
Clarizen's Project Management Software	Team member that involve in project management	Web development, Ms Access or Ms SQL	The system provides solution offers users instant gratification with all aspects of online project progress.
The Design and Implementation of Online Management System for Undergraduates' Thesis (Project)	System administrators, teachers, students and auditors	Web development ASP.NET, Ajax, SQL Server	Improvement of teaching management and the teaching quality
Nanyang Technological University Final Year Project Portal	University FYP undergraduate students	Web development ASP.NET	The system provides all the guidance and details on FYP to

			guide undergraduate students to develop their FYP
Web-Based Evaluation System for Online Courses and Learning Management systems	The approximately 200 students of this course together with four instructors and two administrators	Web development	Implementing a monitoring system of the students' learning behaviour and a consulting system based on the students' results.
Online Document Management System for Academic Institutes	160 students in the Faculty of University of Malaya	PHP5, JSP and MY SQL programming languages	Provide a collection of coordination pathways and interfaces to remove the problems of document access

1.2.1   Research and relationship to current project

1. Managing student Final Year Projects with Redmine

   Redmine has an *update* feature whereby an issue can be "updated" to reflect any problems and findings associating with the specific assigned task. The essential process for it to work is unpretentious. Each student will be given an *issue* (essentially a *task*) Corresponding to their name by either from the supervisor or a teammate, with an estimated date of completion. Once a new *issue* is submitted, all corresponding parties are able to track this task to determine whether it meets the estimated completion deadline or not.

   One of the supervisor's tasks in FYP is to track each student's progress. There have already been some reasonably good systems put in place for this. In the initial part of the project, each FYP team is required to *plan* the entire project duration using *Microsoft Project.* The plan would include each task such as design, development and testing.

   Students are required to create a *Gantt chart* for it. A Gantt chart is a type of bar chart that exemplifies a project schedule. It illustrates the start and finish dates of the terminal elements as well as the summary elements of a project. The intention of the Gantt chart is to help the FYP team to plan their work accordingly.

Figure 1.1: Creating a new issue in Redmine

2. Clarizen's Project Management Software

Clarizen's online project management solution offers users instant gratification with all aspects of online project scheduling – planning, resource load, task updates, scheduling conflicts and milestone progress. This enables project managers to react quickly and easily to all changes in the system without having to wait for team members to "save" or "update" their entries and additions.

Instantly view scheduling dependencies and conflicts – any change made to any project will be instantly updated in the project scheduling view - enabling you to manage these changes and make adjustments as needed
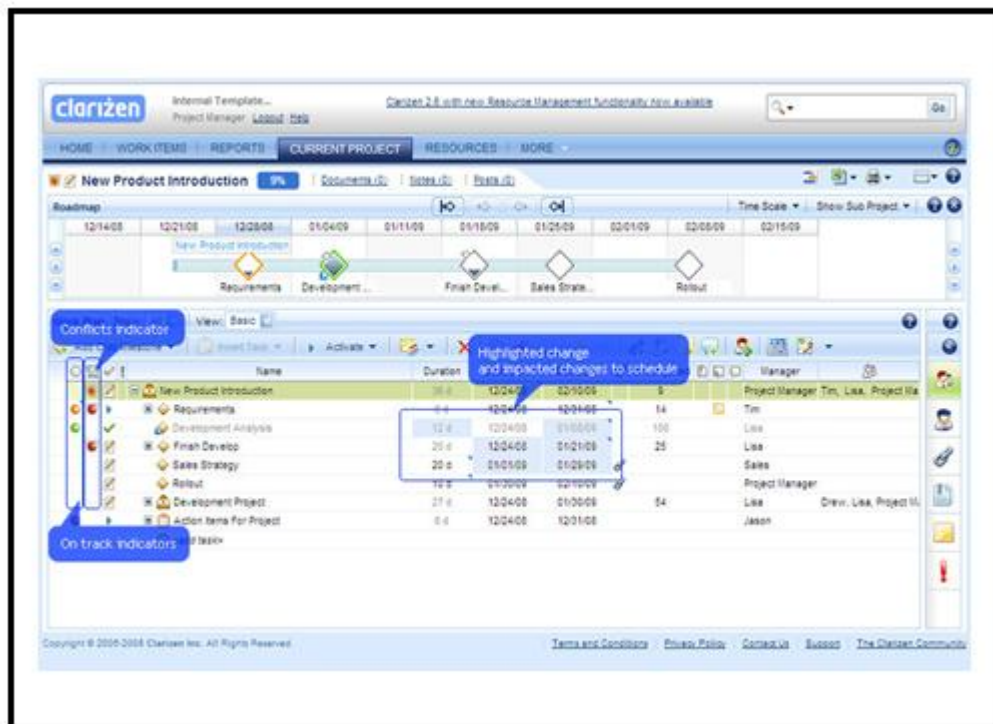

Figure 1.2: Instantly view and manage scheduling conflicts

3. The Design and Implementation of Online Management System for Undergraduates' Thesis (Project)

This system are develop based on online management system for undergraduate's thesis, which is of great practical for improvement of teaching management and quality. The system uses ASP.Net, SQL Server for its development, including four types of users: system administrators, teachers, students and auditors. The paper describes the responsibilities of the four categories of users, workflow, design ideas, and discusses some design methods to enhance the security of the system. The system has been widely promoted in some schools of Huaibei Normal University and achieved good results.

4. Nanyang Technological University Final Year Project Portal

This system provides all the guidance and details on FYP to guide undergraduate students to develop their final year project.

Figure 1.3: Nanyang Tecnological University FYP Portal

5. Web-Based Evaluation for Online Courses and Learning Management System

       This system focus on the Web-based evaluation framework of online courses and learning management system (LMS), based on Web-based questionnaires that are directed at different target groups for the course contents and the design of the LMS as well as the Web site. The evaluation criteria are described in more detail and are included in Web-based questionnaires.

6. Online Document Management System for Academic Institutes

       Provide a collection of coordination pathways and interfaces to remove the problems of document access. This system was develop using PHP, JSP and MYSQL. The respondent in the system require 160 students in the Faculty of University of Malaya.

1.2.2   Comment on existing system

All the systems develop using a web application on platform in order to be accessed by everyone on different places, and it is much easier to apply evaluation, especially when involves a large number of respondents. Most of the system used ASP.NET to develop the GUI of the system. Thus, PHP is the better development programming language as it open source and can be implemented on all platform. From the previous system, it is much more focus on providing guidelines and final submission. Based on my observation, monitoring through online communication must implement in the system. It is can help the FYP process more effective and efficiency.

When comparing PMS with others system, functionality of the system should be consider. The first function in the system are generate report and update project problems. This function is quite important because if the system not provide this function, it can cause problem and the system will become complicated. Not all the system provides the function like PMS. PMS allows student get the feedback from

supervisor. Thus, PMS is an automated solution for FYP student problem. The online progress log feature is provided for students to keep updating the progress. This progress is dates and timed. The supervisor can also put feedback or comments on the progress. This can also be used for online discussion on aspects of the project.

1.3   Current System

Currently, process throughout the undergraduate project is done by manually. PSM coordinator used this current manual process to manage the subject activities such as review report, marks calculation, etc. The following flowchart will describe the process in completing the undergraduate project.
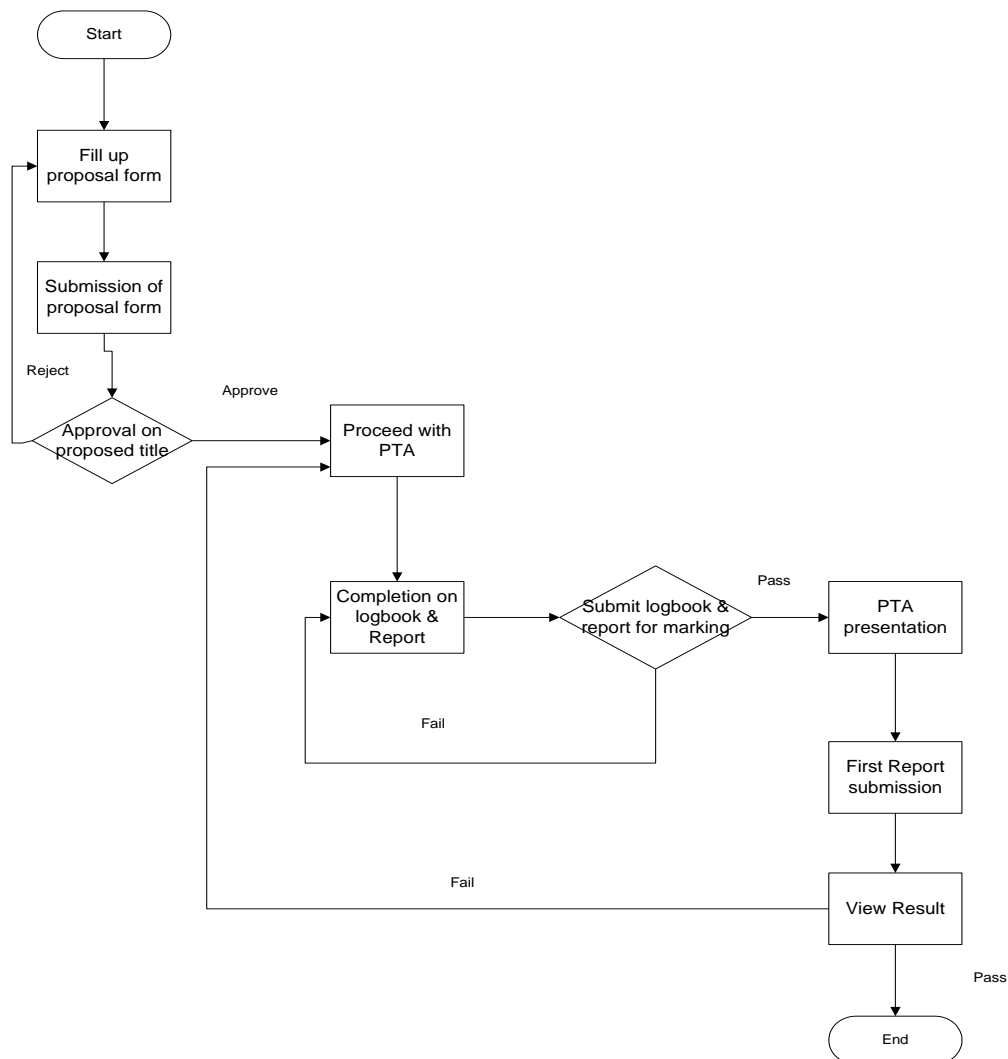

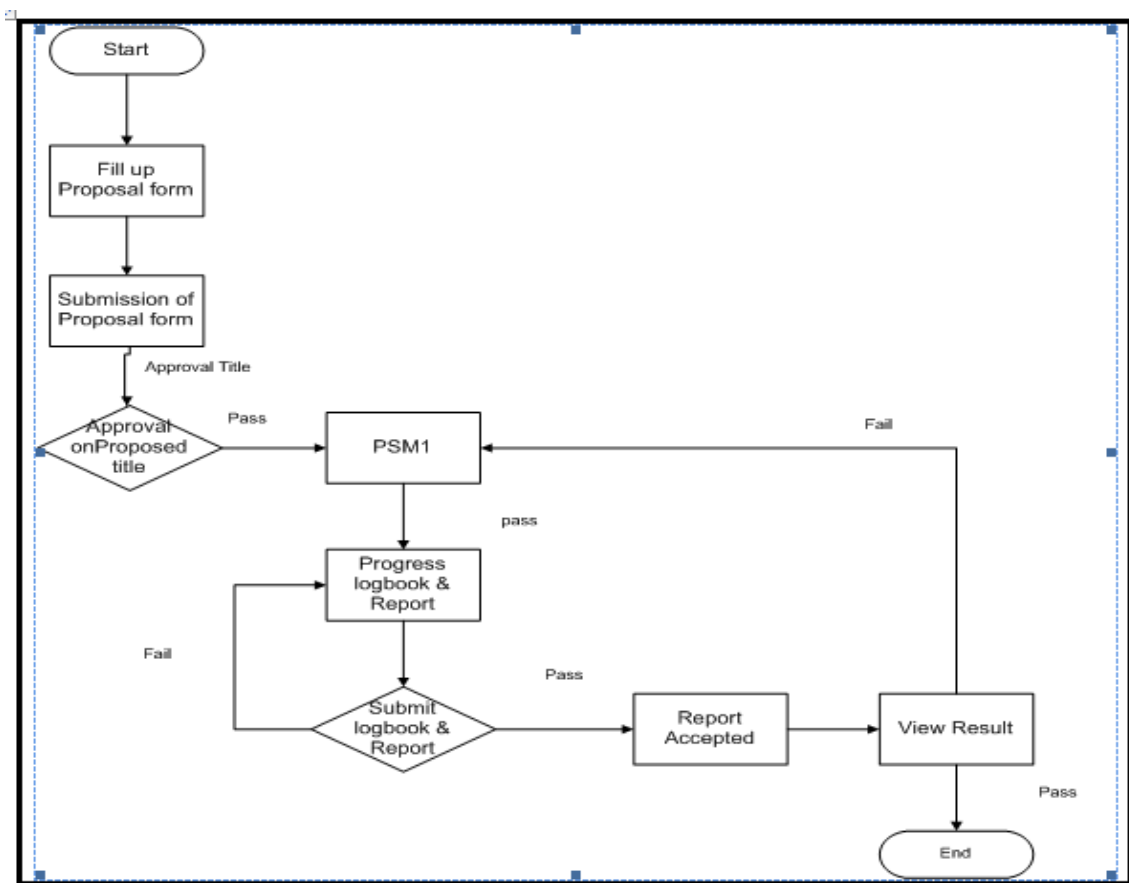Figure 1.4 : PTA current manual process flow chart

Figure 1.5 : PSM1 current manual process flow chart
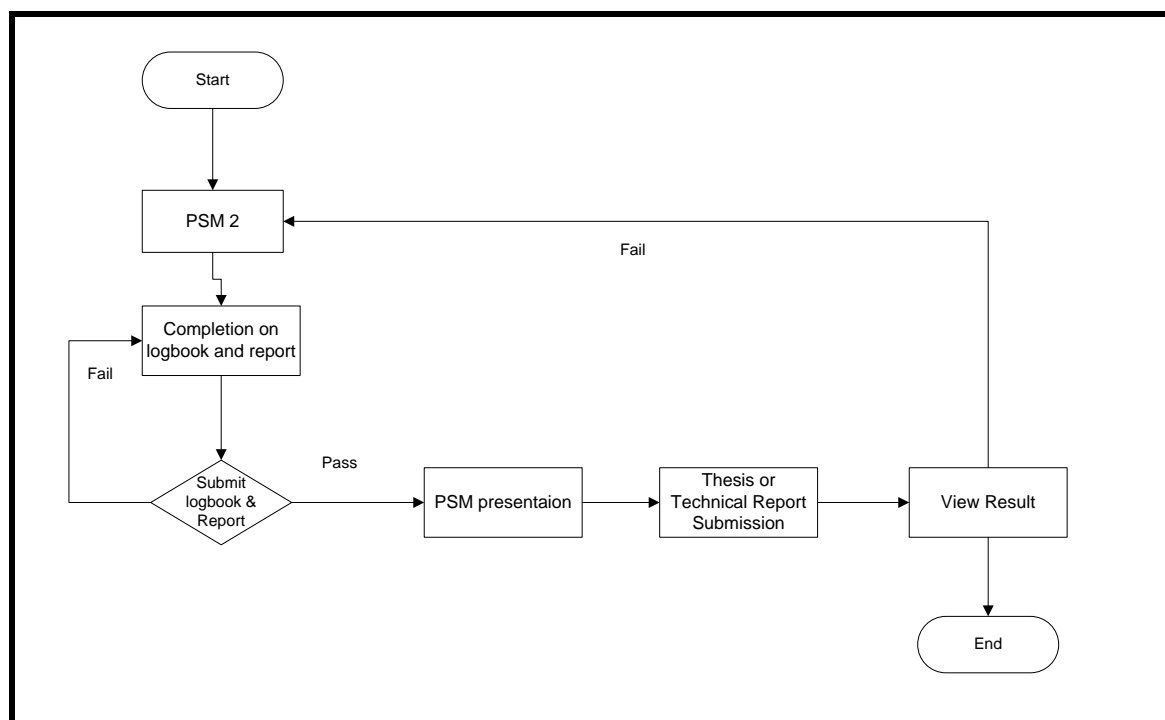
Figure 1.6 : PSM2 current manual process flow chart

1.3.1 Limitation

1. Weekly Evaluation by Supervisor

2. Process for evaluate student too complicated.

3. The report of the FYP can be lost and damage.

4. Miss communication between student and supervisor.

1.3.2 Solution

By using PMS, students can update their logbook at any time via a paperless, environmentally-friendly method as well as submit their logbook and final report through online. Supervisors can access the student's logbook at any time, therefore they can evaluate and grade the student at their own page. Student can submit their report and get feedback from their supervisor. Supervisor will assign marks to students on their progress and performance during presentation. After that, student able to check their result. The result will generate automatically.

1.4 Terminology

Table 1.2: List of Terminology

CSV	comma separated value
DML	Data Manipulate Languge
FSKKP	'Fakulti Sistem Komputer dan Kejuruteraan Perisian' or Faculty of Computer System and Software Engineering
PMS	Progress Monitoring System
FYP	Final Year Project
PSM	"Projek Sarjana Muda"
PTA	"Projek Tahun Akhir"
RAD	Rapid Application Development
SQL	Structured Query Language
UMP	Universiti Malaysia Pahang

1.4   Method of Approach

Model	Sources of method	Stages/phase	Scenario
Systems development life-cycle(SDLC)	(Wikipedia, 2013)	5 phase consist of requirement analysis, design, implementation, testing, evaluation	When want to build a high quality system that meets or exceeds customer expectations.
	(Wiras, 2008)	4 phase consist of Planning, Analysis, Design, Implementation	If the system is not big and complex and have a fixed requirement.
	(AbdouIllia, 2013)	4 stages consist of System Planning and Selection, System Analysis, Systems Design, Systems Implementation and Operation	If the system is not big and complex and have a fixed requirement.
RAD(Rapid Application Development)	(Konstantinou, 2013)	4 stages that consist of requirement planning, user design, construction, implementation	When want to build a quality system in a fast time.
	(Martin , 2013)	4 stages that consist of requirement planning, user design, construction, implementation	When want to build a fast, efficient, accurate program and/or system development and

			delivery
	(Cinoy M.R , 2006)	5 stages that consist of business modeling, data modeling, process modeling, application generation, testing and turn over.	When want to create a fully functional system with in very short time period.
RUP(Rational Unified Process)	(Wikipedia, 2013)	4 stages consist of inception, elaboration, construction, transition.	Good for a complex and risky project or system.
	(IBM, 2001)	5 stages consist of inception, elaboration, construction, transition, iteration.	Suitable for a wide range of projects and organizations.
	(Hessel, 2002)	4 stages consist of inception, elaboration, construction, transition.	
Object Oriented Methodology	(Aviator, 2013)	3 stages consist of analysis phase, design phase, implementation phase.	When a project or system that has the reuse module or component
	(OGCIO, 2013)	Has 6 stages consist of business planning stage, the business architecture definition stage, the technical architecture definition stage, the incremental	When there is another project want to re-use the existing components and facilitates of the systems.

		delivery planning stage, the incremental design and build stage, and the deployment stage	

Table 1.3: Comparison between Methodologies

The disadvantage of the SDLC is the testing process is occurs at the end of the development process so when there are requirements changes from the user, the developer need to changes the overall of the system, this make the time and cost for develop the system increases.

The disadvantage of the RAD is not for large projects. Others disadvantage is if commitment is lacking from either constituency, RAD projects will fail and it also not appropriate when technical risk are high.

For RUP, the disadvantages are the team members need to be expert in their field to develop software under this methodology, if not the system is possibly to fail. The development process is too complex and disorganized so it can make the team get confuses and difficult to flow the model.

The method will be approach for construct PMS is Rapid application development (RAD).Following figure will show the sample of RAD Model.
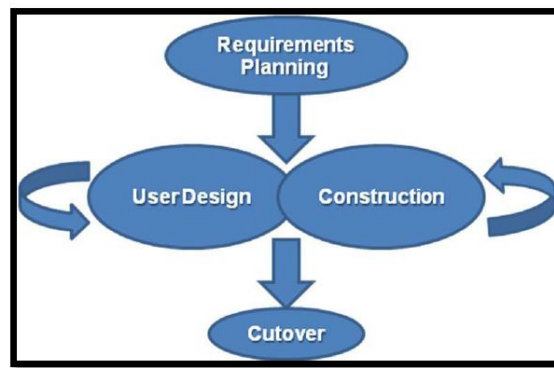
Figure 1.7: Rapid Application Development (RAD Model)

The first phase is requirement planning, combine elements system planning and system analysis phase of Software Development Life Cycle (SDLC). In order to arrange requirement to develop Progress Monitoring System, research on existing system is done in order to gain better understanding on the current situation. Information is gathered during the research gives clearer overview on the flow of the process while answering the question on how to achieve the main goal of the system. The scopes of project are specified and a schedule has been design as guidance throughout the system development process to make sure the delivery of it on timely mannered. This has been done with the help of Gantt chart produced through Microsoft Project. The capture of user requirements is the process of gathering information about user needs. So, user requirements should be clarified through criticism and experience of existing software and prototypes.

During user design phase, the requirements are describe in detail while the requirements discovered during requirements planning stage are developed into a data model such as formalize business rules and creates screen flows also layouts for important parts of the system. From the goal identified in the previous phase of requirement planning as well as research done on existing system, the data technique needed for develop PMS are identified. In addition, users interact with systems analysts and develop models and prototypes that represent all system processes, inputs, and outputs. User Design is a continuous interactive process that allows users to understand, modify, and eventually approve a working model of the system that meets their needs.

In this construction phase, the application is programmed. The most important thing to consider at this point is that just because the application is being constructed does not mean the design process has concluded. When users get something in their hands that actually functions, they will inevitably change their minds as to how they want things to work. The codes in developing system will be implemented by using Macromedia Dreamweaver. It is the technique that will convert by describing the user design phase into executable program.

Cutover phase is the phase which the testing and validation of the system will be implemented. So, the cutover phase are also will implement in PSM II. In this stage, the new prototype is delivered to the users which include final user testing and training to check application system. Trouble shooting is done after the deployment and potential enhancements are identified and tracked. However, for PMS development, it will only involve until testing stage.

1.6   Indication of Scope

1. **Student**

   - Submit project progress by chapter and full report.

   - Submit logbook activities and generate log book.

2. **Supervisor**

   - Evaluate progress and assign mark based on rubric.

   - Give feedback when student submit project progress.

   - Set the time and arrange it for meet their student.

   - Approve or rejects activities submitted by students.

3. **Coordinator**

- Generate all student record into excel in .xls format.

- Upload the format of technical report, thesis, rubric and others.

- Insert all data student into database.

- Assign two evaluator for each single students.

- Able to add, delete and search student.

4. **Software**

- PHP LAnguages

- Apache

- MySQL Database

- Adobe Dreamweaver 8

5. **Hardware**

- Laptop

1.7   Outline of Material

The overall of this report consists of three (3) main parts. Part 1 will discuss on the purpose behind the project, existing system that related to the proposed system.

Part 2 will discuss on user requirement, design description, development plan and testing plan in the system.

Finally, it will discuss in the conclusion obtain in the overall process through the development throughout the system.

PART II

REPORT BODY

2.1   User Requirement

All user requirements located at Appendix   A (Software Requirement Specification).

2.2   Design Description

All design description of the application is documented in the Software Design Documentation (SDD)-Appendix B.

2.2.1   Methods and Material

1.   Method

In development of system, the important thing must be pay attention when choose the right methodology that suite within this project. It is ensure the phase of development running smoothly and achieve the goal of final year project. There are many type of methodology. The method will be approach for construct PMS is Rapid application development (RAD).
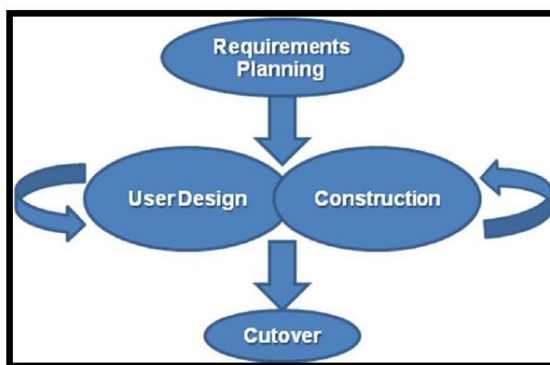
Figure 2.1: Rapid Application Development (RAD Model)

In this phase, user requirement are collect and define what the first priority requirement. In order to arrange requirement to develop Progress Monitoring System, research on existing system is done in order to gain better understanding on the current situation.

At this stage, the user requirement about PSM/PTA management is collected. The planning process included the pelan schedule and pelan what the task should be complete first in order. Thus, the task should be prioritized. For this system, the whole system is using process or flow from PSM/PTA management. At planning stage, the security requirement also is define in order to make the system more secure from being hack by attacker. PMS use validation and hard code for introduced security into more tight.

During user design phase, the requirements are describe in detail while the requirements discovered during requirements planning stage are developed into a data model. From the goal identified in the previous phase of requirement planning as well as research done on existing system, the data technique needed for develop PMS are identified. In addition, users interact with systems analysts and develop models and prototypes that represent all system processes, inputs, and outputs. User Design is a continuous interactive process that allows users to understand, modify, and eventually approve a working model of the system that meets their needs.

In this construction phase, the application is programmed. The most important thing to consider at this point is that just because the application is being constructed does not mean the design process has concluded. When users get something in their hands that actually functions, they will inevitably change their

minds as to how they want things to work. The codes in developing system will be implemented by using Macromedia Dreamweaver. It is the technique that will convert by describing the user design phase into executable program.

Cutover phase is the phase which the testing and validation of the system will be implemented. In this stage, the new prototype is delivered to the users which include final user testing and training to check application system. Trouble shooting is done after the deployment and potential
enhancements are identified and tracked. However, for PMS development, it will only involve until testing stage.

2.   Materials

The hardware and software that will be used for this project are :

Hardware	Function

Laptop
 	
   i.   Hp Pavilion g Series
   ii.   4.00 GB RAM
  iii.   Intel Inside Core i3
  iv.   64-bit Operating System Win 7
	Used for development, testing and maintenance the project until the end.

External Hard Disk : Adata 500GB	Backup data and files

Table 2.1 List of Hardware used

Software	Function
Apache	Web Server
Adobe Dreamweaver 8	Development language
MySQL Database	Database application software
PhpMyAdmin	Server management

Table 2.2 List of Software Used

2.3   Development Plan

As mention earlier, RAD will be used for the development phase of this system. Each module have being defined. While, each type of function will be tested to ensure all the function its work properly. In order to complete the whole project, the each function of this system must have different roles.

1.   Coordinator

Coordinator is able to register students into database; the file should be in excel format which the extension .xls. Coordinator is able to assign two evaluators to each student. Furthermore, coordinator will able to generate student records into excel and able to upload the file such as format of technical report, thesis, and etc for student and supervisor.

2.   Supervisor

Supervisor is able to review the student progress and evaluate it. If have any problem and misunderstanding supervisor shall give the feedback to their student. Supervisor will monitoring their student and manage the time to meet them. Normally, supervisor will approve activity that submit by their students and assign mark to them based on the rubric stated.
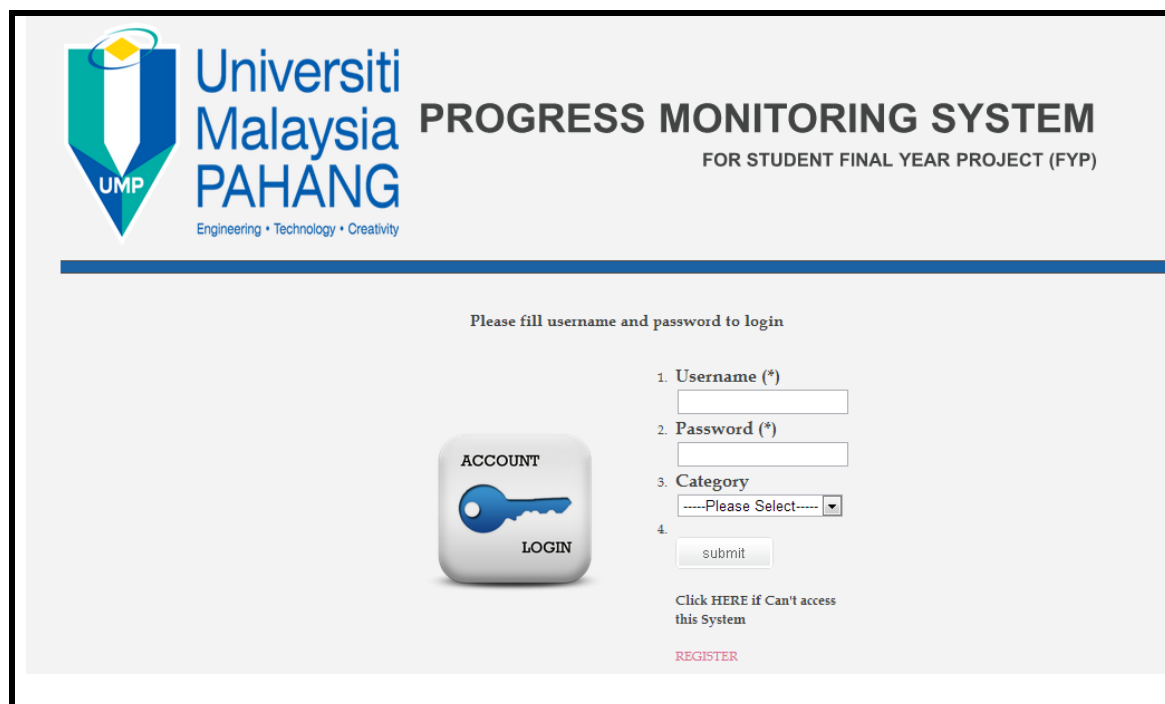
3. Student

Student shall be capable to update their project progress by weekly depends on dateline. Student able to get feedback from their supervisor when they are update the progress as they arise it. Student also able to download format of technical report, thesis and etc for guidelines. However, student able to check evaluator before they are present their project. Student should send their weekly report and upload full report in the PMS.

2.4 Implementation

The main purpose of this section is to document all the process and steps that involves in developing the systems. The Process of developing PMS started with developing the database, then interface, and finally coding the application logic with connects the database with the interface. The database is use for saving the information of student's PSM/PTA management while the user interface allows the user to communicate with the system. The application logic set the behavior of the system towards user interactions with it. PMS was develop using SQL, PHP and HTML.

2.4.1 Development Interface

Interface development is very important for web-based applications. The interface can attract and influence a user's experience of using the system. The interface should be easy to use while presenting a logical flow.

Figure 2.2 : Index page for PMS

2.4.2   Student

2.4.2.1 View Status and schedule

Student can access their own information.

Figure 2.3 : Status for PMS

Student shall to view the schedule that arranged by their supervisor.

Figure 2.4 : Schedule for PMS

2.4.2.2 View Evaluator

Student can view their evaluator before start to present.
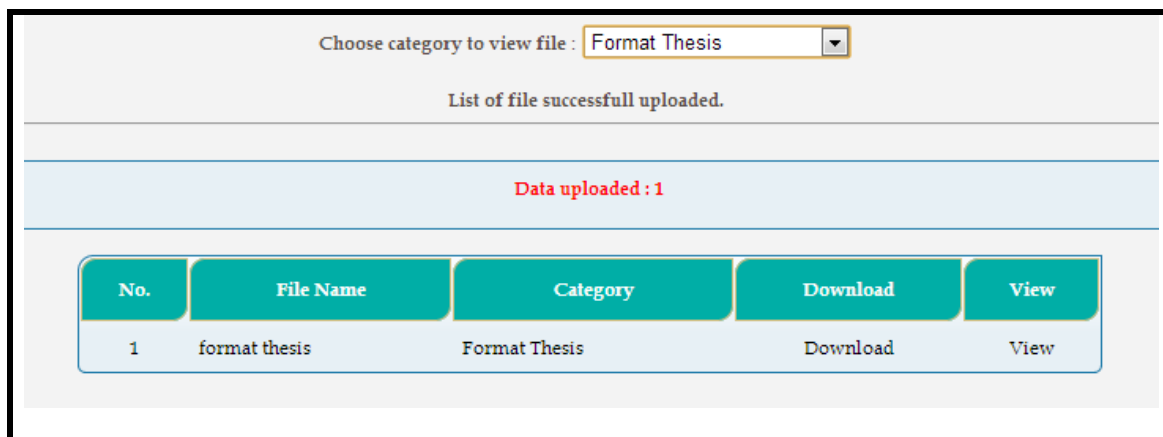

Figure 2.5 : View Evaluator for PMS

2.4.2.3 View Result

Student can view their result after result was released by supervisor. Student is able to see 20% marks only.

Figure 2.6 : View Result for PMS

2.4.2.4 Download File

Student is able to download the file that are uploaded by coordinator.

Figure 2.7 : Download File for PMS

2.4.3   Supervisor

2.4.3.1 Assign Mark

Supervisor should evaluate student based on rubric. Once student completed submit their project progress, student able to view their result. So, the third objective that evaluate the progress and assign marks based on rubric was achieved.

Matric ID	Name	Supervisor Marks(20%)	Supervisor Marks(40%)	Total Marks
CB10072	SUKIMAN MAT JUSOH	0	0	0
CB10083	MOHAMMAD REDZUAN BIN MOHD YUSOF	0	0	0

Matric ID being observed : -choose one--

ROGRESS REPORT SUPERVISOR (20%)

CO	ITEM	WEIGHT %	SCORES (0-5)	MARK PERCENTAGE
CO1	Introduction/ Overview	0.4	---	
CO1	Problem Statements	0.6	---	
CO1	Objective	0.6	---	
CO1	Scope	0.6	---	

Figure 2.8 : Assign Marks for PMS

2.4.3.2 Approve Student

This event shall starts when actors request new title or proposed new title. Supervisor may approve and cancel the request.

HOME	APPROVE	EVALUATE	MANAGE	PROGRESS	DOWNLOAD

You are log in as AzlinaBintiZainuddin (logout)

MATRIC ID	NAME	TITLE	VIEW
CB10072	SUKIMAN MAT JUSOH	Portal Persatuan Sejarah Malaysia Cawangan Pahang	View
CB10083	MOHAMMAD REDZUAN BIN MOHD YUSOF	Health Management Online	View

Figure 2.9 : Approve Student for PMS

2.4.3.3 Set Schedule

Supervisor manage time to their student. Its mean, supervisor set schedule when him/her want to meet face to face their student.

Figure 2.10 : Set Schedule for PMS

2.4.3.4 Download Page

Supervisor able to download the uploaded files by coordinator.

Figure 2.11 : Download Page for PMS

2.4.4 Coordinator Management

2.4.4.1 Insert Student

Coordinator insert all data student into database. The file should be in excel format which the extension .xls.

Figure 2.12 : Insert Student for PMS

2.4.4.2 Add Supervisor

Add new supervisor into database.

Figure 2.13 : Add Supervisor for PMS

2.4.4.3 Assign Evaluator

Coordinator must assign two evaluator for each single student. Student able to check evaluator before they are present their project.

No.	Matric ID	Name	Category	Title	Supervisor	Evaluator 1	Evaluator 2	Start time	End Time	Location	Edit
1	CB10072	SUKIMAN MAT JUSOH	P5M1	Portal Persatuan Sejarah Malaysia Cawangan Pahang	AZLINA BINTI ZAINUDDIN						
2	CB10083	MOHAMMAD REDZUAN BIN MOHD YUSOF	P5M1	Health Management Online	AZLINA BINTI ZAINUDDIN						

Page Number: 1

Total Records: 2

Searched For: P5M1

Figure 2.14 :Assign Evaluator for PMS

2.4.4.4 Timeline

Coordinator should add new timeline as a remainder to student and supervisor.

Figure 2.15 :Timeline for PMS

2.4.4.5 Upload

Coordinator should upload the format of technical report, thesis, rubric and others. Student and supervisor able to download the uploaded files by coordinator.

Figure 2.16 :Upload for PMS

2.4.4.6 Report

Coordinator will able to generate all student records into excel in .xls format.

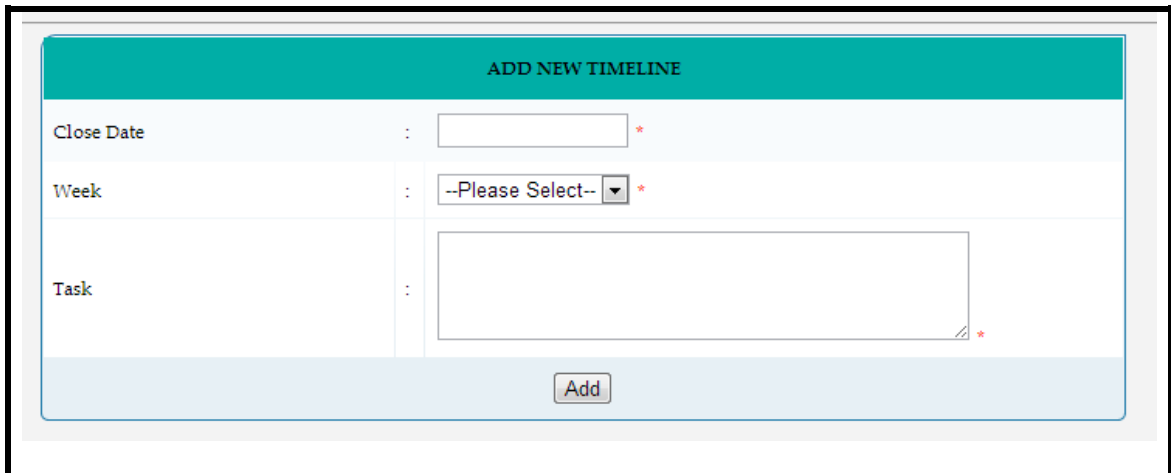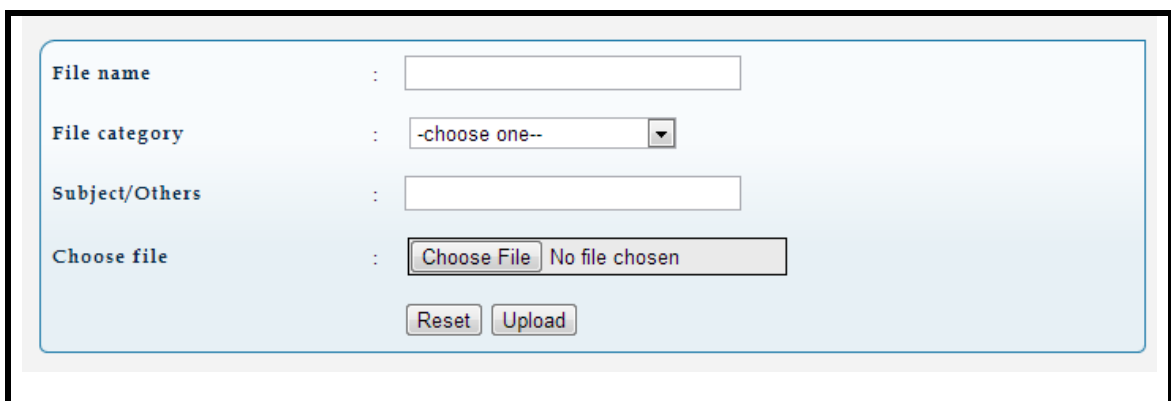| matric_id	name	category	title	sv_name	sv_mark1	sv_mark2	ev1_name	ev1_mark
CB10011	YEOW JIAN QIN	PSM1	Student Location Prediction Using Naïve Bayes Algorithm	MAZLINA BINTI ABDUL MAJID	0	0		0
CB10013	EDMUND WANG HUI CHEE	PSM1	Realtime Data Center Dashboard System By Applying Data Visualization	MAZLINA BINTI ABDUL MAJID	0	0		0
CB10041	MUHD. FAHMI	PSM1	Temperature Based Cooling System	MANSOOR ALAM	0	0		0
CB10048	TAUFEQ ABD AZIZ	PSM1		LUHUR BAYUAJI	0	0		0
CB10059	MUHAMAD NOR ZAKWAN BIN CHE ALI	PSM1	Event Hall & Facility Booking With Intelligent Floor Plan	ROZLINA BINTI MOHAMED	0	0		0
CB10062	AHMAD MUZAIDI AB RAZAB	PSM1	Hatching Tgechnique On Photi ID	MOHAMAD FADLI BIN ZOLKIPLI	0	0		0

Figure 2.17 :Report for PMS

2.5 Database Design

Database Design is a process of developing a database design or data model that will meet user requirements. Database design will show at physical database in the Table. In this system, there are several tables used databases which are:

Table	Action	Rows	Type	Collation	Size	Overhead
coordinator	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2.1 KiB	20B
feedback	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	–
file_upload	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	1.5 MiB	–
forum	Browse Structure Search Insert Empty Drop	~8	InnoDB	latin1_swedish_ci	16 KiB	–
logbook	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	–
marks	Browse Structure Search Insert Empty Drop	~81	InnoDB	latin1_swedish_ci	64 KiB	–
progress	Browse Structure Search Insert Empty Drop	~2	InnoDB	latin1_swedish_ci	48 KiB	–
report	Browse Structure Search Insert Empty Drop	~3	InnoDB	latin1_swedish_ci	2.5 MiB	–
student_psm	Browse Structure Search Insert Empty Drop	~105	InnoDB	latin1_swedish_ci	32 KiB	–
supervisor	Browse Structure Search Insert Empty Drop	66	MyISAM	latin1_swedish_ci	11.1 KiB	–
sv_session	Browse Structure Search Insert Empty Drop	~2	InnoDB	latin1_swedish_ci	16 KiB	–
task	Browse Structure Search Insert Empty Drop	~0	InnoDB	latin1_swedish_ci	16 KiB	–
timeline	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	–
13 tables	Sum	272	InnoDB	latin1_swedish_ci	4.3 MiB	20 B

Figure 2.18 :Database Design for PMS

4.6 Conclusion

The implementation phase of PMS is done in accordance to the SRS, SDD, and STR of PMS. The development of PMS was separated into user interface, application logic and database implementation parts.

PART III

CONCLUSION AND FUTURE WORKS

3.1   Conclusion

As conclusion, PMS is developed to ease the process of PSM/PTA management. The objective has achieved which a web- based system that contains final year project student's information, which based on registration, management and evaluation. This system also embeds smart application where the system can generate weekly activities done by students and can accept excel in .xls format. There are few enhancements that can be done to produce a better system. Requirement gathering it is very important to ensure a good system created and fulfilled. Moreover, user requirement is the best way to implement in the future process. To sum up, questionnaires survey form designed to verify the existing processes which related with the project progress monitoring. A survey was conducted among the FYP students to see how the perceived their final year project and how they felt that their projects successful. ass users' requirement.

3.2   Results

As a result, this project finally achieved their objective and goals. Overall, the achievement objectives as follow:

- To complete the final year project using systematic approach.

- Completion with full repository and complete requirement following the timeline.

- Evaluate the progress and assign marks based on rubric.

3.3   Limitations and advantages of the findings

3.3.1   Limitations

For the supervisor side, it is hard to keep track of their supervise students where they are cannot get notified on the problem students, and they have to approve each student's activities in order.

3.3.2   Advantages

This project fulfilled the objectives where the system successfully developed a prototype for PMS; which embedded the smart applications and successfully tested its functionality.

3.4   Judgment / Evaluation

PMS successfully eases the difficult process of PSM/PTA management. PMS provides improvement in evaluation procedure and monitoring process. It is reduce the workload of coordinator and supervisor, and student management.

3.5   Suggestion and Further Enhancement

There are several enhancements that can be carried out for the future improvement of PMS to ensure that the development throughout the system is more reliable and dependable for prospective management activity.

- The scope of the system can extends or with combinations of other sub modules so that a complete web-based system can be developed to cover all the activities of PSM/PTA management.

- Recommended and encourages increasing the security of the system such as using Oracle database.

REFERENCES

[1]   Adam Hobach. 2008. Advantages and Disadvantages to Using MySQL Vs MS SQL. http://ezinearticles.com/?Advantages-and-Disadvantages-to-Using-MySQL-Vs-MS-SQL&id=1559158 (19 October 2012).

[2]   Kerr, James Richard ,1993. *Inside RAD: How to Build a Fully-Functional System in 90 Days or Less*. McGraw-Hill.

[3]   Andrew Hadinyoto, 27 – 29 September 2011 , Managing Student Final YearProjects With Redmine, International Symposium on Advances in Technology Education : (265-271).

[4]   Al-harthy, A.S.H. *The Consultancy Fee for Structural Design Changes of Reinforced Concrete Building in Oman*. Faculty of Civil Engineering, Universiti Teknologi Malaysia, Skudai, Johor, Malaysia (2006): Ph.D. Thesis.

[5]   H.L. Gantt, *Work, Wages and Profit*, published by *The Engineering Magazine*, New York, 1910.

[6]   Boyd, 2011, Software Engineering Reviews and audits, Taylor Group, LLC Auerbach Publications is an imprint of Taylor & Francis Group, an Informa Bussiness.

[7]   Wikipedia. 2012. Web content management system. http://en.wikipedia.org/wiki/Web_content_management_system (13 September 2012)

[8]   C. Y. David and S. D. William,1999. "Rapid   application   development (RAD)",CRC Press   LLC.

[9]   Boyd Lawrence, 2011. *Software   Engineering Reviews and audits*, Taylor Group, LLC Auerbach Publications is an   imprint of Taylor &   Francis Group, an   Informa Bussiness.

[10]   Michael Chong, W.2012. PSM/PTA Evaluation System.(7 June 2012)

[11]   Hongfu, W. 2011. The Design and   Implementation of Online Management System for Undergraduates' Thesis (Project). (15 October 2012)

[12]   Boyd Lawrence, 2011. *Software   Engineering Reviews and audits*, Taylor
Group, LLC Auerbach Publications is an   imprint of Taylor & Francis Group,
an Informa Bussiness.

[13]   C. Subhas., Mukhopadhyay, A. Gaddam., S.Gupta., 2008. Wireless sensor for
home monitoring. *Recent Patents on Electrical Engineering*. 1: 32-39.

[14]   Linkedin. Advantages & Disadvantages of Making Website in ASP.NET.
http://www.linkedin.com/company/mindpower-weblabs/advantages-
disadvantages-of-making-website-in-asp-net-537455/product (18 October 2012)

[15]   Qian Heng, Y. 2011. System Development of FYP Portal (Registratiom Module).
(18 October 2012)

[16]   Vigneshpalani's Blog. 2010. Advantages & Disadvantages of Mysql.

http://vigneshpalani.wordpress.com/2010/05/10/advantages-disadvantages-of-
mysql/ (19 October 2012)

[17]   Wikipedia. 2012. Web content management system.

http://en.wikipedia.org/wiki/Web_content_management_system

(13 September 2012)

[18]   Haryani, I. Tuition Centre Management System. 9.
http://library.utem.edu.my/index2.php?option=com_docman&task=doc_view&gid=
3050&Itemid=113 (23 September 2012)

[19]   Chakkrit, C., Bruckner, M. 2008. Second IEEE International Conference on Digital
Ecosystems and Technologies. *Web-Based Evaluation System for Online Courses
and Learning Management System.* 332-338.

[20]   British Standards Institution. 2012. What are management systems?
http://www.bsigroup.com/en/Assessment-and-certification-services/management-
systems/at-a-glance/What-are-Management-Systems/ (23 September 2012)

[21]   Martin   Fowler.   2005.   The   New   Methodology.
       http://www.martinfowler.com/articles/newMethodology.html
       (17 November 2013)

[22]   Paul Mah. 2009. Disadvantages of Web Development Using PHP.
       http://www.itbusinessedge.com/cm/blogs/mah/disadvantages-of-web-
       development-using-php/?cs=33397/ (17 November 2013)

APPENDIX A

SRS

2013

SOFTWARE REQUIREMENT SPECIFICATION (SRS)

PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT (FYP)

SITI NURHIDAYAH BT MOHAMAD KHAZALI

CB11008

900821035942

To be submitted to the Undergraduate Project

Bachelor of Computer Science (Software Engineering)

DOCUMENT APPROVAL

	Name	Date
Verified by: _____ Project Manager	Siti NurHidayah Bt Mohamad Khazali	
Authenticated by: _____ Supervisor	Miss Azlina Binti Zainuddin	
Authenticated by: _____ Client	Coordinator Undergraduate Project	

Software   : IBM Rational Software Architecture (RSA), Microsoft Office 2007

Archiving Place   : D:\PMS\DOCUMENTATION\SRS\

Copies Available   : doc, docx,pdf

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDIXES

1. INTRODUCTION

1.1 PURPOSE

One of the important phases of software development life cycle is testing phase. Software Testing is a vital area that is being considered and important in the world of fast changing technology. The more a product is tested, the better the quality would be. Normally, software test result reporting would improve the system about 5 to 10%.

1.2 SYSTEM IDENTIFICATION

Progress Monitoring System for Student Final Year Project (PMS)

1.3 SYSTEM OVERVIEW

The Progress Monitoring System for Student Final Year Project (PSM) is computer software which helps student contacts of University Malaysia Pahang (UMP) under Faculty System Computer and Software Engineering (FSKKP) gain better control of their project planning and implementation through keeping student connected with supervisor, regardless of where student are located. Supervisor in FSKKP can monitor every detail, simply, and easily. However the system is about managing projects from remote destinations. So that, this system helps student to complete projects, keep within budget, stay on track, and collaborate with supervisor.

1.4 REFERENCES

This section contains the references used in producing this document.

i. IEEE.*IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

ii. http://share.pdfonline.com/2a295f5c17d24edfb35ba48c43b9a2b5/cmp.htm

iii. Addison Wesley, "Software Engineering", Eighth Edition, Ian Soomerville, 2007

iv. Description about how to write Software Requirement Specification (SRS)

1.5 DOCUMENT OVERVIEW

This document follows the outline based on the IEEE Standard 830-1998 for Software Requirement Specification (SRS). The remainder of this document is four chapter :

Chapter 1   **Introduction**

This chapter describes the purpose of this document. The reason behind the conception of PMS is also stated in this chapter included problem statement

Chapter 2   **Overall Description**

This chapter describes an overall description of the systems as well as the system's functions and the characteristics of its users. This chapter also specifies the constraints, assumptions and dependencies of the system.

Chapter 3   **Specific Requirement**

In this chapter, the requirement of the system will be defined clearly and in detail. Each module is described and accompanied by its sequence diagram. The external interface requirements, software product features, performance requirements and requirements traceability are also described.

Chapter 4   **Definition and Acronyms**

In this chapter, the definitions and acronyms are listed. This chapter serves as a reference for users to look up should any confusion regarding the acronyms used arise.

2.   OVERALL DESCRIPTION

2.1   Product Perspective

Progress Monitoring System for Student Final Year Project will develop using web based application, Adobe Dreamweaver 8 and PHP scripting language, and interact with MySQL Server.

1. The web pages (XHTML/PHP) are present to provide the user interface on the client side.

2. The Client Software is to provide the user interface of system user on client side, and for this TCP/IP protocols are used.

3. Communication between client and server is provided through HTTP/HTTPS protocols.

4. On the server-side, web server is for PHP and database server is for storing the information.

2.1.2   System Interfaces

The system interfaces are below:
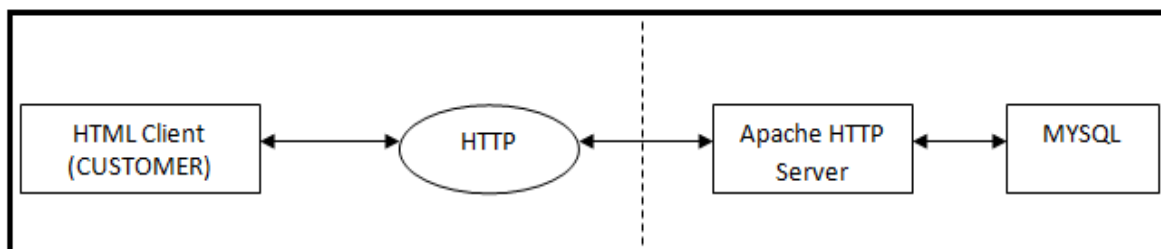
Figure 2.1: System Interfaces

2.1.3 User Interfaces

There are three different category of users who will use the system; coordinator, student and lecturer. All users will access the system via the web browser. The application should allow the basic process such as insert, update, delete, and view for all the users.

2.1.4 Software Interfaces

This subsection will be explained in detail in subsection 3.1.3

2.2 Product Functions

The PMS can be accessed by students, supervisor, and coordinator through the website. In this system, coordinator who is administrator have full authority in manage database. Coordinator will generate report from PMS. The lecturers play two roles within the system, which is as supervisor and evaluator. As a supervisor, they are can assign marks to their supervise students progress. As an evaluator, they are will assign marks to students on their performance during presentation. Students will have to login in the PMS for submit their progress, generate log book and view their status.

2.3 User Characteristics

There are three different categories of users in PMS, which are coordinator as administrator, supervisor, and student. All users are assumed to have basic knowledge of computers and internet browsing. Coordinator should know the flow and process within the system in order to assist other users when they face problems in using the application. So, the target user must have the following characteristics to successfully use PMS.

i. **Basic Computer Skills**

User must at least know how to use a keyboard and mouse as interact with the GUI of the system. Knowledge is using internet browsers is also required.

ii. **Prior Experience**

No experience is needed. The interface will be easy to use as well as provides guidelines to the user.

2.4 Constraints

1. System is limited to HTTP/HTTPS protocols as the system is a web-based application.

2. The used of hardware and software by user should fulfill the minimum requirement to the system.

3. Server must be always available.

2.6   Assumptions and Dependencies

1.   The system will be able to access by major of internet browser such as Google Chrome and Mozilla Firefox. The best solution is Google Chrome that supported graphical user interfaces.

2.   The speed of accessing the system depends on the network speed.

3.   Higher RAM provides higher performance of the system.

4.   Roles and responsibilities are already established.

3. SPECIFIC REQUIREMENTS

3.1 External Interface Requirements

3.1.1 User Interfaces

User interfaces is the part of the system that interacts with user. The user interface is vital to determining the application usability; therefore user interface must be easy to use while meeting the user requirements. There will be users who are inexperienced and have limited knowledge in using a system, thus a user a friendly system is especially crucial.

There will be different set of user interface, according to the user's role and responsibilities. The user of this system are divided into 3 types which are coordinator, student and supervisor.

3.1.1.1 Coordinator Interfaces

In the coordinator's interface consists of a login interface, Upload interface, Insert student, Evaluator interface, Manage Student interface, Rubric interface and Report interface.

Coordinator will be redirected to their main page, which contain a list of coordinator options. Coordinator able to register students into database; the file should be in excel format which the extension .xls.

Then, coordinator should assign two evaluators to each student. Furthermore, coordinator will able to generate all student records into excel in .xls format. Coordinator also able to upload the file such as format of technical report, thesis, and etc for student and supervisor references.

3.1.1.2 Supervisor Interfaces

In the supervisor's interface consists of a login interface, Feedback Interface, Manage time interface, Assign mark interface, Approve Interface and Progress interface. First, supervisor should login to the system using their own username and password.

Supervisor will have redirected to their main page, which contains the list of supervisor options. Supervisor able to review the student progress and evaluate it. If have any problem and misunderstanding, supervisor shall give the feedback to their student.

Supervisor will monitoring their student and manage the time to meet them. Normally, supervisor will approve activity that submit by their students and assign mark to them based on the rubric stated.

3.1.1.3 Student Interfaces

In the student's interface consists of login interface, download interface, Check evaluator interface, Progress interface, Feedback interface, Apply form interface and Result interface.

First, student should login to the system using their own username and password. Before monitoring started, student should propose their title and select supervisor. After that, student will be redirected to their main page, which contains the list of student options.

Student shall be capable to update their project progress by weekly depends on dateline. Student able to get feedback from their supervisor when they are update the progress as they arise it. Student also able to download format of technical report, thesis and etc for guidelines. However, student able to check evaluator before they are present their project. Student should send their weekly report and upload full report in the PMS.

3.1.2 Hardware Interface

Not Applicable.

3.1.3   Software Interface

Table 3.1: Software Interface

Software	Purpose
Microsoft Windows Operating System • Windows 7 Ultimate	• As a platform for a system to run • Operating system which will be used to develop the system
• Microsoft Word 2007 • Microsoft PowerPoint 2007 • Microsoft Project 2007 • Microsoft Visio 2007	• Prepare proposal and documentation • Prepare slide for presentation • Scheduling, planning and prepare Gantt Chart • Design and draw chart and diagram
IBM Rational Rose Professional	Design and draw use case, sequence diagram
Adobe Dreamweaver 8	Design interface and generate coding
XAMPP/Apache MySQL phpMyAdmin	Database for the system; generate database, database management and database platform

3.2   Software Product Features

3.2.1   Use Case Login (SRS-REQ-PMS-2013-1-00)
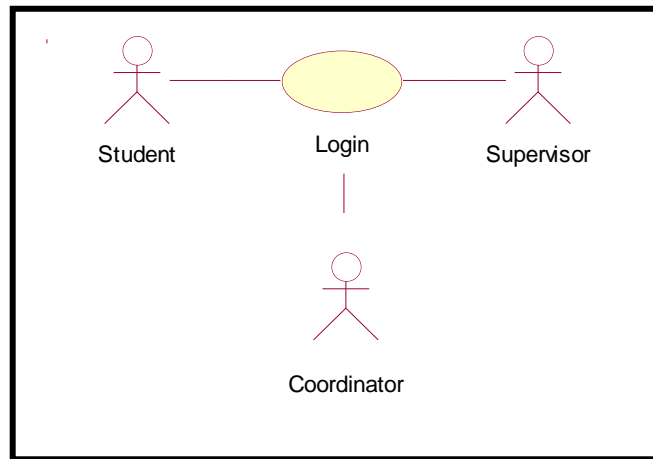
Figure 3.1 Login Use Case Diagram

Table 3.2 Login Use Case Diagram

Use Case:	Login System
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case is to login the Progress Monitoring System
Primary Actor:	Student, Supervisor, Coordinator
Supporting Actors:	▪ None
Stakeholders:	▪ None
Precondition:	Login the system to access the information through the system
Trigger:	None
Normal Flow:	Refer to **Appendix A-1 and Appendix B-1** 1. This use case shall starts when actors request system

	through their username and password. (SRS-REQ-PMS-2013-1-01)
	2. Cannot access the system if wrong username and password. (SRS-REQ-PMS-2013-1-02)
	3. The system displays the home page of particular user.
Sub-Flows:	▪ None
Alternate Flow/ Exceptions:	• All user have authority depends on their category (SRS-REQ-PMS-2013-1-03)
Post-Condition:	User can access system functionality
Non-Behavioral Requirements:	1. Three scopes are defined to allow for user access the system.
Activity Diagram :	Refer **Appendix A-1**
Sequence Diagram:	Refer **Appendix B-1**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.2   Submit Activities (SRS-REQ-PMS-2013-2-00)
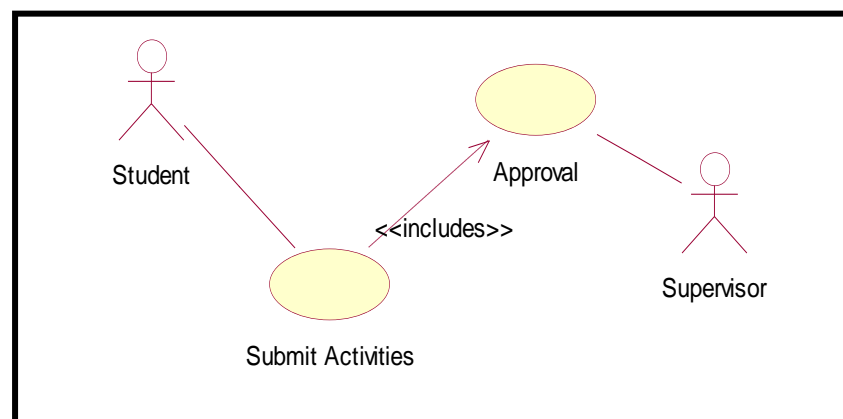
Figure 3.2 : Submit Activities Use Case

Table 3.3 : Submit Activities Use Case

Use Case:	Submit Activities
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case shall start when student submit the logbook activities. In this module, student will submit every activity done, the activities then will send to their supervisor. Student
Primary Actor:	Student, Supervisor
Supporting Actors:	▪ None
Stakeholders:	▪ None
Precondition:	Student should submit activities before get approval from their supervisor.
Trigger:	None
Normal Flow:	Refer to **Appendix A-2 and Appendix B-2** 1. This use case shall starts when Student register their title. **(SRS-REQ-PMS-2013-2-01)** 2. Student check approval on their title. **(SRS-REQ-PMS-2013-2-02)** 3. Students submit activities to supervisor. 4. Student can check approval from supervisor. . **(SRS-REQ-PMS-2013-2-03)** 5. Then, student is able to generate logbook .. **(SRS-REQ-PMS-2013-2-04)**
Sub-Flows:	▪ None
Alternate Flow/ Exceptions:	● All user have authority depends on their category
Post-Condition:	Supervisor and Student can check every activities submitted.
Activity Diagram :	Refer **Appendix A-2**
Sequence Diagram:	Refer **Appendix B-2**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali

Revision & Date	Revision 01, 06 October 2013

3.2.3　Progress (SRS-REQ-PMS-2013-3-00)
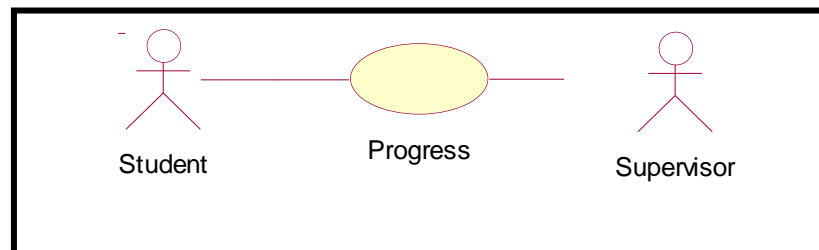
Figure 3.3 Progress

Table 3.4 Progress

Use Case:	Progress
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case is shown communication between student and supervisor. Student will submit their project progress and supervisor shall to review the student progress.
Primary Actor:	Student, Supervisor
Supporting Actors:	▪ None
Stakeholders:	▪ None
Precondition:	1. Student must submit their project progress by weekly. 2. Supervisor will view student progress. 3. Supervisor is able to give feedback after review project progress.
Trigger:	None
Normal Flow:	Refer **Appendix A-3 and Appendix B-3** 1. This use case shall starts when actors submit their project progress. 2. When project submitted, supervisor will check the project progress.

Sub-Flows:	▪ None
Alternate Flow/ Exceptions:	• All user have authority depends on their category
Post-Condition:	1. Project progress must submit by weekly and depends on dateline of FYP. 2. The progress also must sent by chapter and depends on schedule.
Activity Diagram :	Refer **Appendix A-3**
Sequence Diagram :	Refer **Appendix B-3**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.4   Feedback (SRS-REQ-PMS-2013-4-00)
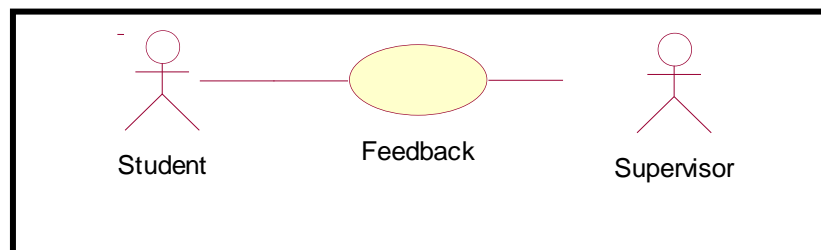
Figure 3.4 Feedback

Table 3.5 Feedback

Use Case:	**Feedback**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case is shown communication between student and supervisor. Student will get feedback from supervisor when submit their progress.
Primary Actor:	Student, Supervisor
Supporting	▪ None

Actors:	
Stakeholders:	▪ None
Precondition:	1. Student must submit their project progress by weekly. 2. Supervisor will view student progress. 3. Once student submit their progress, student will get feedback from supervisor.
Trigger:	None
Normal Flow:	1. This use case shall starts when actors submit their project progress. 2. When student submit their project, supervisor will check and update the problem as they arise it. 3. Supervisor able to give feedback if have any problem or student misunderstanding about their project.
Sub-Flows:	▪ None
Alternate Flow/ Exceptions:	• All user have authority depends on their category
Post-Condition:	1. Project progress must submit by weekly and depends on dateline of FYP. 2. Student will get feedback from their supervisor.
Activity Diagram :	Refer **Appendix A-4**
Sequence Diagram :	Refer **Appendix B-4**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.5   View Result (SRS-REQ-PMS-2013-5-00)
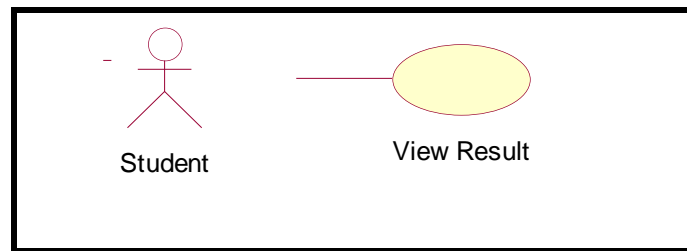
Figure 3.5 View Result

Table 3.6 View Result

Use Case:	View Result
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case is shown
Primary Actor:	Student
Supporting Actors:	▪ Supervisor
Stakeholders:	▪ None
Precondition:	1. Student must submit their project progress by weekly. 2. Supervisor should evaluate student based on rubric. 3. Once student completed submit their project progress, student able to view their result.
Trigger:	None
Normal Flow:	Refer **Appendix A-5** 1. This use case shall starts when actors submit their project progress. 2. Student able to view their result after they are completed and submit their project progress.
Sub-Flows:	▪ None
Post-Condition:	1. Project progress must submit by weekly and depends on dateline of FYP. 2. Student should submit full report of project progress.
Activity Diagram :	Refer **Appendix A-5**

Sequence Diagram :	Refer **Appendix B-5**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.6   Assign Mark (SRS-REQ-PMS-2013-6-00)
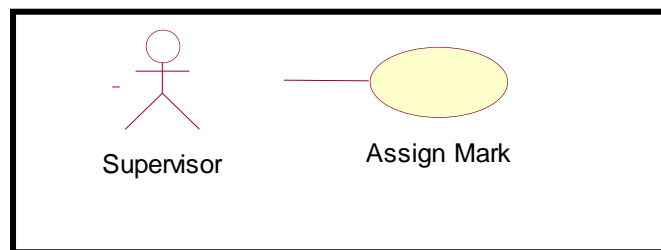
Figure 3.6: Assign Mark

Table 3.7 Assign Mark

Use Case:	**Assign Mark**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when supervisor assign marks to their student. Student will check their result after supervisor release out the marks.
Primary Actor:	Supervisor
Supporting Actors:	▪ None
Stakeholders:	▪ None
Precondition:	1. Student must submit their project progress by weekly. 2. Supervisor should evaluate student based on rubric. 3. Once student completed submit their project progress, student able to view their result.
Trigger:	None
Normal Flow:	Refer **Appendix A-6** 1. This use case shall starts when actors submit their

	project progress.
	2. Student able to view their result after they are completed and submit their project progress.
Sub-Flows:	▪ None
Post-Condition:	1. Project progress must submit by weekly and depends on dateline of FYP.
	2. Student should submit full report of project progress.
Activity Diagram :	Refer **Appendix A-6**
Sequence Diagram :	Refer **Appendix B-6**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.7   Manage Schedule (SRS-REQ-PMS-2013-7-00)
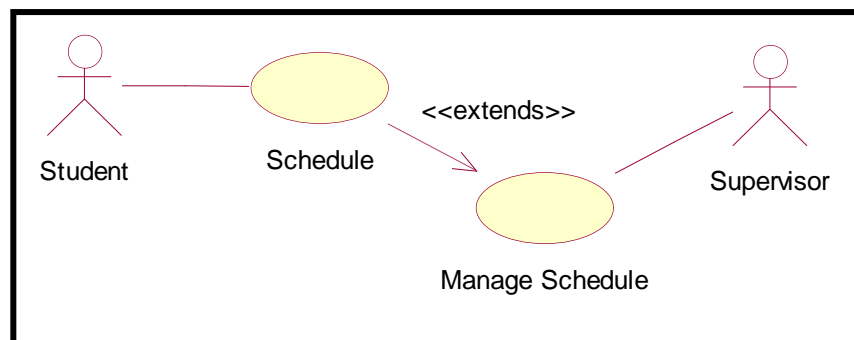
Figure 3.7 Manage Schedule

Table 3.8 Manage Schedule

Use Case:	**Manage Schedule**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when supervisor manage time to their student. Its mean, supervisor set schedule when him/her want to meet face to face their student.

Primary Actor:	Supervisor
Supporting Actors:	▪ Student
Precondition:	1. Supervisor should arrange their time and set the time for meet their student to see student progress.
Trigger:	None
Normal Flow:	Refer **Appendix A-7** 1. This use case shall starts when actors want to see their students. 2. Supervisor will set the schedule and arrange it. 3. Student able to view the schedule.
Sub-Flows:	▪ None
Post-Condition:	None
Activity Diagram :	Refer **Appendix A-7**
Sequence Diagram :	Refer **Appendix B-7**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.8   Upload and Download (SRS-REQ-PMS-2013-8-00)
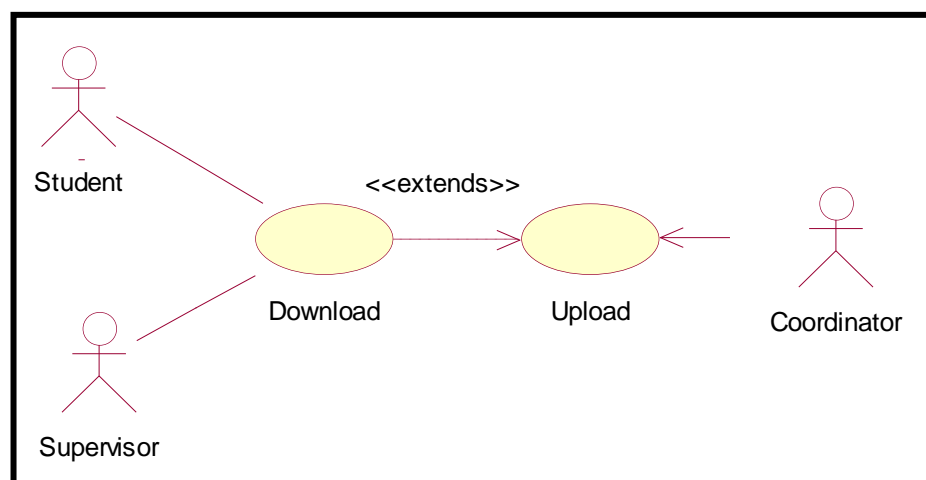
Figure 3.8: Upload and Download

Table 3.9 : Upload and Download

Use Case:	Upload and Download
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when coordinator upload the format of technical report, thesis, rubric and others. Student and supervisor able to download the uploaded files by coordinator.
Primary Actor:	Coordinator, Student, Supervisor
Supporting Actors:	▪ Student, Supervisor, Coordinator
Precondition:	1. First, coordinator should upload the files. 2. Coordinator can delete and edit the files.
Trigger:	None
Normal Flow:	Refer **Appendix A-8** 1. This use case shall starts when coordinator upload the files. 2. Supervisor and student able to download the uploaded files by coordinator.
Sub-Flows:	▪ None
Post-Condition:	None
Activity Diagram :	Refer **Appendix A-8**
Sequence Diagram :	Refer **Appendix B-8**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.9   Insert Student (SRS-REQ-PMS-2013-9-00)
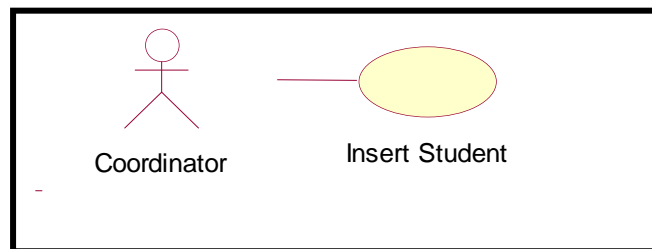
Figure 3.9: Insert Student

Table 3.10 : Insert Student

Use Case:	Insert Student
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when coordinator insert all data student into database. The file should be in excel format which the extension .xls.
Primary Actor:	Coordinator
Supporting Actors:	▪ None
Precondition:	1. The file should be in excel format which the extension .xls
Trigger:	None
Normal Flow:	Refer **Appendix A-9** 1. This use case shall starts when coordinator insert all data student into database. 2. Student will access this system easily.
Sub-Flows:	▪ None
Post-Condition:	None
Activity Diagram :	Refer **Appendix A-9**

Sequence Diagram :	Refer **Appendix B-9**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.10 Assign Evaluator (SRS-REQ-PMS-2013-10-00)
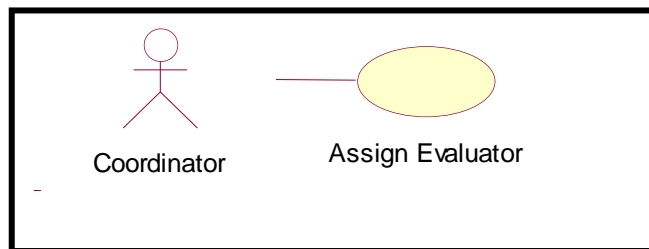
Figure 3.10: Assign Evaluator

Table 3.11 : Assign Evaluator

Use Case:	**Assign Evaluator**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when coordinator assign evaluator to student.
Primary Actor:	Coordinator
Supporting Actors:	▪ Student
Precondition:	1. Coordinator must assign two evaluator for each single student. Evaluator can't be same.
Trigger:	None
Normal Flow:	1. This use case shall starts when coordinator assign evaluator to each students. 2. Student able to check evaluator before they are present their project.
Sub-Flows:	▪ None
Post-Condition:	None

Activity Diagram :	Refer **Appendix A-10**
Sequence Diagram :	Refer **Appendix B-10**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.11 Manage Student (SRS-REQ-PMS-2013-11-00)
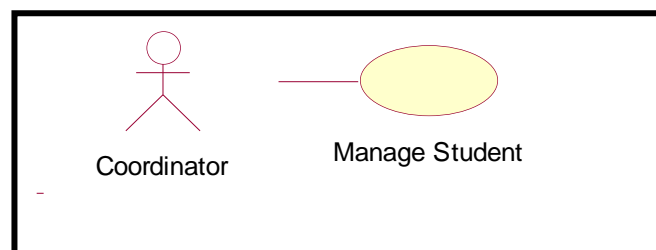
Figure 3.11 : Manage Student

Table 3.12 : Manage Student

Use Case:	**Manage Student**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when coordinator insert data student, and search student. Coordinator have authority in manage database.
Primary Actor:	Coordinator
Supporting Actors:	▪ None
Precondition:	None
Trigger:	None
Normal Flow:	1. This use case shall starts when coordinator manage data in database.
Sub-Flows:	▪ None

Post-Condition:	None
Activity Diagram :	Refer **Appendix A-11**
Sequence Diagram :	Refer **Appendix B-11**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2012

3.2.12 Set Timeline (SRS-REQ-PMS-2013-12-00)
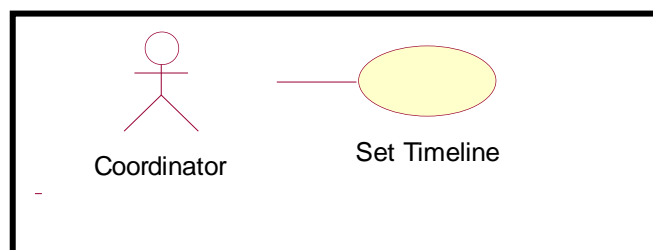
Figure 3.12 : Set Timeline

Table 3.13 : Set Timeline

Use Case:	Set Timeline
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case shall start when coordinator set the timeline.
Primary Actor:	Coordinator
Supporting Actors:	▪ None
Precondition:	Coordinator must set the timeline based on PSM schedule
Trigger:	None
Normal Flow:	This use case shall starts when coordinator manage data in database.
Sub-Flows:	▪ None

Post-Condition:	None
Activity Diagram :	Refer **Appendix A-12**
Sequence Diagram :	Refer **Appendix B-12**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.13 Generate Record (SRS-REQ-PMS-2013-13-00)
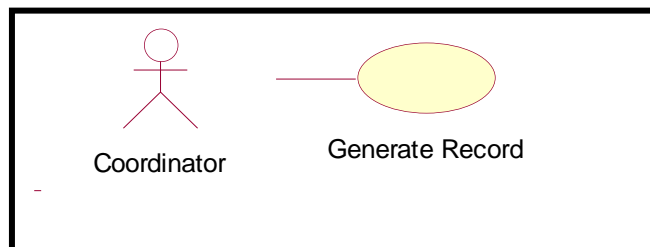
Figure 3.13 : Generate Record

Table 3.14 : Generate Record

Use Case:	**Generate Record**
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	The purpose of this use case start when coordinator will able to generate all student records into excel in .xls format.
Primary Actor:	Coordinator
Supporting Actors:	▪ None
Precondition:	None
Trigger:	None
Normal	Refer **Appendix A-13**

Flow:	This use case shall starts when coordinator manage data in database. Coordinator will able to generate all student records into excel in .xls format.
Sub-Flows:	▪ None
Post-Condition:	None
Activity Diagram :	Refer **Appendix A-13**
Sequence Diagram:	Refer **Appendix B-13**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.2.14 Approve (SRS-REQ-PMS-2013-13-00)
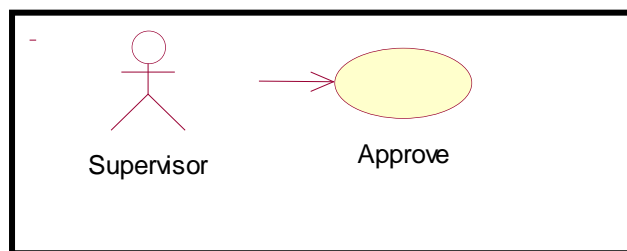
Figure 3.14 : Approve

Table 3.15 : Approve

Use Case:	Approve
ID:	PMS001
Scope:	Progress Monitoring System for Student Final Year Project
Priority:	Essential
Summary:	This use case show the supervisor approves or rejects activities submitted by their students.
Primary Actor:	Coordinator
Supporting Actors:	▪ None
Precondition:	At least one activity had been submitted

Trigger:	None
Normal Flow:	Refer **Appendix A-13** 1. Supervisor check students activities 2. Supervisor approve or reject that activities 3. Supervisor click submit
Sub-Flows:	▪ None
Post-Condition:	Student can view their status of activities submitted.
Activity Diagram :	Refer **Appendix A-13**
Sequence Diagram:	Refer **Appendix B-13**
Source:	Requirement Statement
Author	Siti NurHidayah Bt Mohamad Khazali
Revision & Date	Revision 01, 06 October 2013

3.3 Requirements Traceability

Table 3.16 : Requirement Traceability

Requirement ID	Description
(SRS-REQ-PMS-2013-1-00) **(SRS-REQ-PMS-2013-1-01)**	**Login** • At the Home Page interface, user needs to login first by entering username, password and category (coordinator, Student and supervisor) in the Login section to open the Index Menu under his / her account. • PMS displays the Index menu page. • Before login, student must register their information.
(SRS-REQ-PMS-2013-2-00)	**Submit Activities**

	• Allow lecturer to Approve/Resubmit activities submitted by their supervisee • Student will submit their activities to supervisor and to get approval. • The student's logbook will be shown in .pdf.
(SRS-REQ-PMS-2013-3-00)	**Progress** • Student will submit their project progress and supervisor shall to review the student progress. • Student must submit their project progress by weekly. • The progress also must sent by chapter and depends on schedule.
(SRS-REQ-PMS-2013-4-00)	**Feedback** • Student will get feedback from supervisor when submit their progress. • Student must submit their project progress by weekly. • Once student submit their progress, student will get feedback from supervisor. • When student submit their project, supervisor will check and update the problem as they arise it.
(SRS-REQ-PMS-2013-5-00)	**View Result** • Student must submit their project progress by weekly. • Supervisor should evaluate student based on rubric. • Student able to view their result after they are

	completed and submit their project progress.
(SRS-REQ-PMS-2013-6-00)	**Assign Mark** • Lecturers who play as supervisor and evaluator will assign marks to student based on the rubric. • The system displays list of approved students and a form marks rubric will displayed. • Lecturers choose to supervise or evaluate their students.
(SRS-REQ-PMS-2013-7-00)	**Manage Schedule** • Supervisor set the time and arrange it for meet their student.
(SRS-REQ-PMS-2013-8-00)	**Upload and Download** • Coordinator uploads the format of technical report, thesis, rubric and others. • Student and supervisor able to download the uploaded files by coordinator.
(SRS-REQ-PMS-2013-9-00)	**Insert Student** • Coordinator insert all data student into database • The file should be in excel format which the extension .xls.
(SRS-REQ-PMS-2013-10-00)	**Assign Evaluator** • Coordinator will choose two lecturer to be the evaluator of a student. • Evaluator can't be same. • Student able to check evaluator before they are present their project.

(SRS-REQ-PMS-2013-11-00)	**Manage Student** • Coordinator has authority in manage database. • Coordinator is able to add, delete and search student information.
(SRS-REQ-PMS-2013-12-00)	**Set Timeline** • Coordinator is able to set the project timeline. • Coordinator will view and deleted the timeline.
(SRS-REQ-PMS-2013-13-00)	**Generate Record** • Coordinator will able to generate all student records into excel in .xls format.
(SRS-REQ-PMS-2013-14-00)	**Approve** • Supervisor approves or resubmit activities submitted by their students • Lecturers check student activities. • Student can view their status of activities submitted.

4. ACRONYMS AND ABBREVIATION

Table 4.1: Definition

Name	Definition
FYP	Final Year Project
PMS	PMS stands for Progress Monitoring System for Student Final Year Project.
Supervisor	The person who is charge of grading student's weekly progress based on rubric.
Coordinator	A person who is charge of assigning students to supervisors in she/he faculty.
Student	The user will be updating the logbook and send their progress by weekly report.
IEEE	The Institute of Electrical and Electronic Engineers (IEEE) is an international non-profit, profesional organization for the advancement of technology related to electricity.

APPENDIX A

Activity Diagram

A-1: Normal Flow of Login Activity Diagram
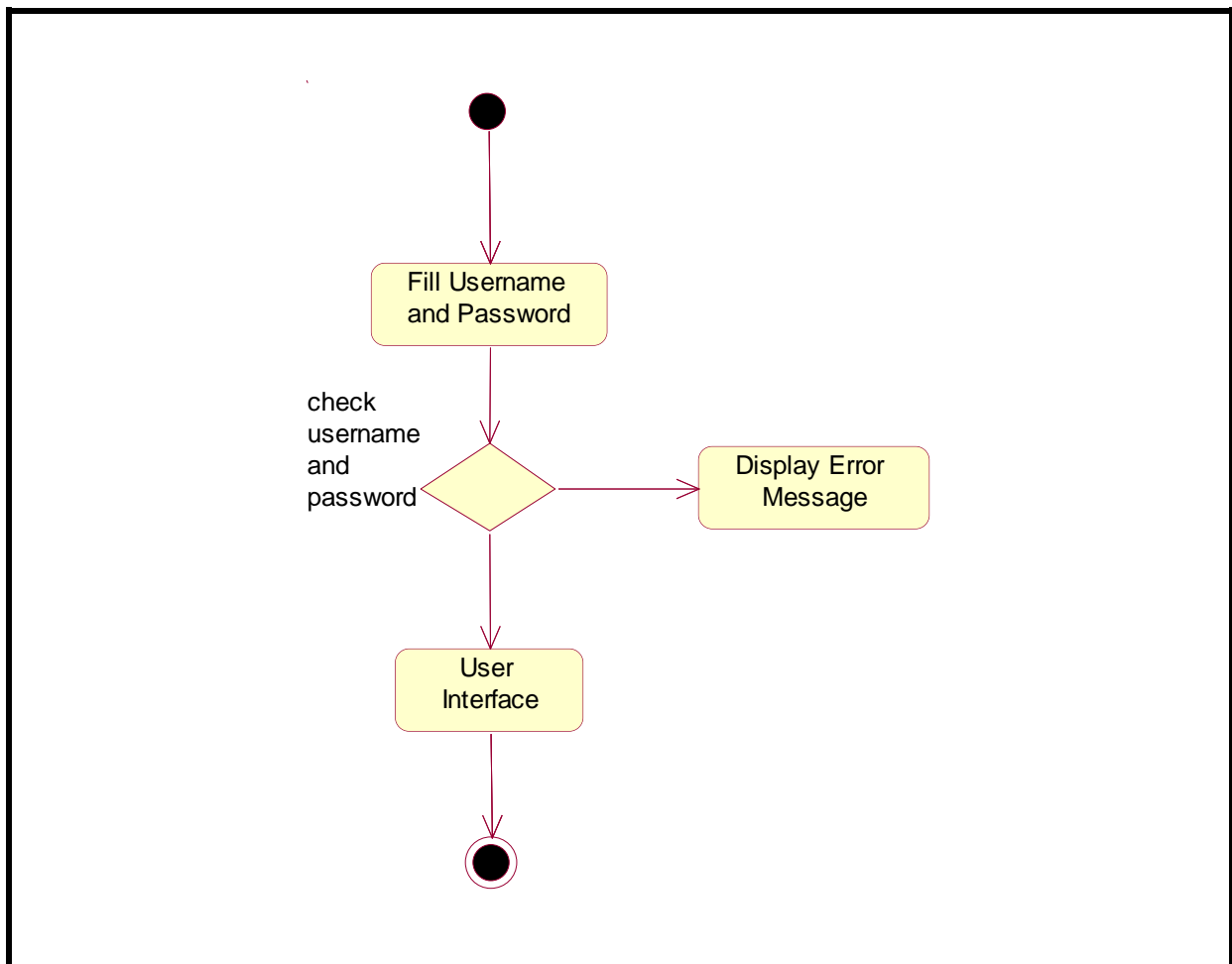
A-2: Normal Flow of Submit Activities Diagram

```
  ●  →  Choose   →  Upload File  →  Submit      →  ◉
        Submit                      Activities
```
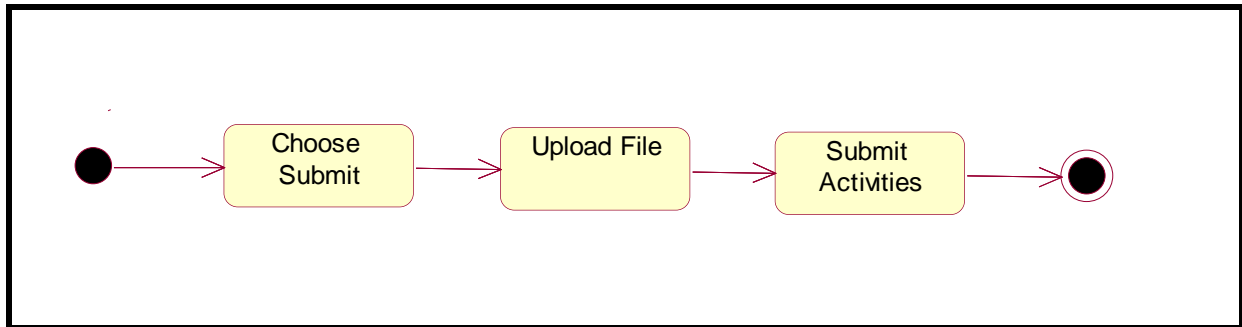
A-3: Normal Flow of Progress Diagram

```
        ●
        │
        ▼                           Check
     Progress  ──────────────►  ◇  dateline  ◄──────────┐
                               (Check                    │
                               dateline)                 │
                                   │                      │
                                   ▼                      │
                              Progress                    │
                              by chapter                  │
                                   │                      │
   Check                           ▼                  Update
   Progress               ◇  ──────────────►         Progress
                                   │
                                   ▼
                              Completion
                              Progress
                                   │
                                   ▼
                                   ◉
```

A-4: Normal Flow Feedback Activity Diagram
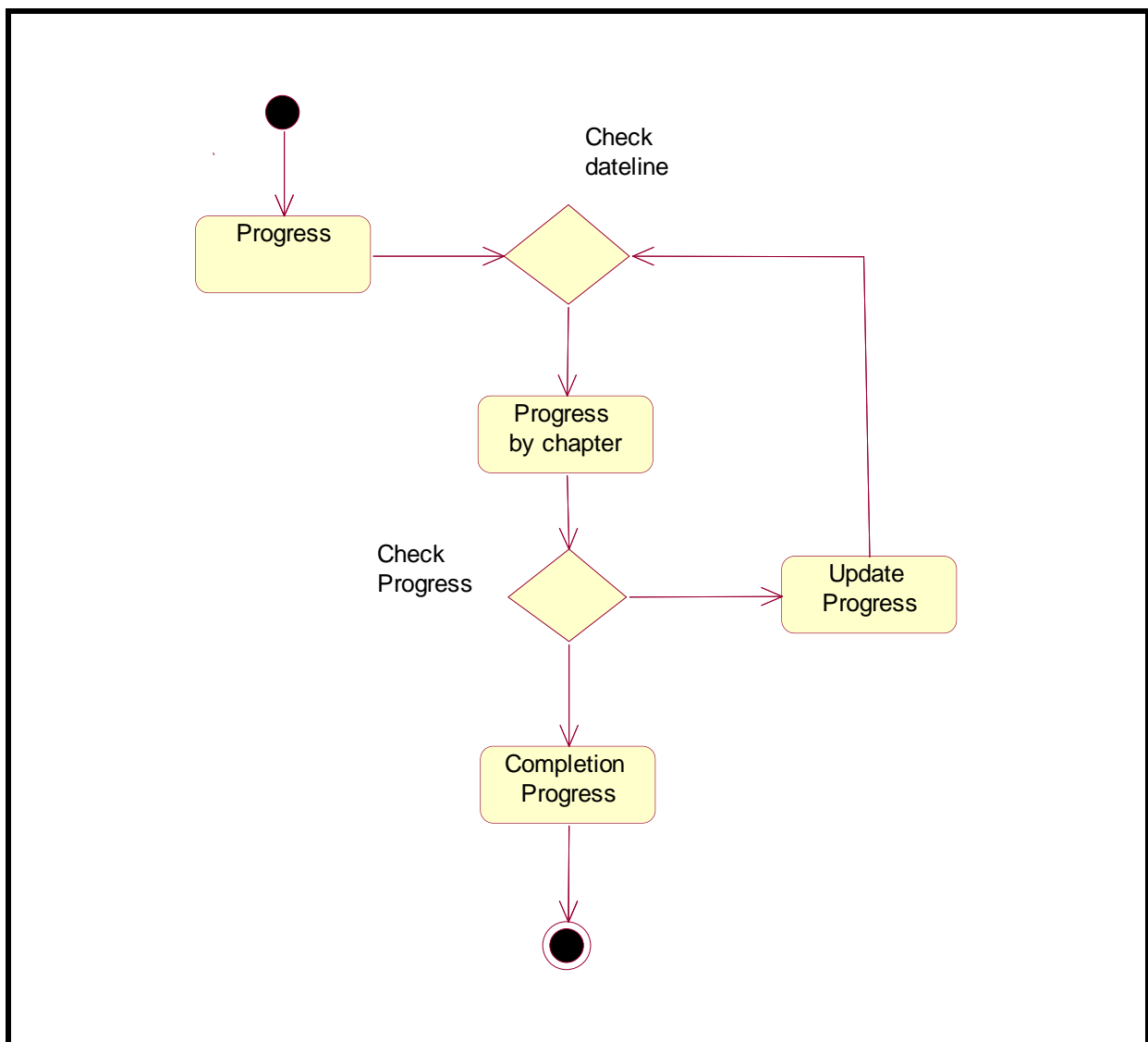
A-4: Normal Flow Feedback Activity Diagram
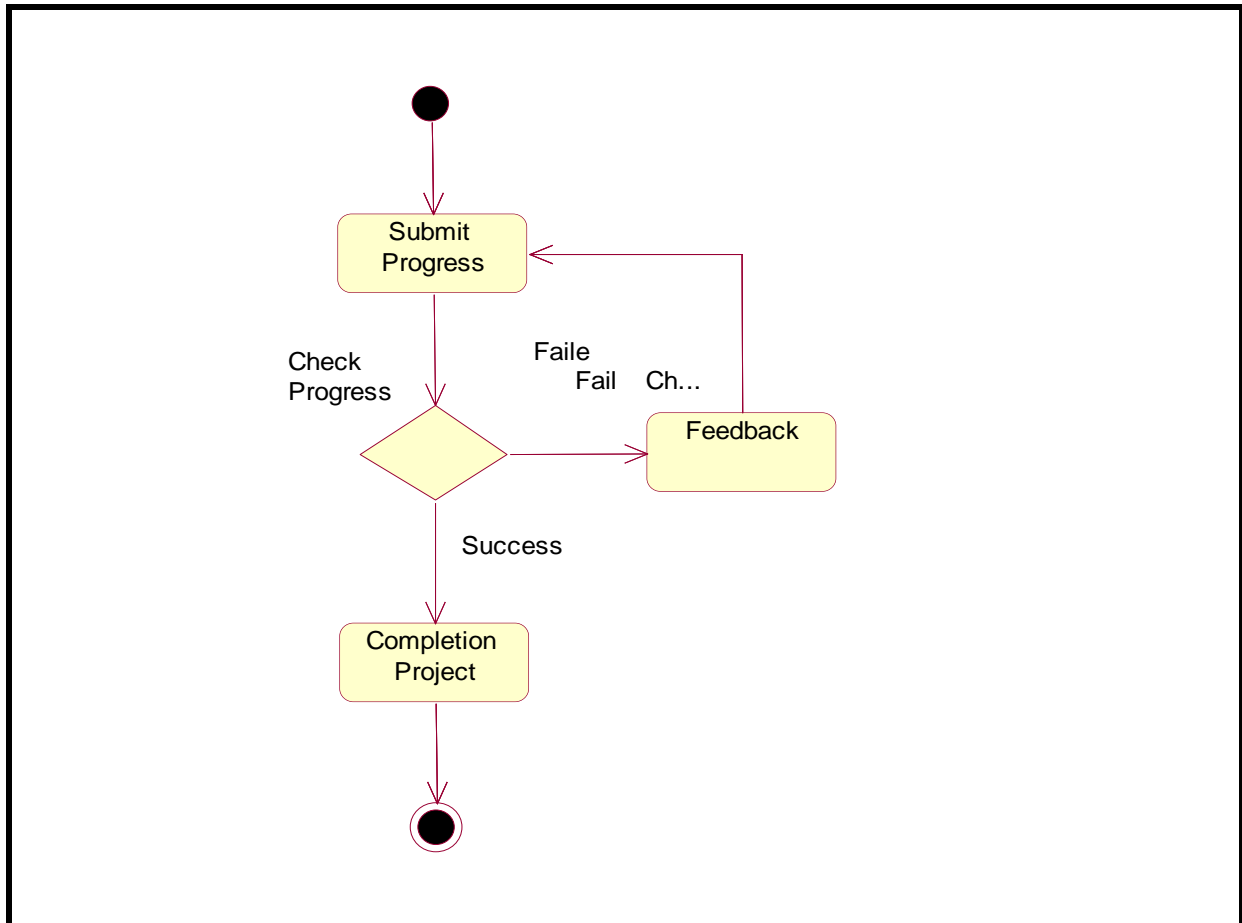
A-5: Normal Flow View Result Activity Diagram
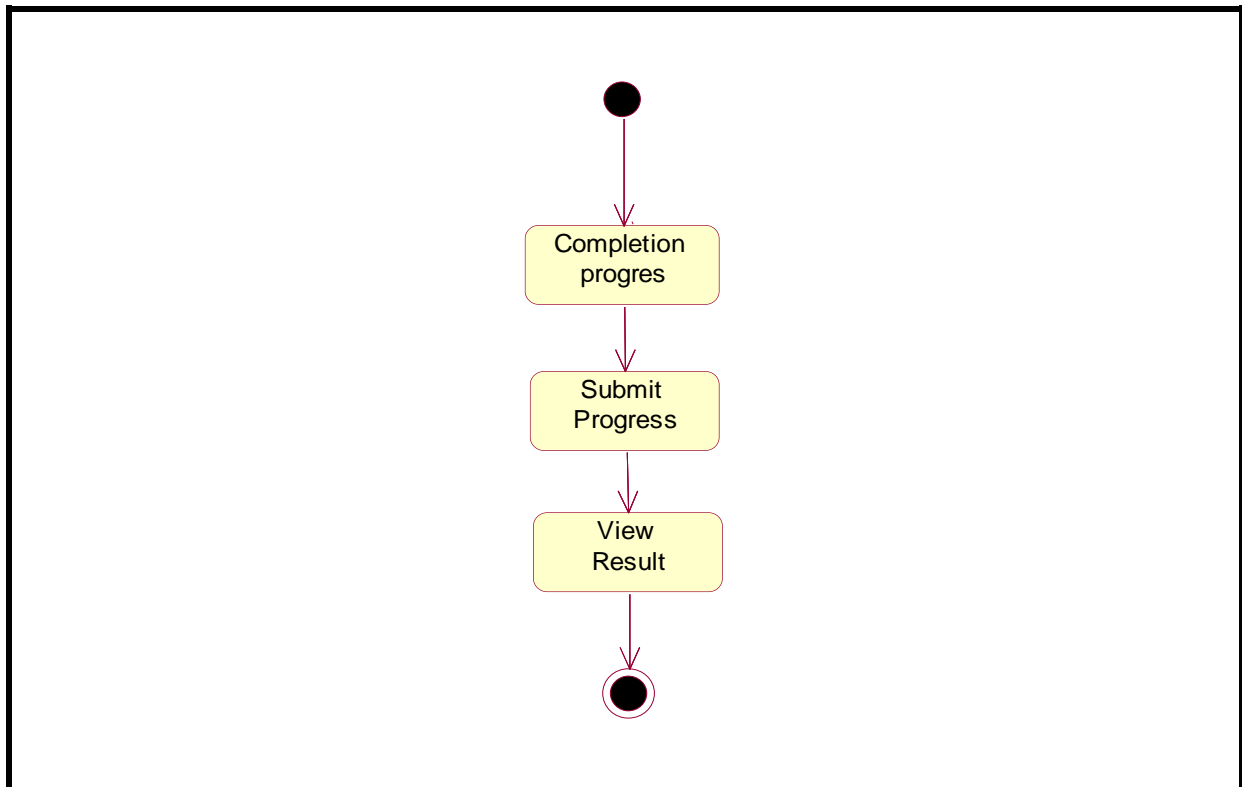
A-6 : Normal Flow Assign Mark Activity Diagram
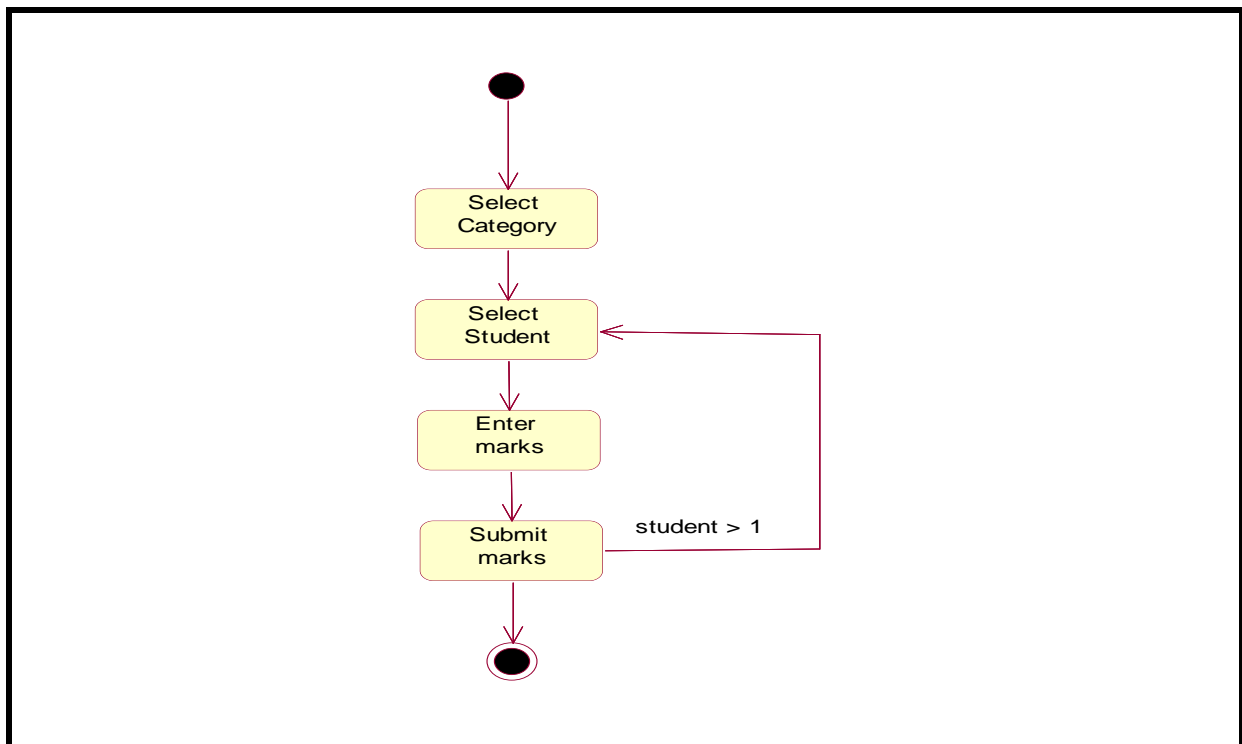
A-7: Normal Flow Manage Schedule Activity Diagram
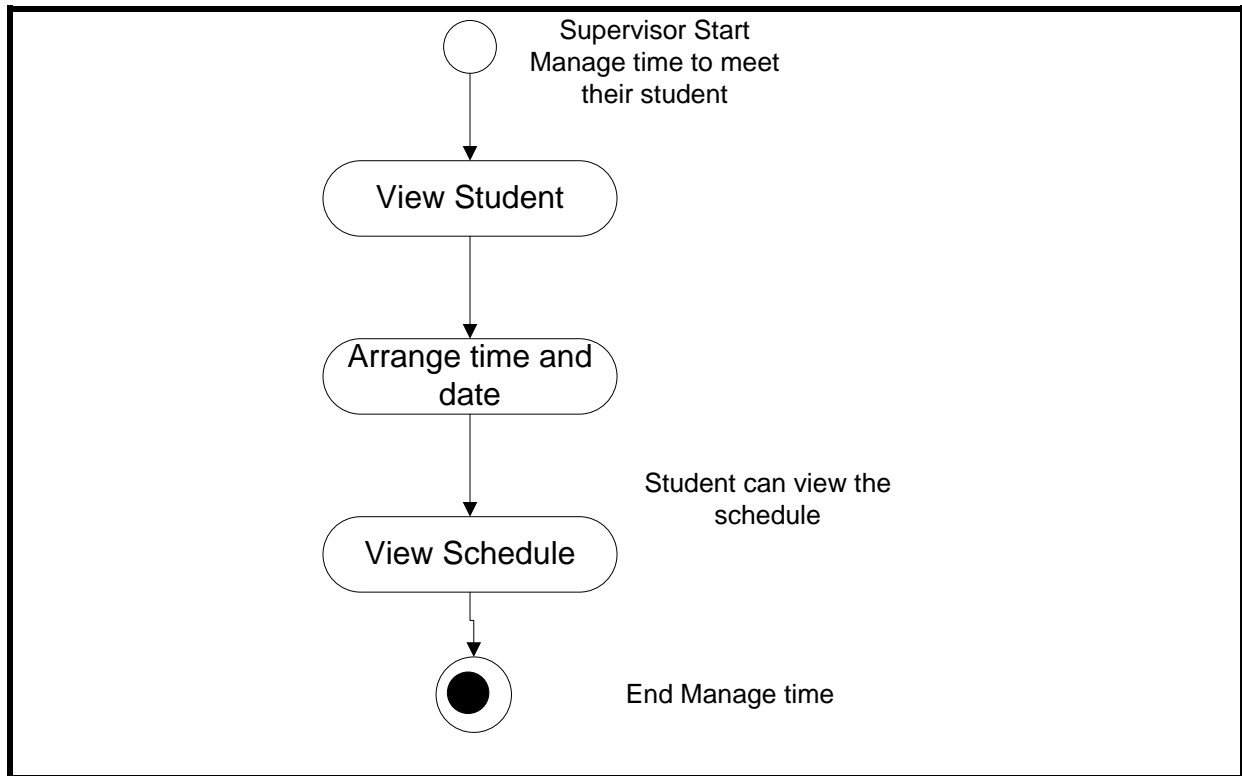
A8: Normal Flow Upload and Download Activity Diagram
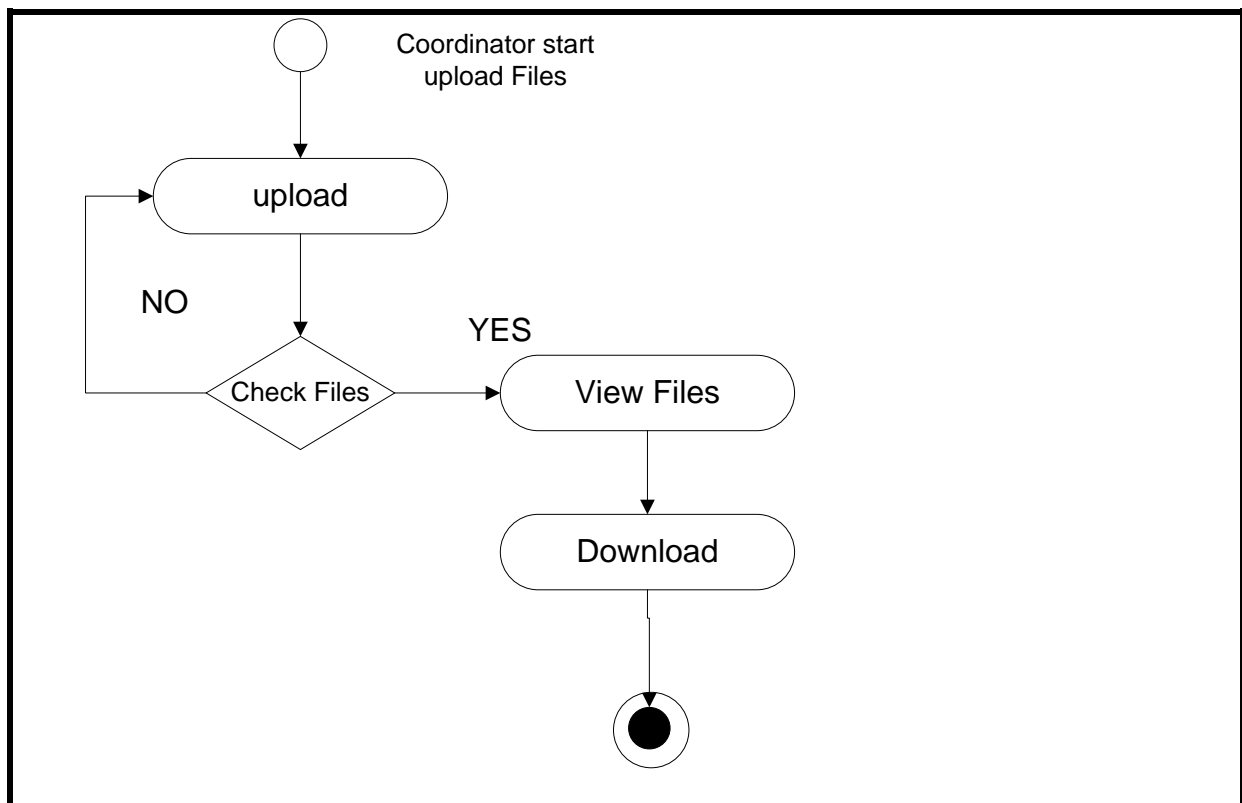
A-9 : Normal Flow Insert Student Activity Diagram
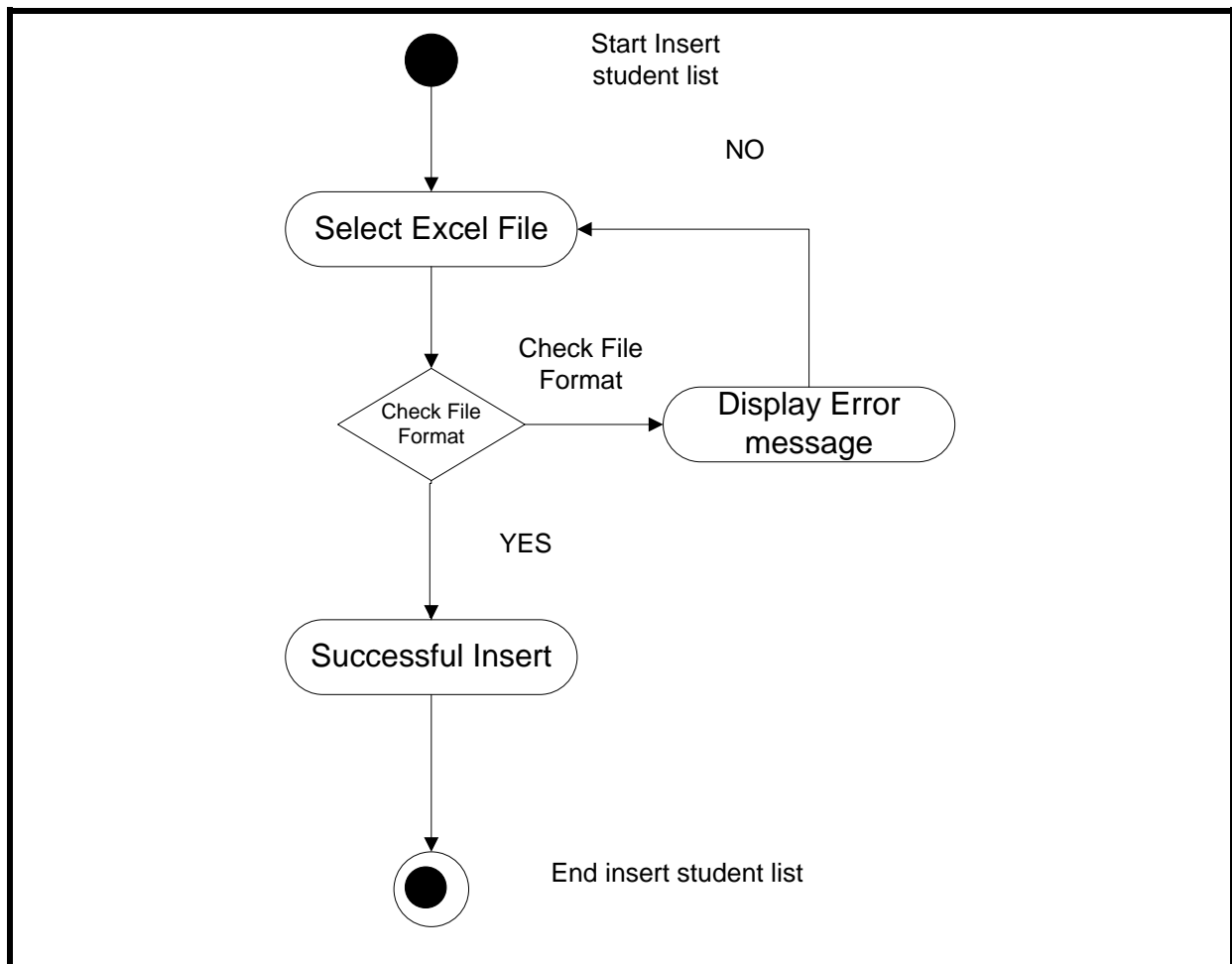
A-10: Normal Flow Assign Evaluator Activity Diagram
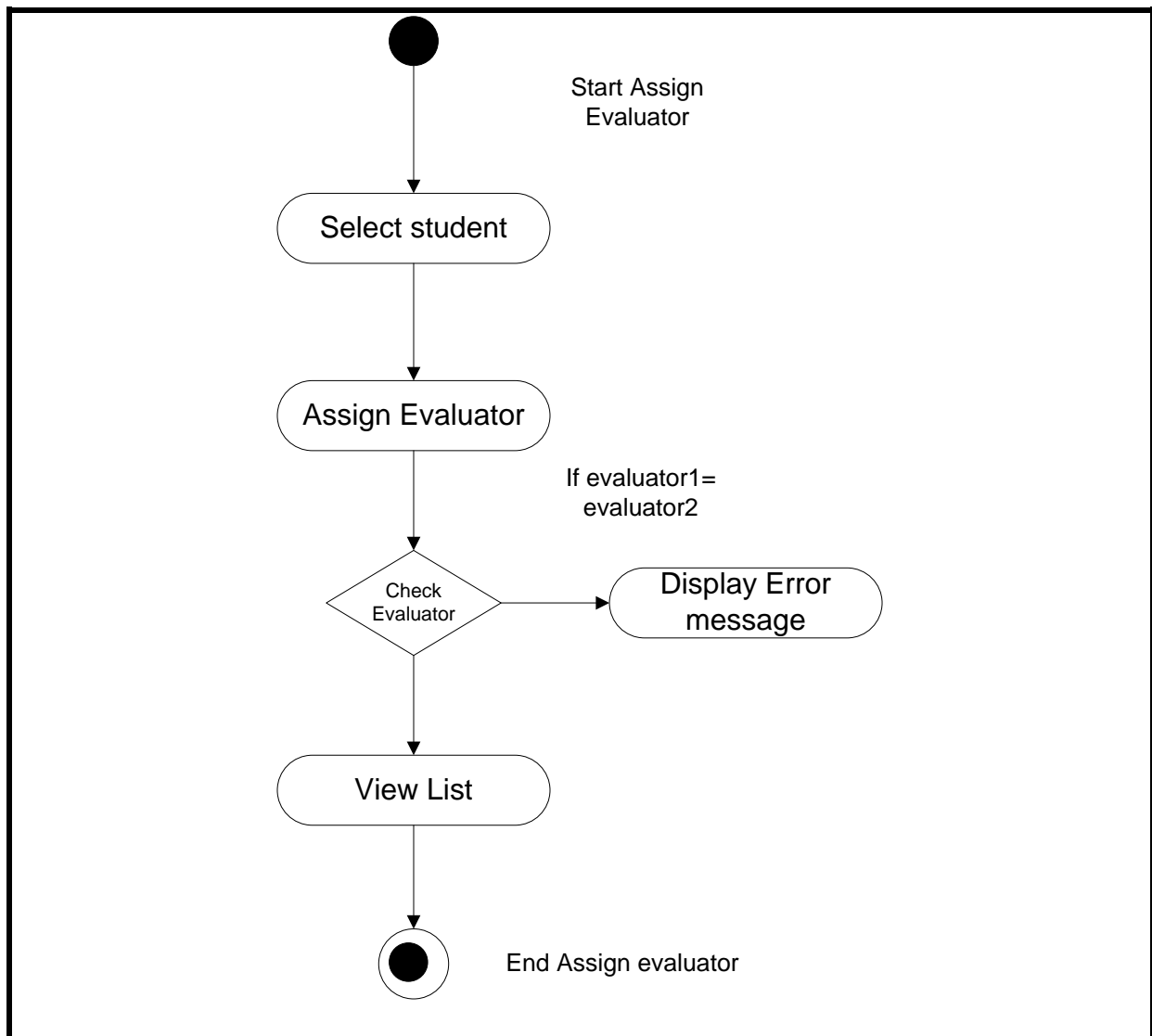
A-11 : Normal Flow Manage Student Activity Diagram
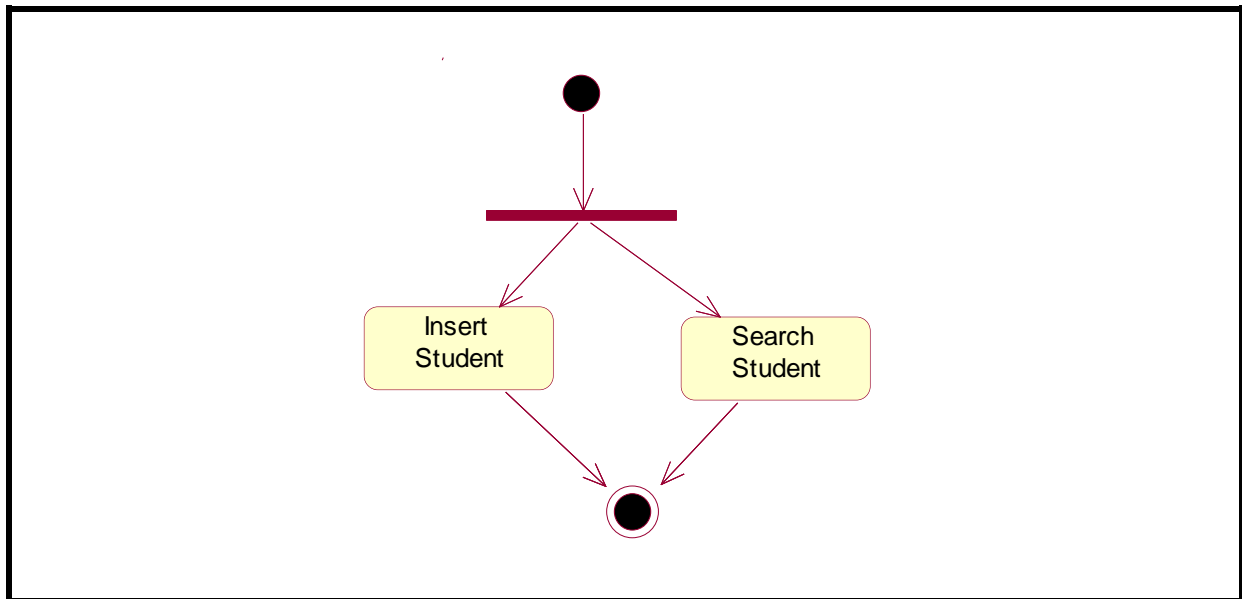
A-12: Normal Flow Set Timeline Activity Diagram
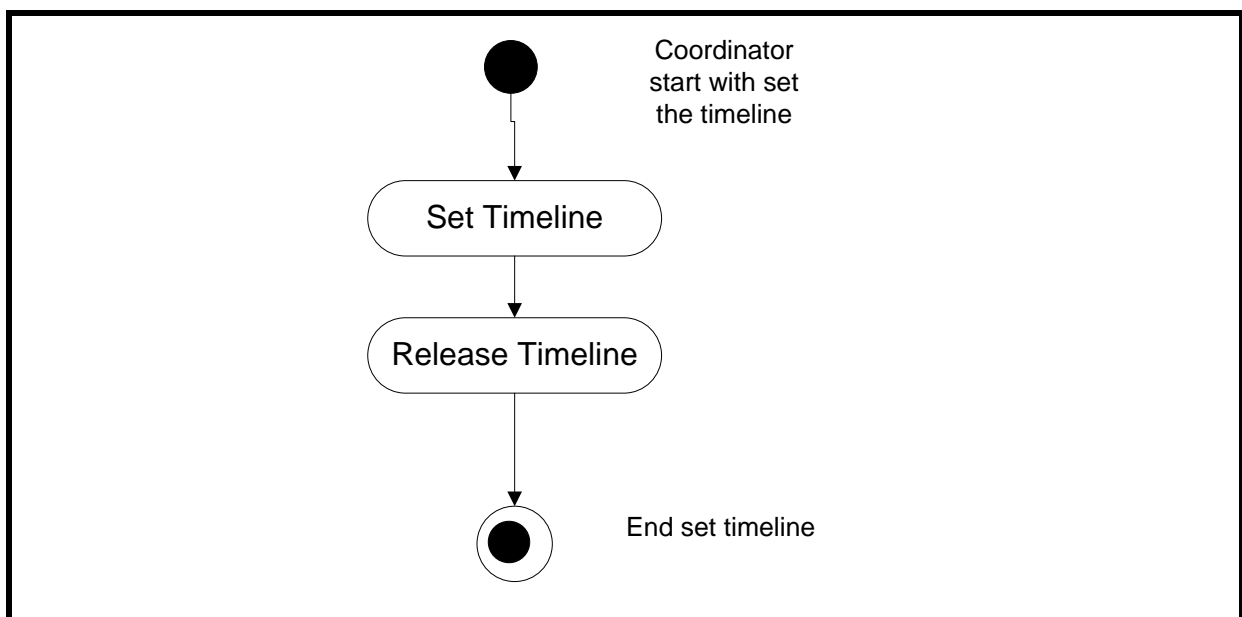
A-13 : Normal Flow Generate Record Activity Diagram
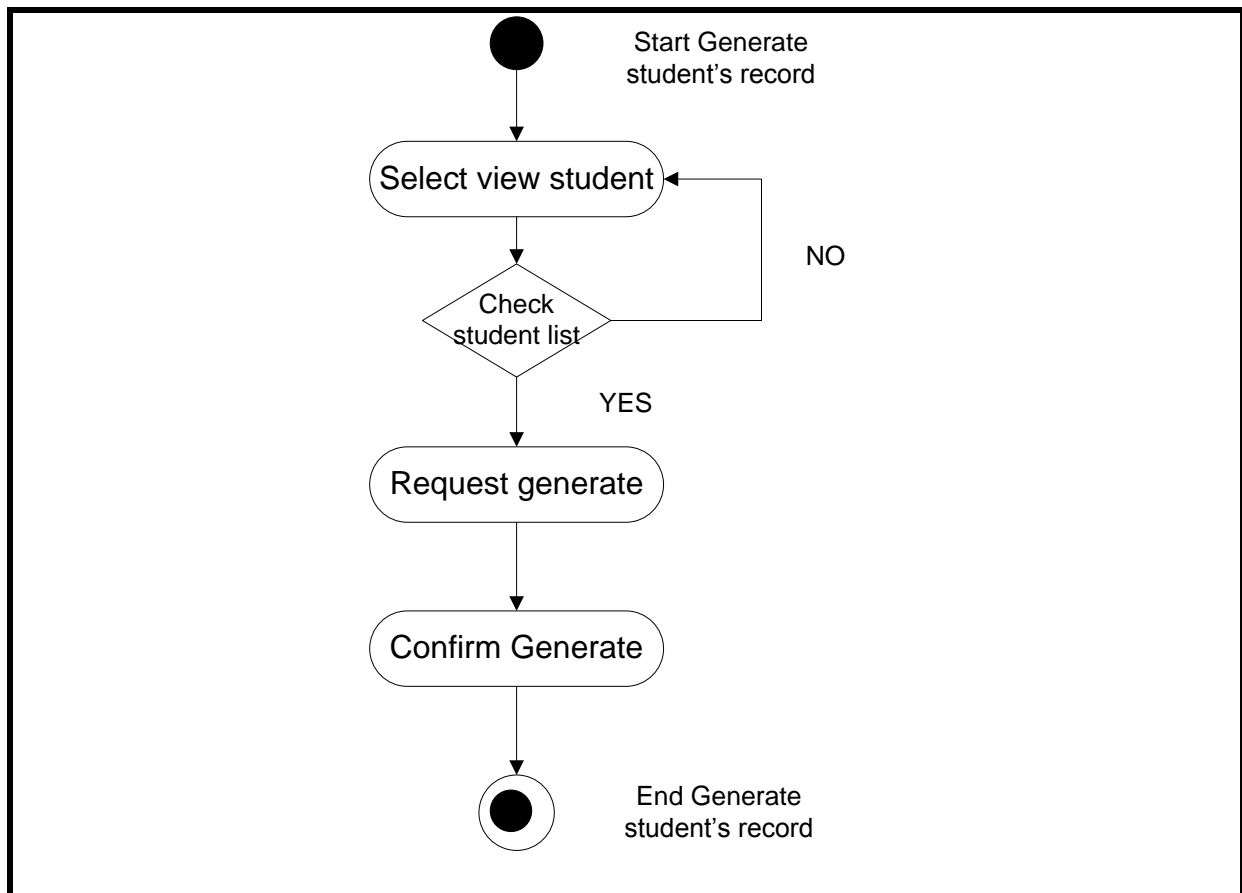
A-14 : Normal Flow Approve Activity Diagram
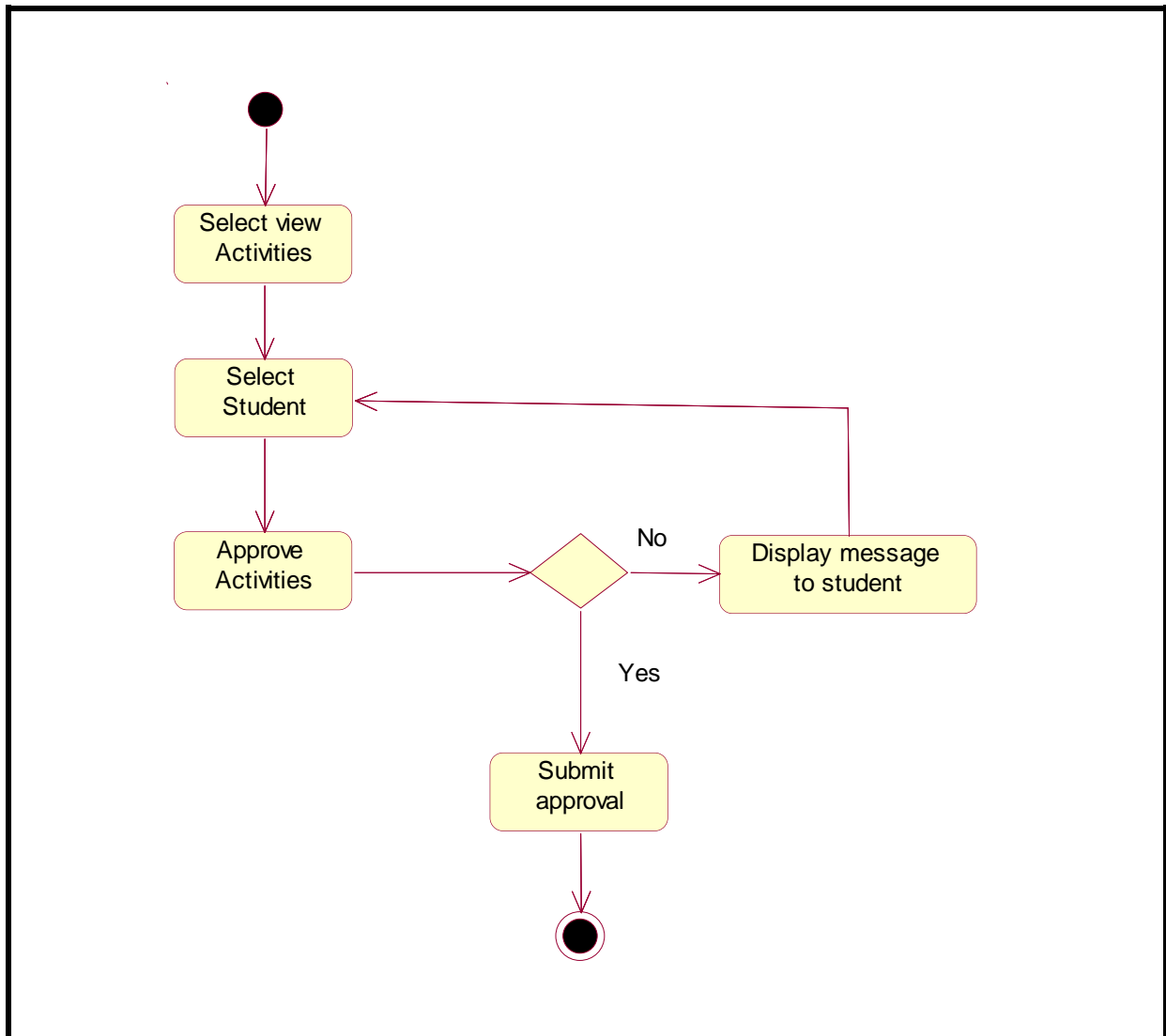
APPENDIX B

Sequence Diagram

B-1 : Login Sequence Diagram
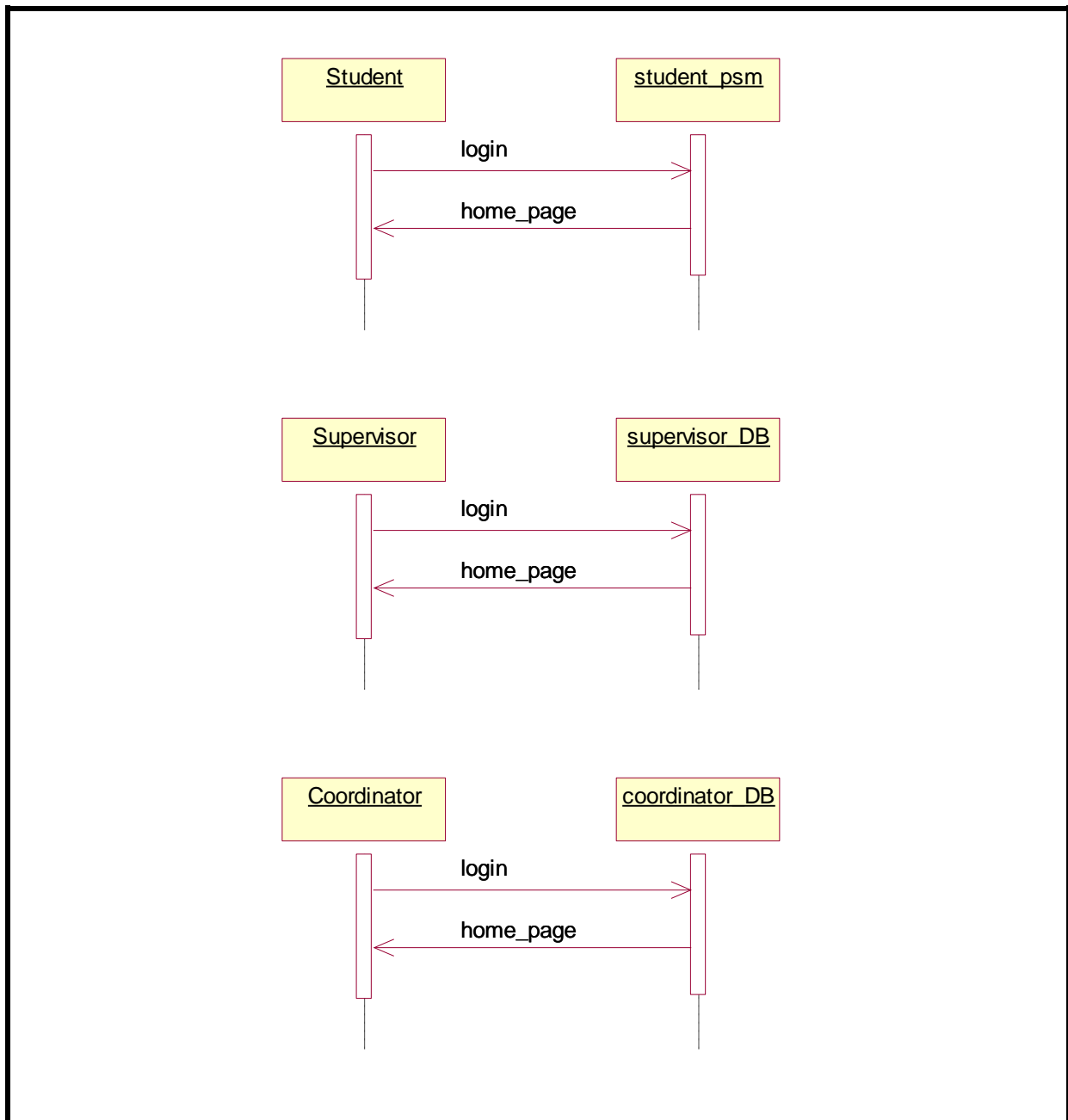
B-2 : Submit Activities Sequence Diagram
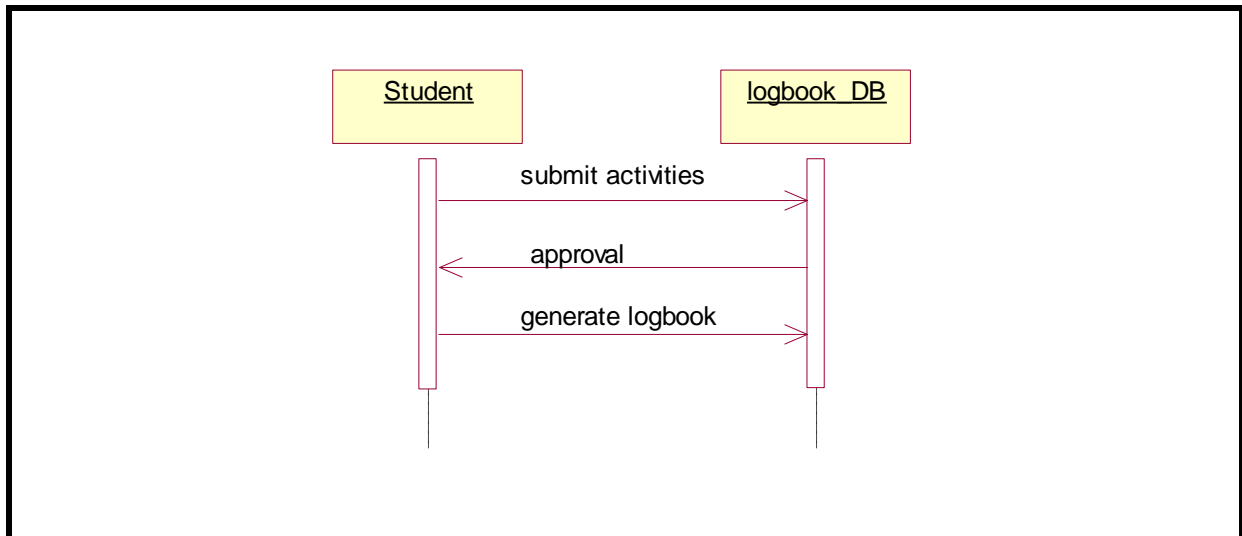
B-3 : Progress Sequence Diagram
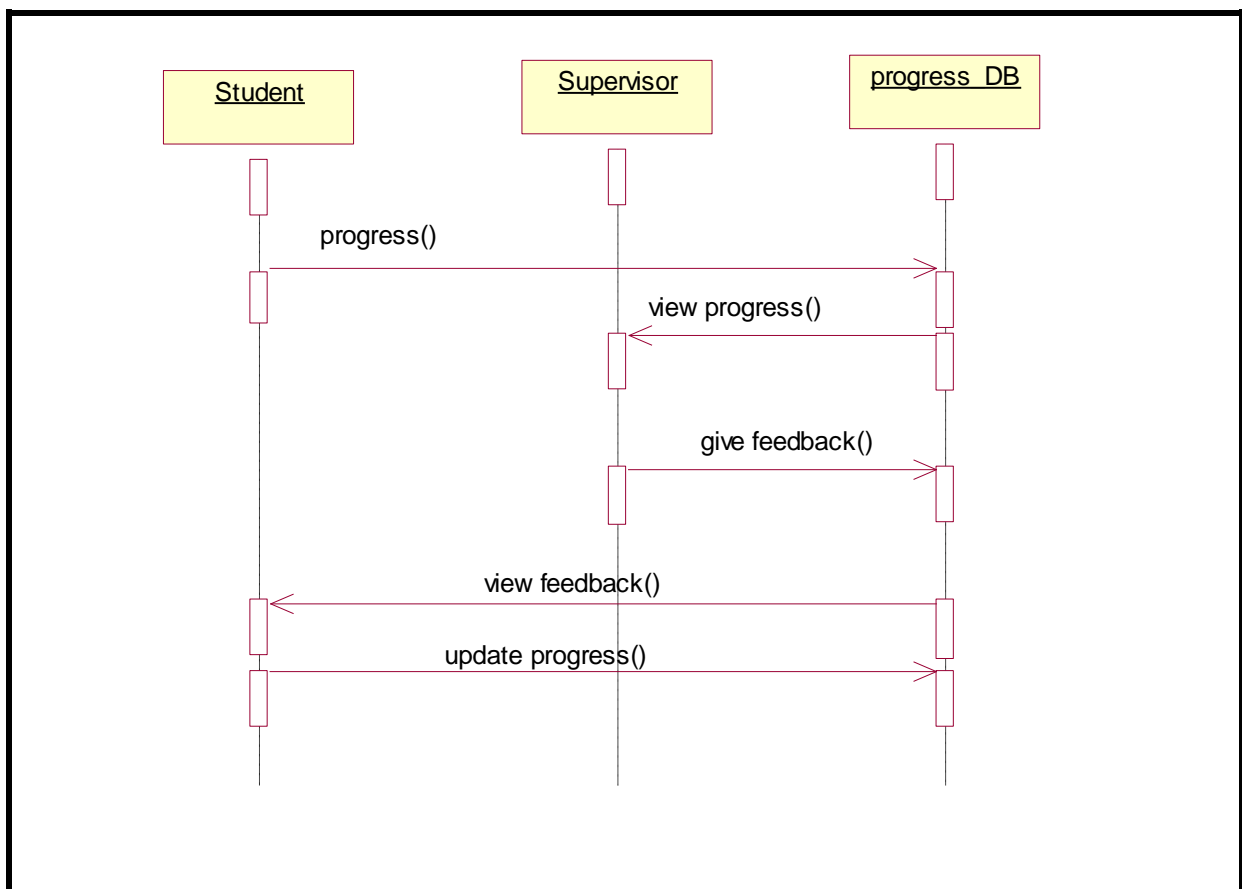
B-4 : Feedback Sequence Diagram
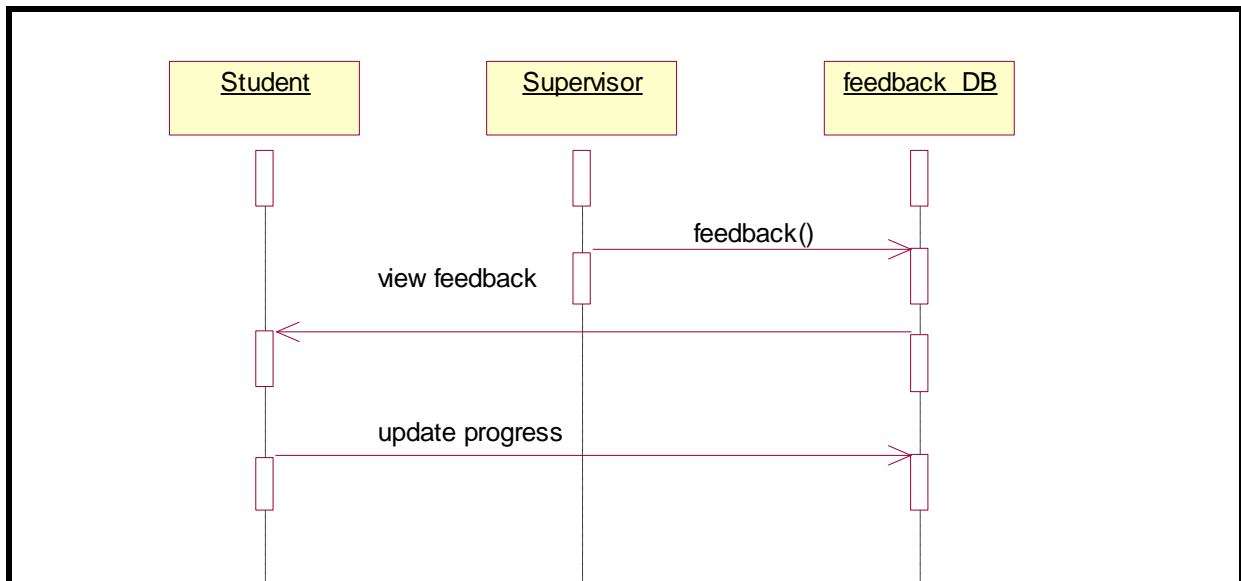
B-5 : View Result Sequence Diagram
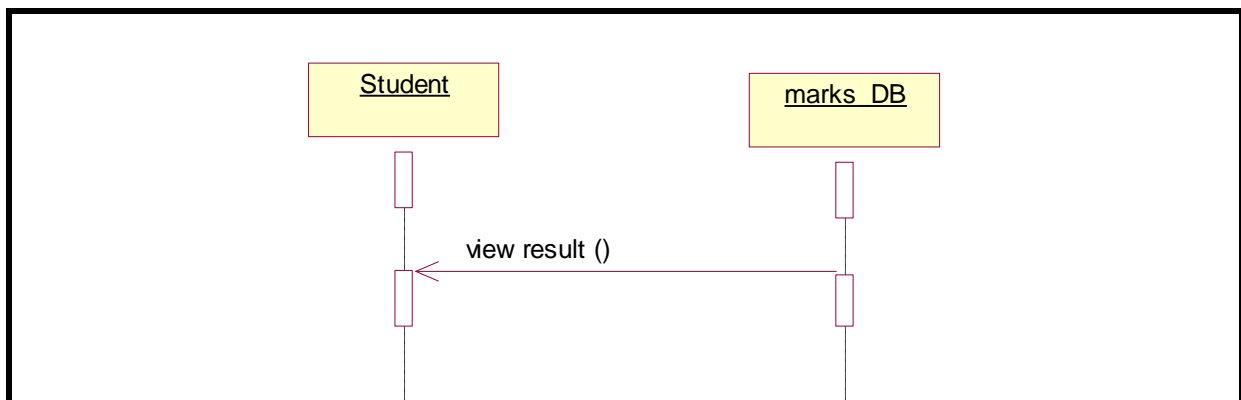
B-6 : Assign Mark Sequence Diagram
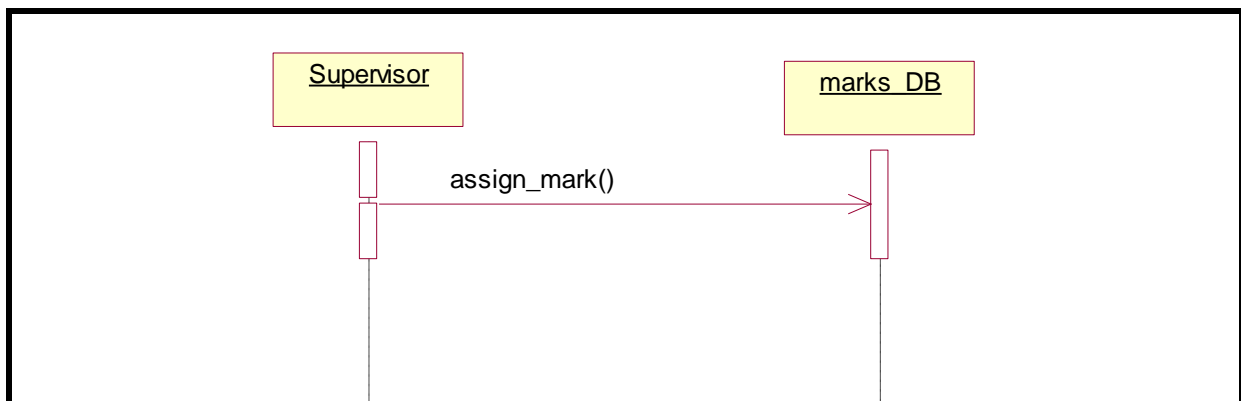
B-7 : Manage Schedule Sequence Diagram
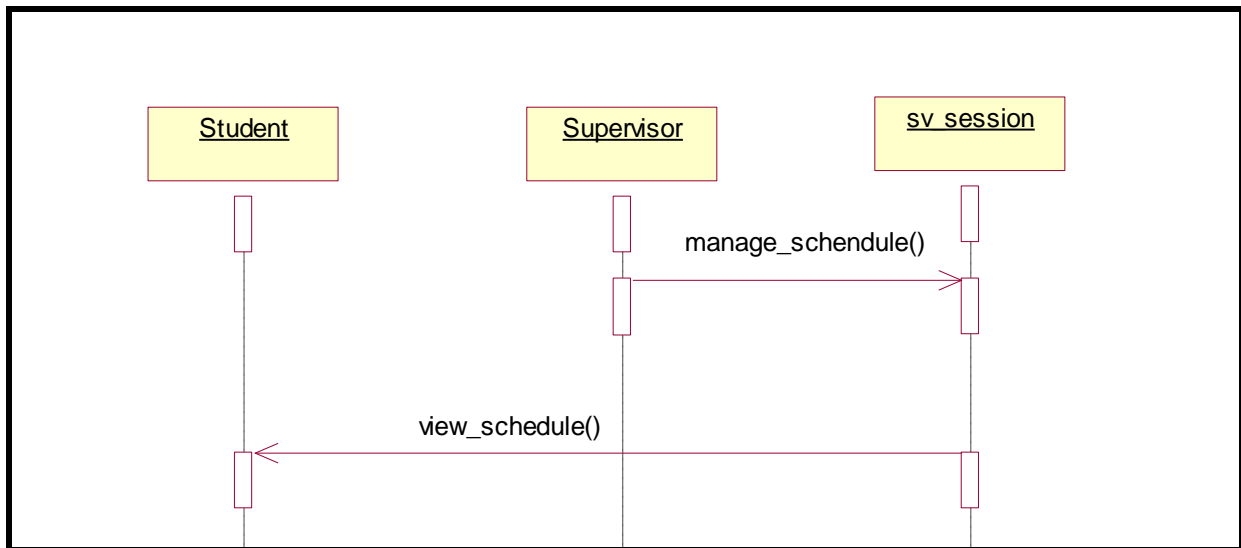
B-8 : Upload and Download Sequence Diagram
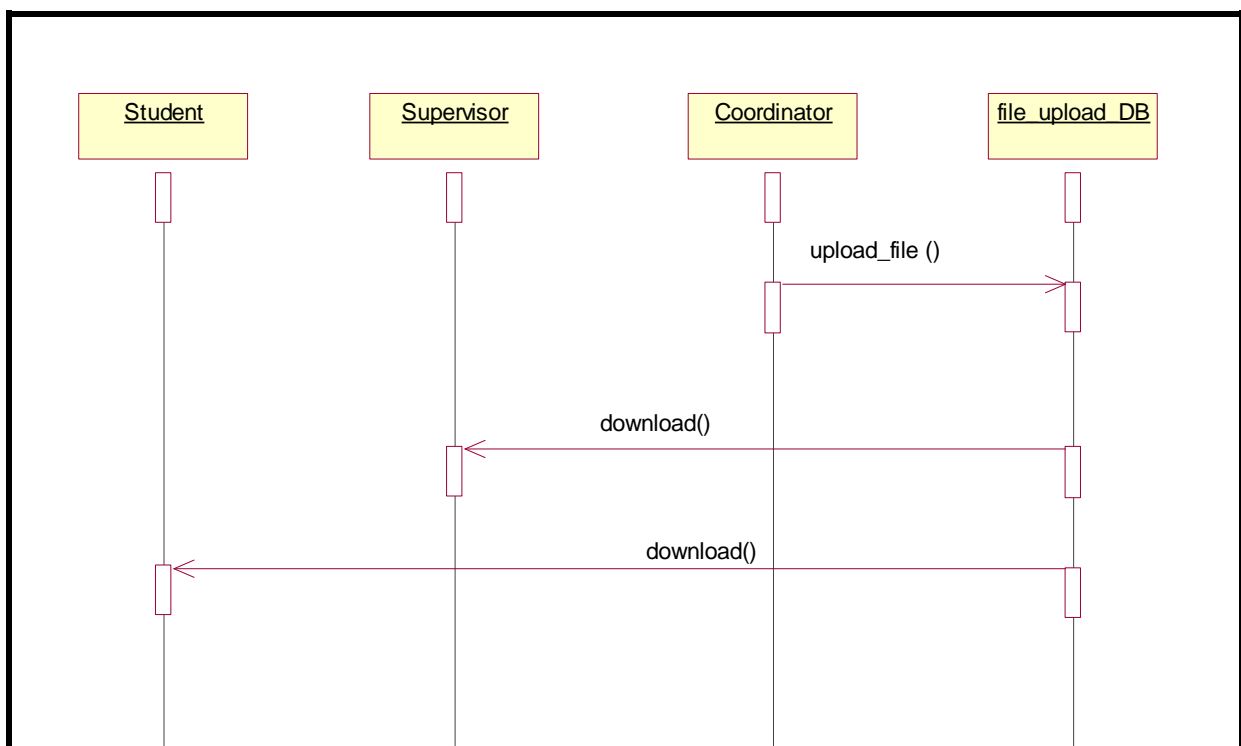
B-9 : Insert student Sequence Diagram
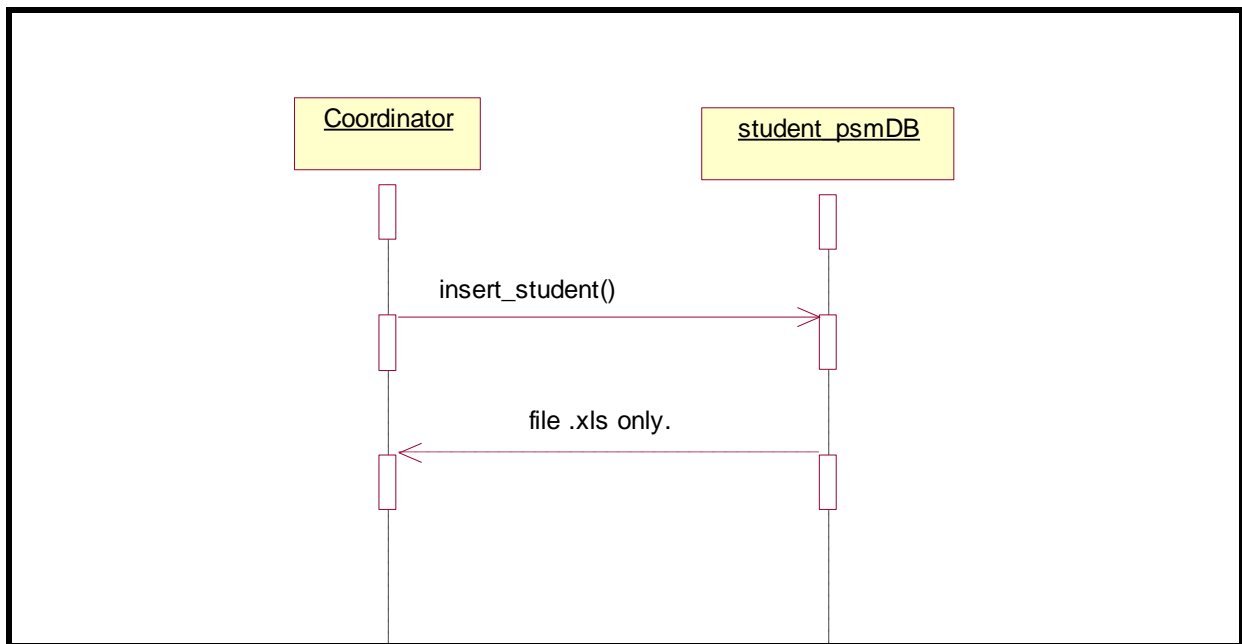
B-10 : Assign Evaluator Sequence Diagram
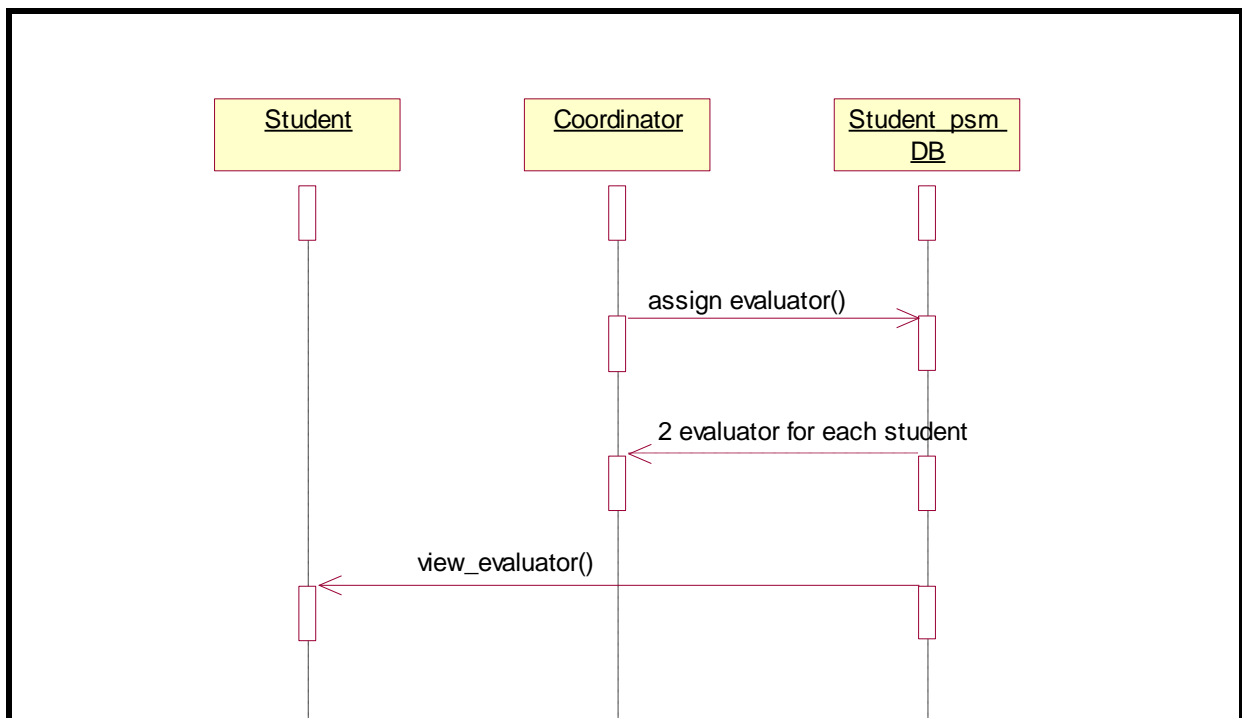
B-11 : Manage student Sequence Diagram
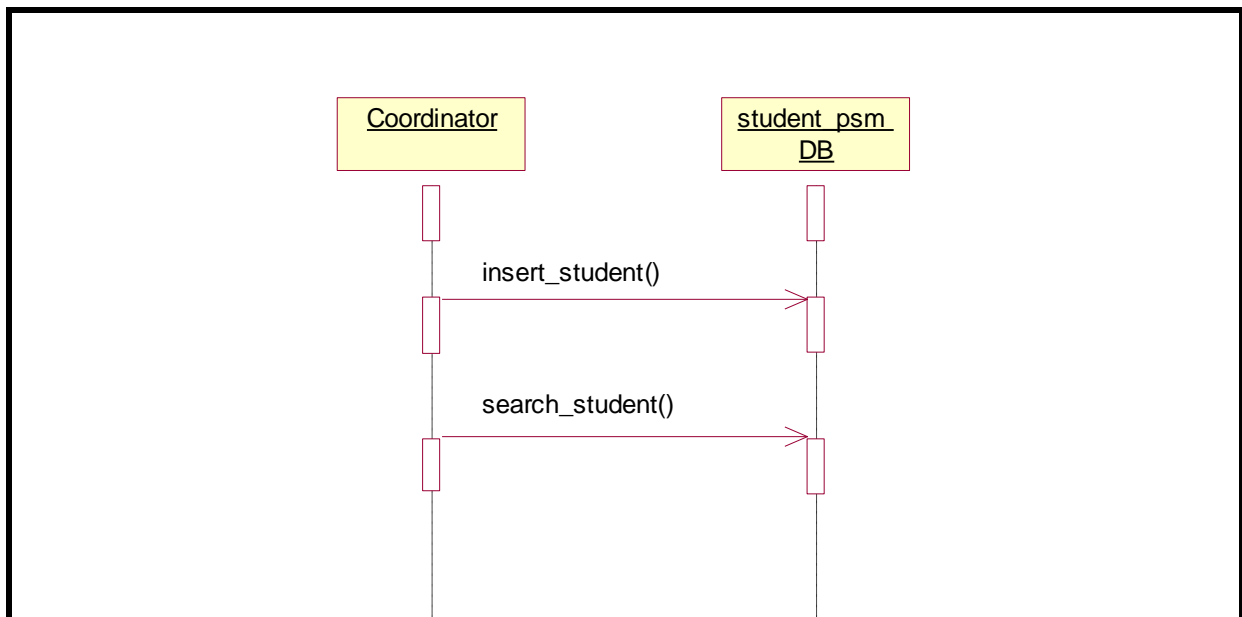
B-12 : Set timeline Sequence Diagram
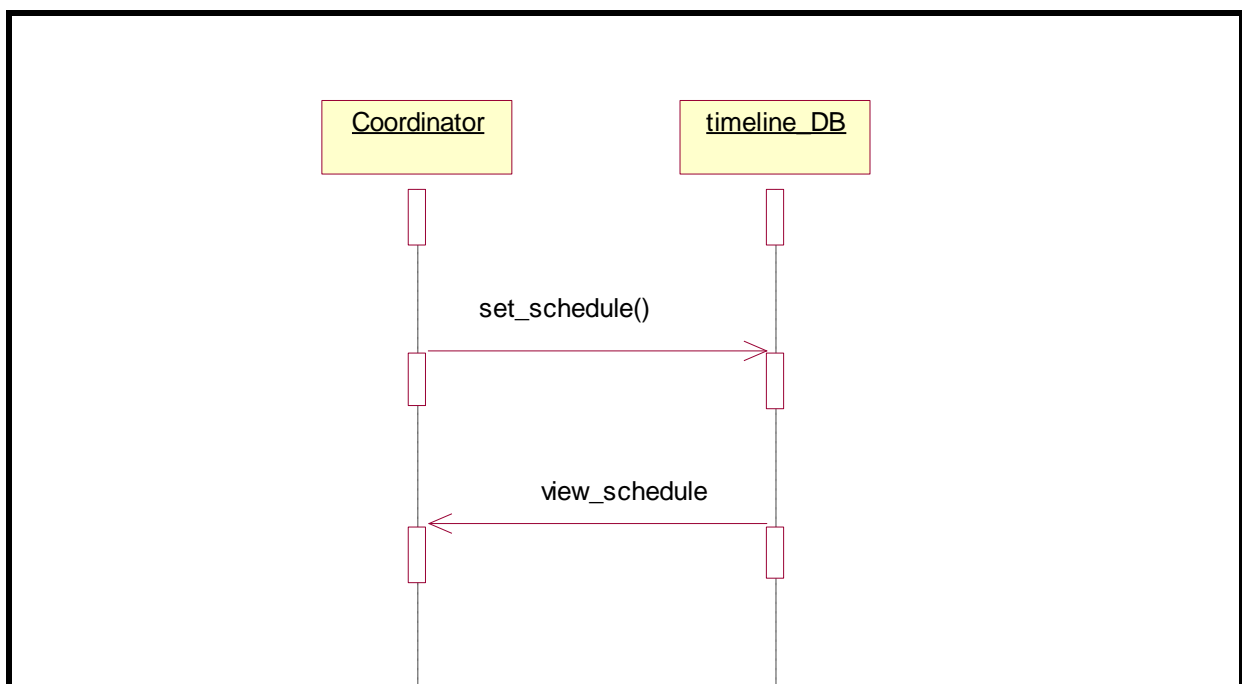
B-13 : Generate Record Sequence Diagram
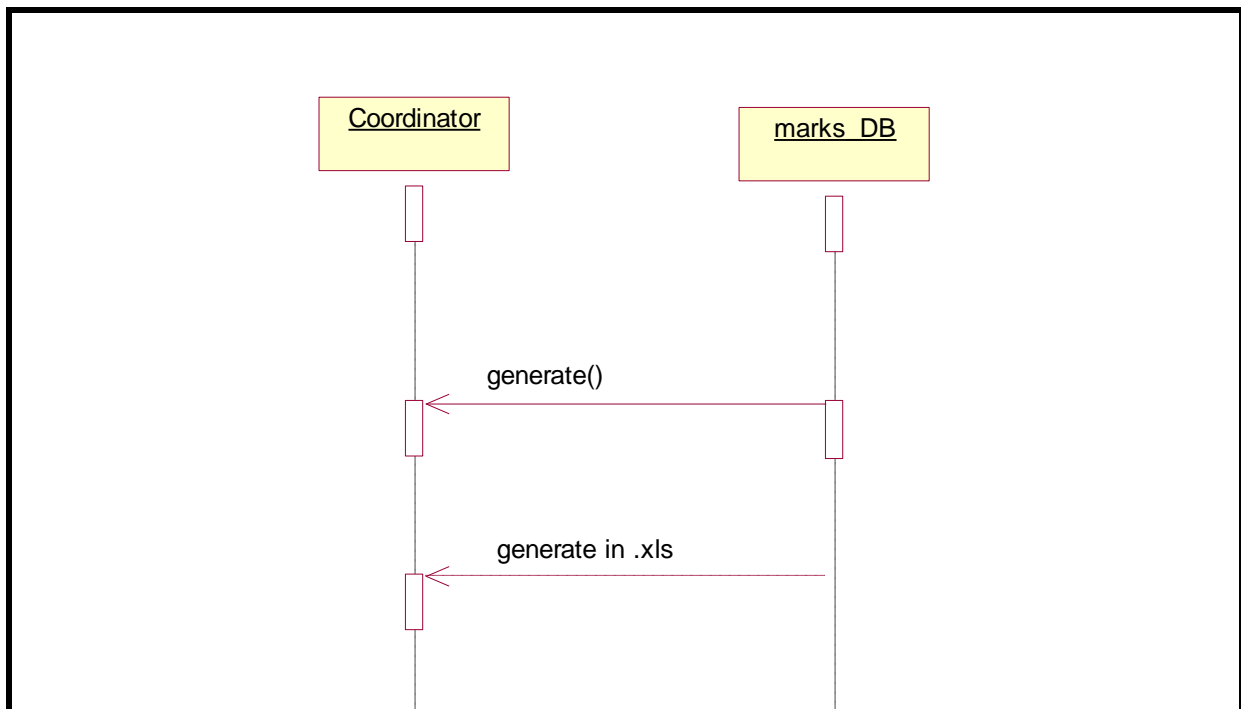
B-14 : Approve Sequence Diagram
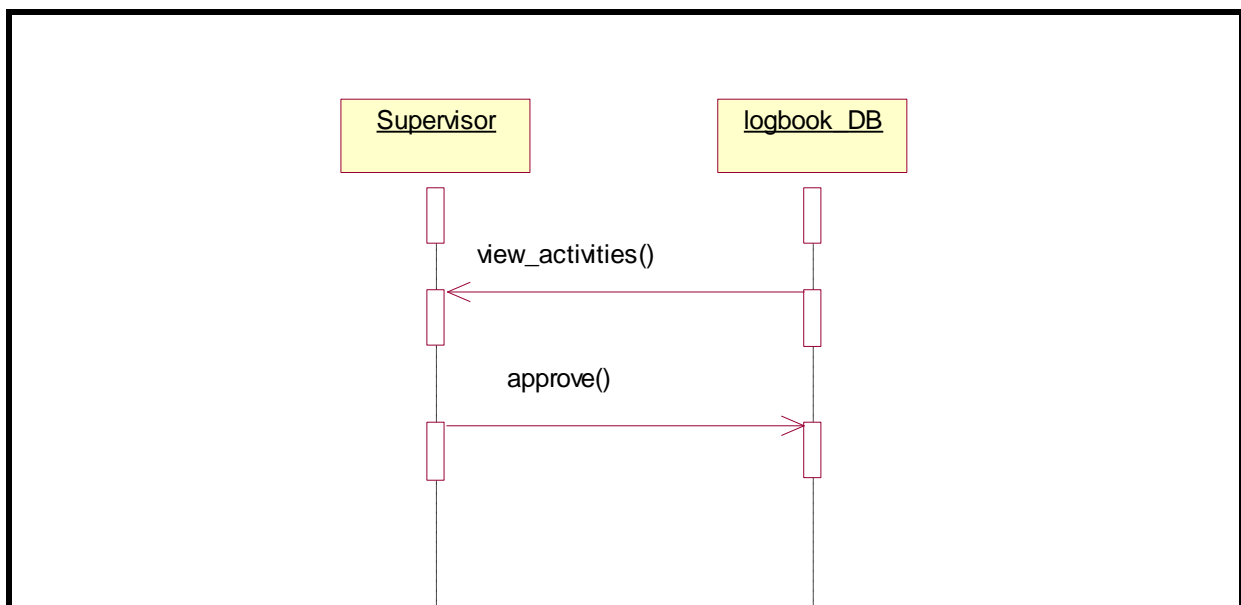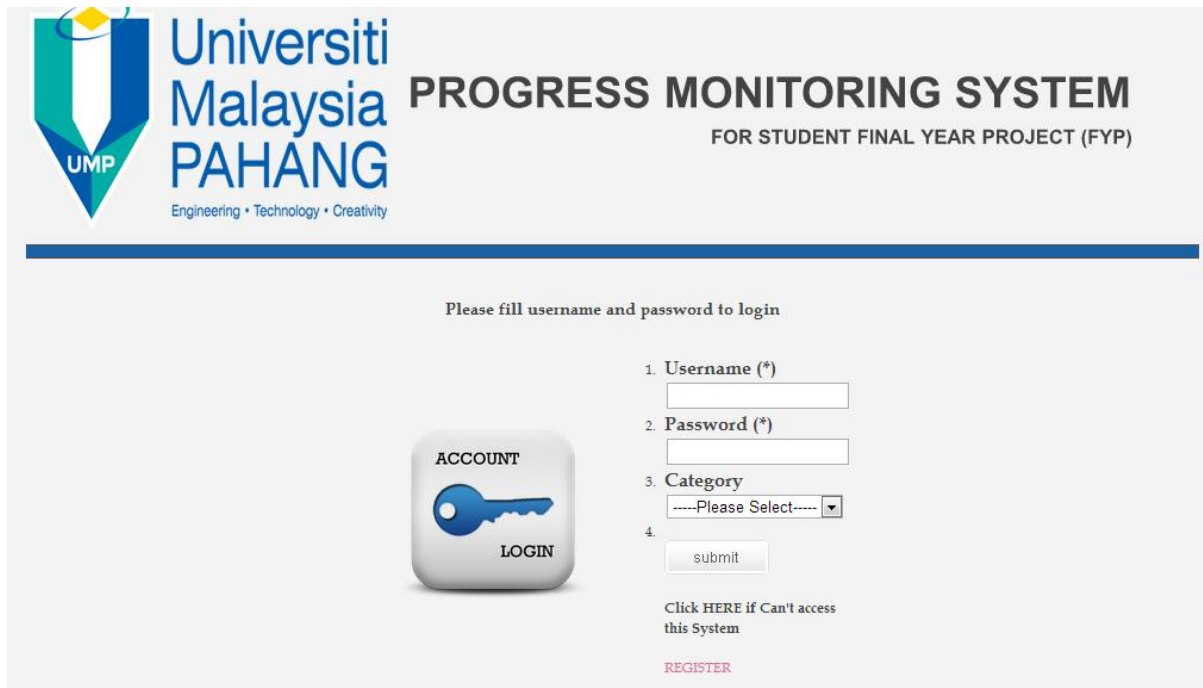
APPENDIX C

GRAPHICAL USER INTERFACE

C-1 : Login Interface

C-2 : Submit Activities Interface

Student ID	:	CB11008
Student Name	:	SITI NURHIDAYAH BT MOHAMAD KHAZALI
Project Name	:	PROGRESS
Meeting Date	:	[ ] *
Meeting Time (Start)	:	[ ] 🕐▼ *
Meeting Time (End)	:	[ ] 🕐▼ *
Week	:	--Please Select-- ▼ *
Progress	:	[ ] *

Add

LIST OF SUBMITTED ACTIVITIES			
Week	Date	Status	Action
6	08/11/13	Reject	Delete
7	14/11/13	Approve	Delete
11	14/11/13	Approve	Delete

Generate

C-3 : Progress Interface

C-4 : Feedback Interface

MATRIC ID	TITLE	VIEW	DELETE
cb11008	PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT	View	Delete

Student ID : cb11008

Student Name : SITI NURHIDAYAH BT MOHAMAD KHAZALI

Project Name : PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT

Supervisor : Azlina Binti Zainuddin

Category : PSM2

Type Report : -choose one-

Abstract	Part1 Introduction	Part2 Report Body	Part3 Conclusion
References	Appendices		

Abstract :

```
=============================
xcccccccccccccccccccccccccccccccccccccccccc
abstract
sddddddddddddddddddddddddddddddddddddddd
```

Feedback :

```
dsadsadsa
zxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

xzxhujan

mcvkmckvmcx
czxxxxxxxxxxxxxxxxxxxxx
DMSMDSADASDASKILLLLL
```

(Reset) (Submit)

Choose category to view file : -choose one-

Category :

List of file successfull uploaded.

C-5 : View Result Interface

NAME	TITLE	SUPERVISOR	PROGRESS 20%
SITI NURHIDAYAH BT MOHAMAD KHAZALI	PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT	AZLINA BINTI ZAINUDDIN	13.2

Generate

Universiti
Malaysia
PAHANG
Engineering • Technology • Creativity

Name   : SITI NURHIDAYAH BT MOHAMAD KHAZALI

Matric No   : CB11008

Category   : PSM2

Project Title : PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT

Supervisor   : AZLINA BINTI ZAINUDDIN

Evaluation 20%   : 13.2

C-6: Assign Mark Interface

Matric ID	Name	Supervisor Marks(20%)	Supervisor Marks(40%)	Total Marks
CB10083	MOHAMMAD REDZUAN BIN MOHD YUSOF	14.6	0	14.6
cv11111	ddddd	0	0	0

Matric ID being observed : -choose one-- ▾

STUDENT PROGRESS REPORT SUPERVISOR (20%)

CO	ITEM	WEIGHT %	SCORES (0-5)	MARK PERCENTAGE
CO1	Introduction/ Overview	0.4	--- ▾	
CO1	Problem Statements	0.6	--- ▾	
CO1	Objective	0.6	--- ▾	
CO1	Scope	0.6	--- ▾	
CO1	Compare system/method/ techniques/ algorithm	0.6	--- ▾	
CO1	Analysis on previous system/method/ highlight features/strength and weakness	0.6	--- ▾	
CO1	Explaination and discussion on existing system and limitation/ Application of proposed techniques of similar problem	0.6	--- ▾	
			Calculate	
			TOTAL	

Submit   Reset

C-7 : Manage Schedule Interface

ADD NEW SCHEDULE	
Supervisor	: AzlinaBintiZainuddin
Meeting Date	: [_____] *
Meeting Time (Start)	: [_____] 🕐▾ *
Meeting Time (End)	: [_____] 🕐▾ *
Week	: --Please Select-- ▾ *
Location	: [_____] *

Add

NO.	NO MATTRIC	DATE	WEEK	START_TIME	END_TIME	LOCATION	DELETE
1	cb11008	18/10/13	12	1:15 am	5:30 am	bilik saya	Delete
2	CB10083	02/12/13	8	1:30 am	3:45 am	fff	Delete

C - 8A : Upload

UPLOAD FILE

File name	:	
File category	:	-choose one--
Subject/Others	:	
Choose file	:	Choose File No file chosen
		Reset Upload

C - 8B : Download

DOWNLOAD FILE

Choose category to view file : Format Thesis

List of file successfull uploaded

Data uploaded: 2

No.	File Name	Category	Download	View
1	Format Thesis	Format Thesis	Download	View
2	format thesis	Format Thesis	Download	View

C-9 : Insert Student

Make sure the Excel file to be imported with this script, it is saved as a Excel 97-2003 (. XLS) first, because the PHP Excel Reader class can not work for the excel file format. XLSX (Excel 2007/2010). Make sure the header in your excel file are same as below:

A	B	C	D	E	F	G	H	I	J	K	L
matric_id	password	name	category	title	sv_name	ev1_name	ev2_name	date_p	strat_time_p	end_time_p	location_p
CB10011	CB10011	YEOW JIAN QIN	PSM1	Student Location Prediction Using Naïve Bayes Algorithm	MazlinaB intiAbdul Majid						

Select excel file to import (.xls): [Choose File] No file chosen   [Import]

C-10A : Assign Evaluator

PLEASE CHOOSE YOUR CATEGORIES AND SUPERVISOR

--Select Categories-- [▼]   -----------------------Please Select----------------------- [▼] [Search]

RESULTS

No.	Matric ID	Name	Category	Title	Supervisor	Evaluator 1	Evaluator 2	Start time	End Time	Location	Edit
1	CB09093	TRACEY CHIA	PSM2		AzlinaBintiZainuddin	ZaliliBintiMusa	BariahBintiYusob	2:30 pm	4:45 pm	k	🖉
2	CB10051	SHAHRUL RIDZUAN BIN ALIYAS	PSM2	Public Land Transport Schedule (Bus System) Using Mobile Application	AzlinaBintiZainuddin	ZaliliBintiMusa	BariahBintiYusob	2:45 am	2:30 am	k	🖉
3	CB10055	HIMALA DEWI A/P JAGANATHAN	PSM2	Estimation System Using Crude Function Points	AzlinaBintiZainuddin	ZaliliBintiMusa	BariahBintiYusob	5:45 pm		j	🖉
4	cb11008	SITI NURHIDAYAH BT MOHAMAD KHAZALI	PSM2	PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT	AzlinaBintiZainuddin	ZaliliBintiMusa	BariahBintiYusob	1:30 am	3:45 a	h	🖉

Page Number: 1

Total Records: 4

Searched For: PSM2

C-10B : Assign Evaluator

Matric ID	: CB10051
Name	: SHAHRUL RIDZUAN BIN ALIYAS
Title	: Public Land Transport Schedule (Bus System) Using Mobile Application
Supervisor	: AzlinaBintiZainuddin
First Evaluartor	: --------------------------Please Select------------------------ ▼ *
Second Evaluartor	: --------------------------Please Select------------------------ ▼ *
Date	: 03/10/13 *
Start Time	: 2:45 am ⊙▼ *
End Time	: 2:30 am ⊙▼ *
Location	: *
	submit

C-11 : Manage Student

C-12 A: Set Timeline

C-12 B: View Timeline

NO.	CLOSED DATE	WEEK	TASK	DELETE
1	06/12/13	14	hantar markah 20 %	Delete

C-13 : Generate Record

Generate Student Record By Department

C-14: Approve

MATRIC ID	NAME	TITLE	VIEW	DELETE
CB09093	TRACEY CHIA		View	Delete
CB10051	SHAHRUL RIDZUAN BIN ALIYAS	Public Land Transport Schedule (Bus System) Using Mobile Application	View	Delete
CB10055	HIMALA DEWI A/P JAGANATHAN	Estimation System Using Crude Function Points	View	Delete
cb11008	SITI NURHIDAYAH BT MOHAMAD KHAZALI	PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT	View	Delete

Matric ID: cb11008   Name: SITI NURHIDAYAH BT MOHAMAD KHAZALI   Title: PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT   Category: PSM2

Date	Week	Progress	Status	Action
08/11/13	6	sdsadsad	Reject	Submitted
14/11/13	7	hhhh	Approve	Submitted
14/11/13	11	rrrr	Approve	Submitted

APPENDIX D

SRS EVALUATION FORM

SRS CHECKLIST

Implementation Identification   :   _____

Document Name   :   _____

Evaluation Date   :   _____

Put (√) if quality metric has been fulfilled or (X) if not

Document Content Completion		
a) Chapter 1 and Sub-chapters		
b) Chapter 2 and Sub-chapters		
c) Chapter 3 and Sub-chapters		
d) Chapter 4 and Sub-chapter		
Standard Documentation Format		
Cover Page		
Table of Content		
Page Number		
Line Spacing		
a) Paragraph (1.0 Lines)		
b) Label for Tables and Figures		

Justification / Alignment		
Numbering and Indentation		
Font (Type & Size)		
Header and Footer		

Internal Consistency		
Contradictory statement in a document		
Only listed acronym & abbreviation used in document		
Understandability		
Right use of grammar, punctuation, symbols		
Standard Abbreviation		
All terms used in forms are defined		
Tables and figures are labeled properly		
Traceability To Other Documents		
Terms, acronym & abbreviation means the same thing in the existing documents		
Requirement contradictory		

Consistency With Other Documents		
Contradictory of items in a document		
Same acronyms & abbreviation used in existing documents		

Verified by   : _____

Post   : _____

Date Verified   : _____

Time   : _____

Venue   : _____

APPENDIX B

SDD

2013

SOFTWARE DESIGN DOCUMENT (SDD)

PROGRESS MONITORING SYSTEM FOR STUDENT FINAL YEAR PROJECT (FYP)

SITI NURHIDAYAH BT MOHAMAD KHAZALI
CB11008
900821035942
To be submitted to the Undergraduate Project II
Bachelor of Computer Science (Software Engineering)

DOCUMENT APPROVAL

	Name	**Date**
Verified by: _____ Project Manager	Siti NurHidayah Bt Mohamad Khazali	
Authenticated by: _____ Supervisor	Miss Azlina Binti Zainuddin	
Authenticated by: _____ Client	Coordinator Undergraduate Project	

Software   : IBM Rational Software Architecture (RSA), Microsoft Office 2007

Archiving Place   : D:\PMS\DOCUMENTATION\SRS\

Copies Available   : doc, docx,pdf

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDIXES

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to provide a complete description of the system's requirements into a description of the software structure, software components, interfaces, and data necessary for implementation.

1.2 System Identification

System Number : SDD-REQ-PMS-2013-01-V1

System Name : Progress Monitoring System for Student Final Year Project

System Abbreviation : PMS

1.3 System Overview

The Progress Monitoring System for Student Final Year Project (PSM) is computer software which helps student contacts of University Malaysia Pahang (UMP) under Faculty System Computer and Software Engineering (FSKKP) gain better control of their project planning and implementation through keeping student connected with supervisor, regardless of where student are located. Supervisor in FSKKP can monitor every detail, simply, and easily. However the system is about managing projects from remote destinations. So that, this system helps student to complete projects, keep within budget, stay on track, and collaborate with supervisor. This system is able to register user, give feedback, upload and download, assign mark, approve activity and generate report.

1.4 Module Overview

1.4.1 **Login Module** (SRS-REQ-PMS-2013-1-00)

Users are required to login to the system before they can access the other modules or functions as single user has their own unique permissions and access. To login to the system, user must key in username and password, as well as selecting the user category.

1.4.2 Logbook Module (SRS-REQ-PMS-2013-2-00)

- Students can access this module to view a student's personel and FYP details, however the scope of student they can view differs according to their permissions.

- Supervisor can access this module to view student's logbook.

- Student will submit their activities to supervisor and to get approval.

1.4.3 Progress Module (SRS-REQ-PMS-2013-3-00)

- This module discuss about student is able to submit their progress.

- Student must submit their project progress by weekly.

1.4.4 Feedback Module (SRS-REQ-PMS-2013-4-00)

- In this module, student will get feedback from supervisor when submit their progress.

1.4.5   Result Module (SRS-REQ-PMS-2013-5-00)

- In this module, student is able to view their result after they are completed and submit their project progress.

1.4.6   Evaluate Module (SRS-REQ-PMS-2013-6-00)

- In this module, lecturer who play as supervisor and evaluator will assign marks to student based on the rubric .

1.4.7   Download Module (SRS-REQ-PMS-2013-8-00)

- In this module, supervisor and student able to download the uploaded files by coordinator

1.4.8   Approve Module (SRS-REQ-PMS-2013-14-00)

- In this module, supervisor is able to Approve / Resubmit activities submitted by their students.

1.4.9   Manage (SRS-REQ-PMS-2013-7-00)

- In this module, supervisor is able to set the time and arrange it for meet their student.

- Supervisor will able to generate records into excel in .xls format.

1.4.10   Student (SRS-REQ-PMS-2013-11-00)

- In this module, coordinator have authority in manage database.

- Coordinator is able to add, delete and search student information.

1.4.11 Evaluator (SRS-REQ-PMS-2013-10-00)

- In this module, coordinator is able to choose two lecturer to be the evaluator of a student.

- Coordinator is able to add new supervisor.

1.4.12 Timeline (SRS-REQ-PMS-2013-12-00)

- In this module, coordinator is able to set the project timeline.

1.4.13 Upload (SRS-REQ-PMS-2013-8-00)

- In this module, coordinator is able to upload the format of technical report, thesis, rubric and others.

1.4.13 Report (SRS-REQ-PMS-2013-13-00)

- In this module, coordinator is able to generate all student records in excel format.

1.5 Document Overview

This document is divided into six sections:

Section 1   **Introduction**

This section describes about the purpose of the document and its identification, the system overview and modules, and also the overview of the document.

Section 2   **Referenced Documents**

This section list all the references used in completing the document.

Section 3   **Architecture Description**

This section gives detailed descriptions about the system architecture that includes application, business services and middleware and the definition. This section also provides the packaging of system modules and data dictionary.

Section 4   **Detailed Designs**

This section gives the detailed explanations about design of the system.

Section 5   **Requirement Traceability**

This section provides the requirement traceability of the requirement of the SRS

Section 6   **Notes**

This section list the acronyms and abbreviations used in the document to be used as a reference. It also includes the appendices.

2. REFERENCED DOCUMENT

2.1 Government Document

Not applicable

2.2 Non - Government Document

Table 2.1 : Non - Government Document

Guideline	Source
[1] Guideline SDP-TMS-2010-V1	Software Development Pelan Document for TMS.
[2] Guideline SRS-TMS-2010-V1	Software Requirement Specification Document for TMS
[3] Guideline Software Engineering	Addison Wesley, "Software Engineering" 8th Edition, Ian Sommerville, England, 2006
[4] Guideline IEEE	IEEE Recommended Practice for Software Design Descriptions IEEE Std 1016-1998

3.   SPECIFIC REQUIREMENTS

3.1   General Architecture

In PMS, a three - tier architecture s used. The interface is located at the Application Layer   (Presentation Tier), the logic or control of the application is located at the Business Service Layer (Business Tier), and the database at the Middleware layer (Data Tier).
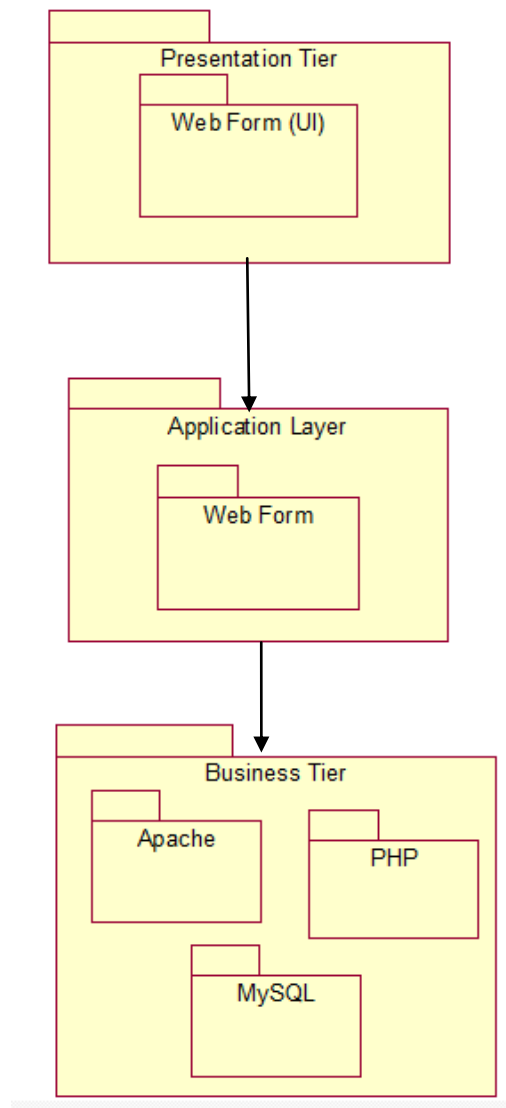
Figure 3.1 : General Architecture

3.1.1   Layer Architecture

3.1.1.1 Application Layer

The application layer consists of the interfaces, representing the design elements. The interfaces enables interaction between users and the systems. The interfaces consists of each type of users (Coordinator, Supervisor and Student), and home page. The Application Layer consists of application-specific design elements which are Web Form (UI) package are called Boundary Package.

Package Diagram
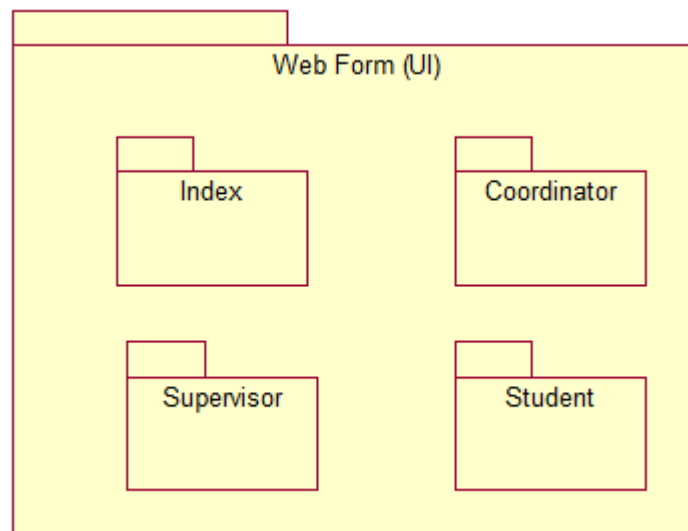
Figure 3.2 : Boundary Package

3.1.1.2 Business Service

The Business Layer is the layer that controls the actions in the application layer. The Business Layer controls the applications functionality by performing detail processing. The Business Services layer consist business-specific elements that are used in several applications. There are Web Form package which called as Control Package and Data Store (DB class) package which called as Entity Package.

Package Diagram
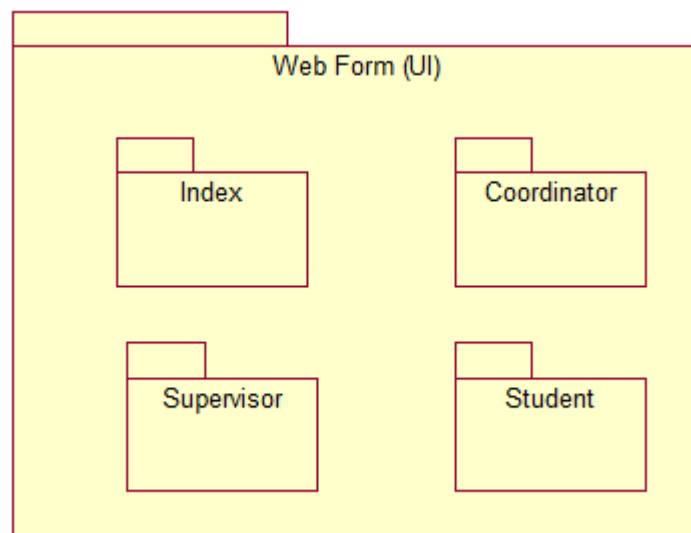
Figure 3.3 : Web Form package

3.1.1.3 Middleware

The Middleware consist of Database Sequel Server or script that connecting the IIS with the database. For the connection for connect between SQL Server and PMS inside IIS server which are C# and ASP.Net code. This tier consists of database server, web server and provides services for enable communication and management of data between system and database server.
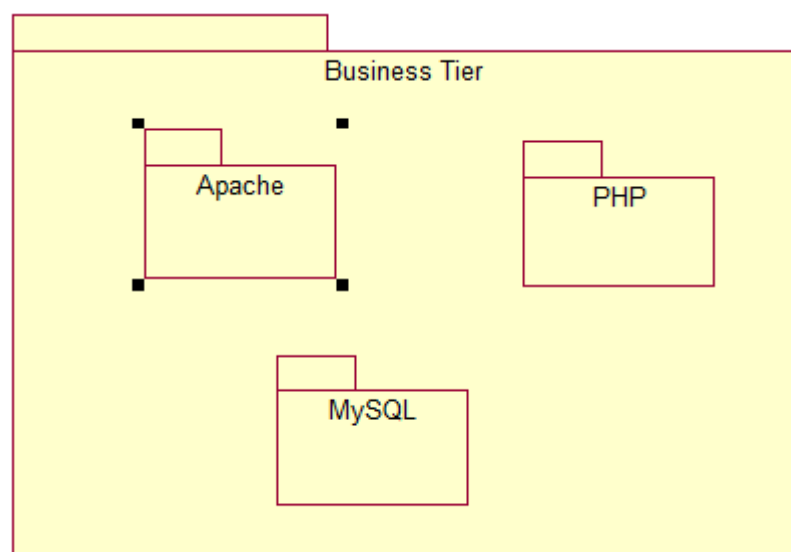
Figure 3.4 : Middleware Layer

3.2 Module Package

3.2.1 Class Diagram for Index Interface Package

Figure 3.5 : Class Diagram for Index Interface Package
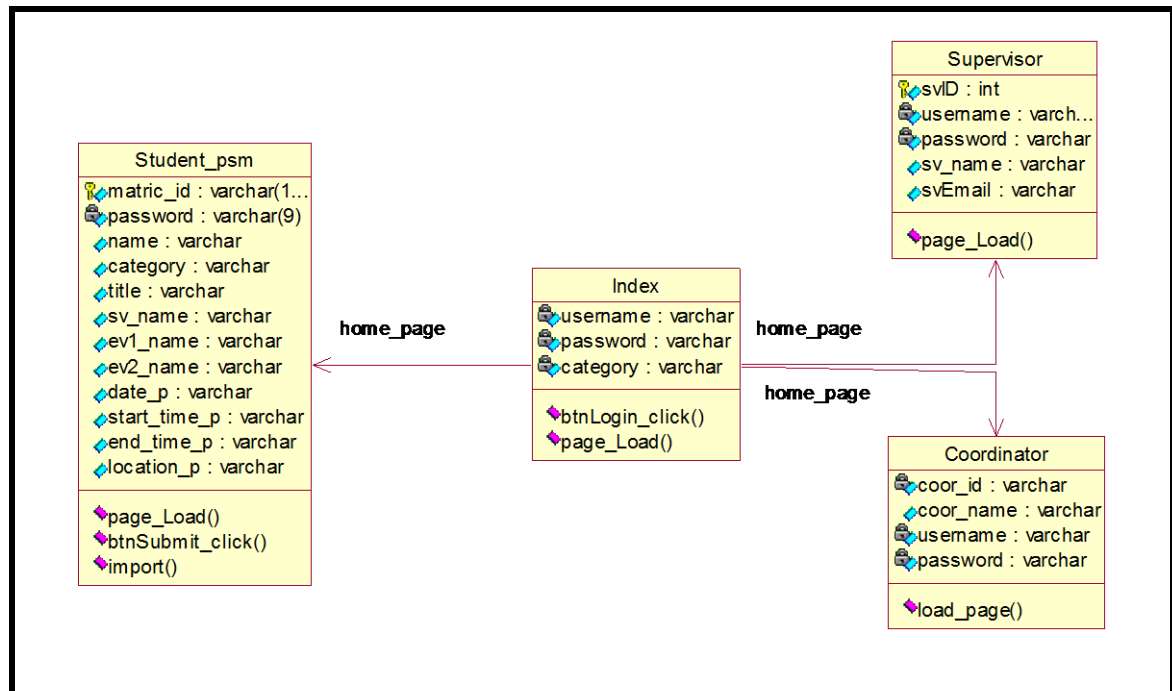
Figure 3.5 : Class Diagram for Index Interface Package

3.2.2   Class Diagram for Student Interface Package
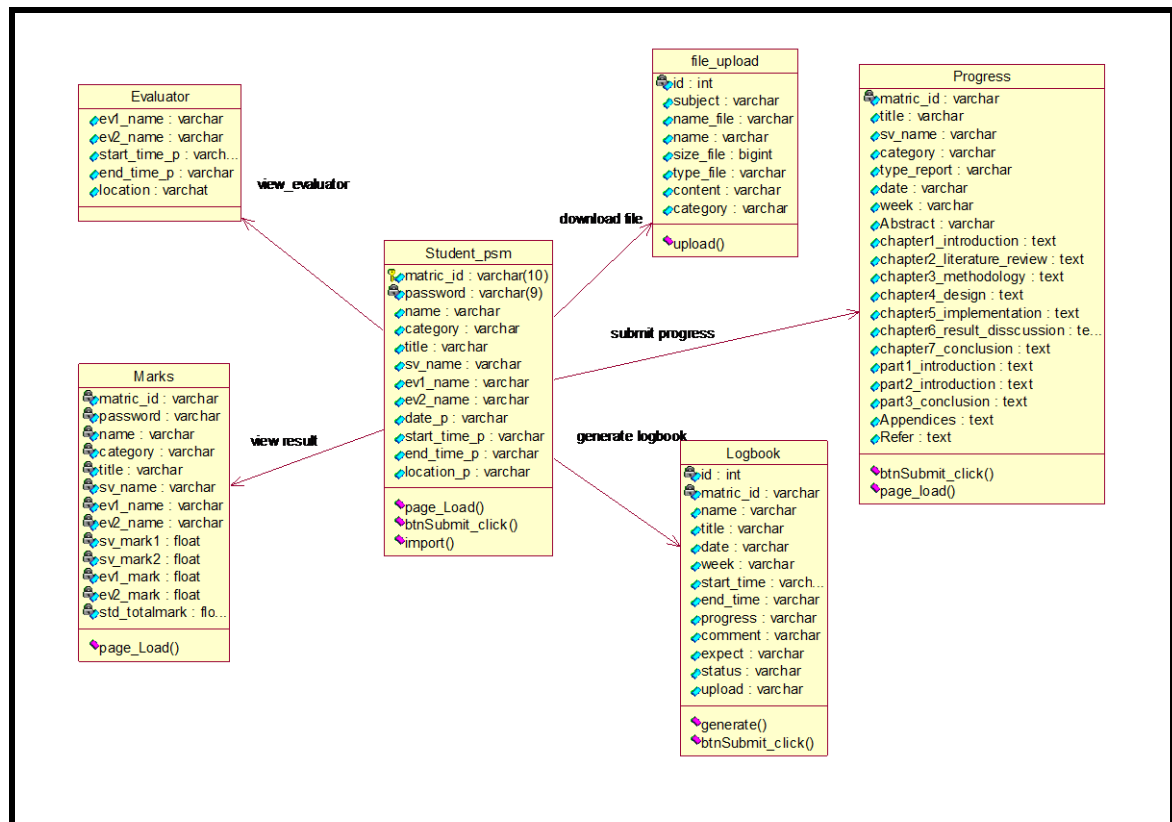
Figure 3.6 : Class Diagram for Student Interface Package

3.2.3 Class Diagram for Student Interface Package
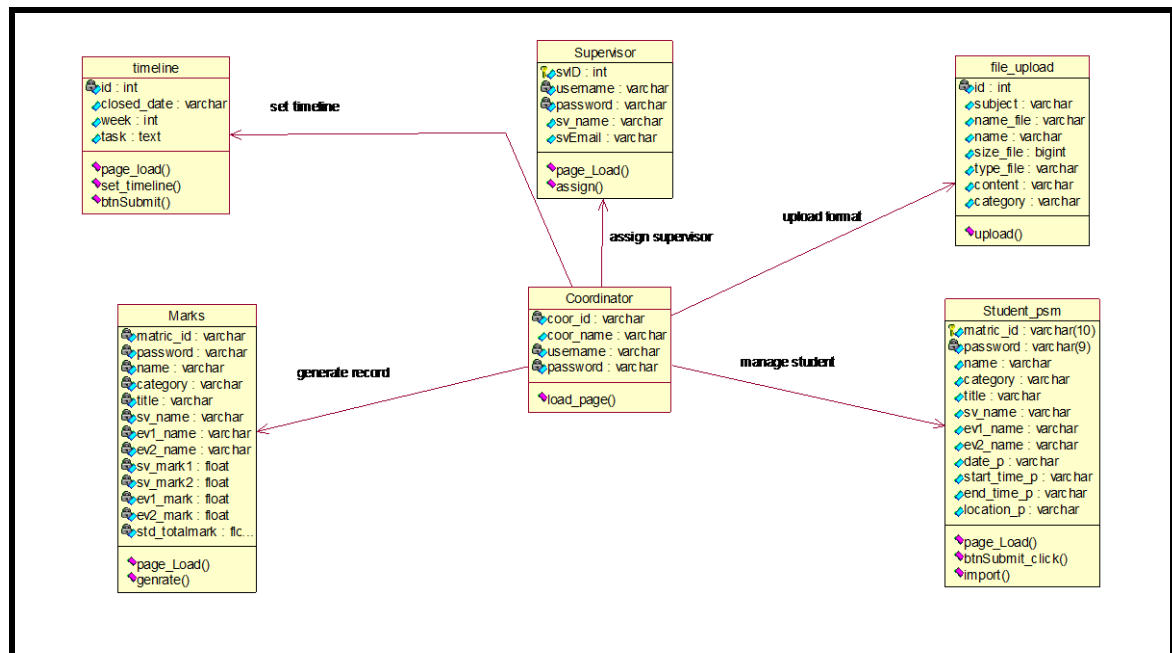
Figure 3.7 : Class Diagram for Coordinator Interface Package

3.2.4 Class Diagram for Supervisor Interface Package
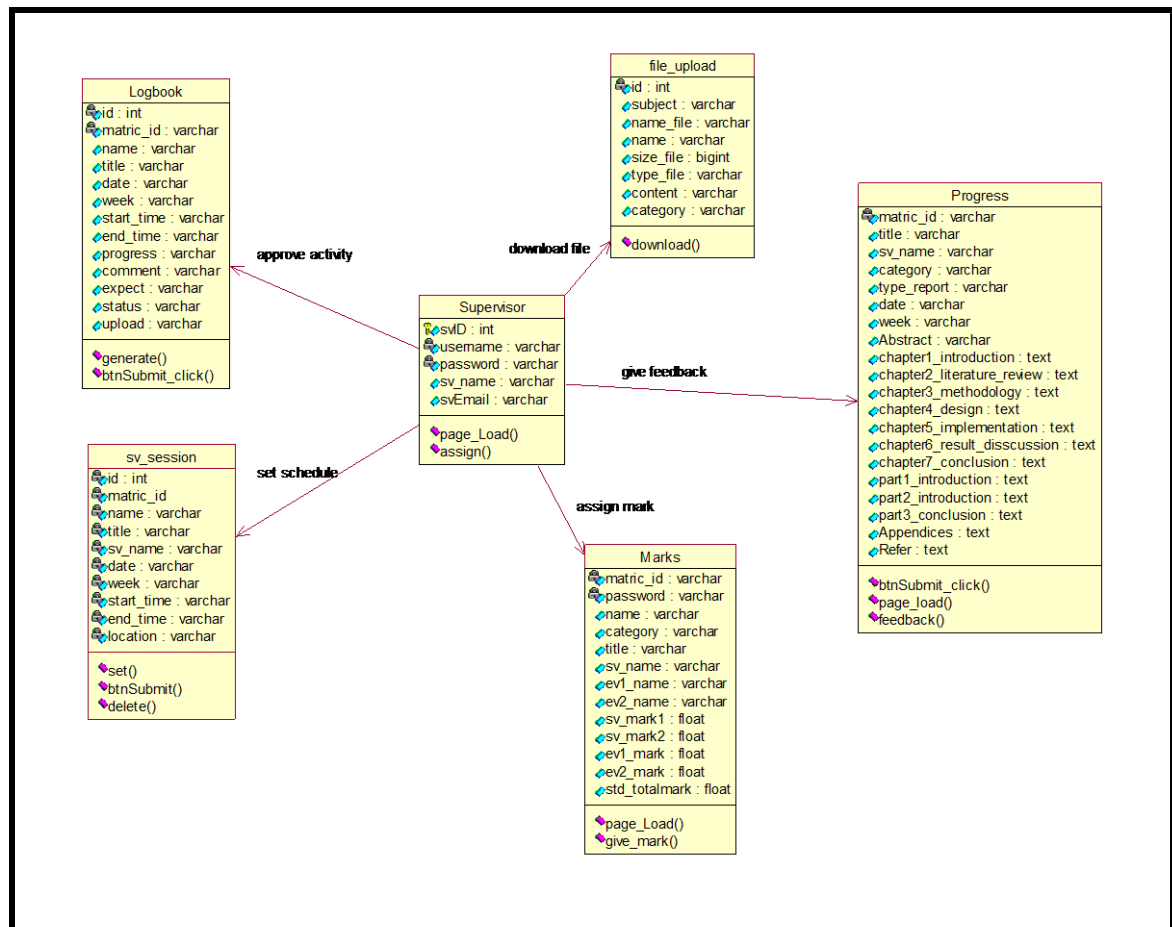
Figure 3.8 : Class Diagram for Supervisor Interface Package

3.3   Data Dictionary

Data dictionary is the detailed data model of a database. The logical data model contains all needed logical and physical designs. Physical storage parameters are needed to generate a design in a Data Definition Language, which can be used to create database.

Database Name : eprogress.sql

It has four tables that are item, denda, pinjaman and user.

3.3.1   student_psm (the details of the student)

Table 3.1 : Data Dictionary for Student

Attributes	Description	Data type	Constraint
Matric_id	Defines student Matric no.	Varchar(7)	PK
Password	Define student password	Varchar(9)	
Name	Defines student name	Varchar(200)	
Category	Student's category	Varchar(10)	
Title	Student's title	Text	
Sv_name	Defines student supervisor	Varchar(50)	
Ev1_name	Defines student first evaluator	Varchar(50)	
Ev2_name	Define student second evaluator	Varchar(50)	
Date_p	Define Date to present	Varchar(10)	
Start_time_p	Define time to present	Varchar(10)	
Location_p	Define Location	Varchar(50)	

3.3.2 supervisor (The details of the Supervisor)

Table 3.2 : Data Dictionary for Supervisor

Attributes	Description	Data type	Constraint
svID	Defines supervisor ID	Int(5)	PK
Username	Define supervisor username	Varchar(20)	
Password	Defines supervisor password	Varchar(10)	
Sv_name	Defines supervisor password	Varchar(50)	
svEmail	Define Supervisoremail	Varchar(50)	

3.3.3 coordinator (The details of the Coordinator)

Table 3.3 : Data Dictionary for Coordinator

Attributes	Description	Data type	Constraint
Coor_id	Defines coordinator ID	Int(5)	PK
Coor_name	Define coordinator name	Text	
Username	Defines coordinator username	Varchar(20)	
Password	Defines coordinator password	Varchar(20)	

3.3.4   Mark (The details of the Mark)

Table 3.4 : Data Dictionary for Marks

Attributes	Description	Data type	Constraint
Matric_id	Defines Progress id	Varchar(7)	FK
Password	Defines Student Password	Varchar(10)	
Name	Defines student name	Varchar(150)	
Category	Defines student category	Varchar(4)	
Title	Defines Student title	text	
Sv_name	Defines student name	Varchar(50)	
Ev1_name	Defines student first evaluator	Varchar(50)	
Ev2_name	Defines student second evaluator	Varchar(50)	
Sv_mark1	Defines first evaluation mark from supervisor	Float	
Sv_mark2	Defines second evaluation mark from supervisor	Float	
ev1_mark	Defines evaluator mark	Float	
ev2_mark	Defines evaluator mark	Float	
std_totalmark	Define student total marks	Float	

3.3.5   progress (the details of Progress)

Table 3.5 : Data Dictionary for Progress

Attributes	Description	Data type	Constraint
Matric_id	Define student matric number	Varchar(7)	FK
Title	Defines project title	Varchar(300)	
Sv_name	Define Supervisor	Varchar(150)	
Category	Defines category of progress	Varchar(4)	
Type_report	Defines type of report	Varchar(30)	
Date	Defines date submission	Varchar(10)	
Week	Defines week submission	Int(2)	
Abstract	Defines Abstract	Text	
Chapter1_introduction	Defines introduction of thesis	Text	
Chapter2_literature_review	Defines literature review of thesis	Text	
Chapter3_methodology	Defines methodology of thesis	Text	
Chapter4_design	Defines design of thesis	Text	
Chapter5_implementation	Defines implementation of thesis	Text	
Chapter6_result_discussion	Defines result and discussion of thesis	Text	
Chapter7_conclusion	Defines conclusion of thesis	Text	
Part1_introduction	Defines introduction of technical report	Text	
Part2_report_body	Defines report body of technical report	Text	

Part3_conclusion	Defines conclusions of technical report	Text	
Appendices	Defines appendices of both report	Text	
Refer	Defines references of both report	Text	

3.3.6 file_upload (The details of Upload file from Coordinator)

Table 3.6 : Data Dictionary for Upload File

Attributes	Description	Data type	Constraint
<u>Id</u>	Defines ID	Int(6)	PK
Subject	Define subject	Varchar (30)	
Name_file	Define file name	Varchar(30)	
Name	Define name for the file	Varchar(100)	
Size_file	Defines size of file	Bgint (50)	
Type_file	Defines type of file	Varchar (30)	
Content	Defines Content	Longblob	
Category	Defines Category	Varchar (30)	

3.3.7 report (The details of Upload file from Student)

Table 3.7 : Data Dictionary for Report

Attributes	Description	Data type	Constraint
<u>Id</u>	Defines ID	Int(7)	PK
Matric_id	Define student matric	Varchar (7)	FK
Name_student	Defines student name	Varchar(100)	
Title	Defines student tile	Text	

Sv_name	Defines supervisor name	Varchar(100)	
Subject	Defines subject	Varchar(70)	
Week	Defines week to submit	Int(2)	
Date	Defines date to submit	Varchar(10)	
Name_file	Defines file name	Varchar(30)	
Name	Defines file name	Varchar(100)	
Size_file	Defines size	Bgint(50)	
Type_file	Defines type of file	Varchar(100)	
Content	Defines Content	Longlob	
Category	Defines Category	Varchar(30)	

3.3.8   timeline (The details of Coordinator task)

Table 3.8 : Data Dictionary for Task

Attributes	Description	Data type	Constraint
Id	Defines ID	Int(5)	PK
Closed_date	Defines close date	Varchar(10)	
Week	defines week	Int(2)	
Task	Defines Coordinator task	Text	

3.3.9   feedback (The details of feedback)

Table 3.9 : Data Dictionary for Feedback

Attributes	Description	Data type	Constraint
Matric_id	Defines student matric	Varchar(7)	FK
Name	Defines student name	Varchar(100)	
Title	Defines student title	Text	
Sv_name	Defines supervisor name	Varchar(100)	
Feedback1	Defines feedback 1	Text	
Feedback 2	Defines feedback 2	Text	
Feedback 3	Defines feedback 3	Text	
Feedback 4	Defines feedback 4	Text	
Feedback 5	Defines feedback 5	Text	
Feedback 6	Defines feedback 6	Text	
Feedback 7	Defines feedback 7	Text	
Feedback 8	Defines feedback 7	Text	
Feedback 9	Defines feedback 9	Text	
Feedback 10	Defines feedback 10	Text	
Feedback 11	Defines feedback 11	Text	
Feedback 12	Defines feedback 12	Text	
Feedback 13	Defines feedback 13	Text	

4. DETAILED DESIGN

4.1 Coordinator Package
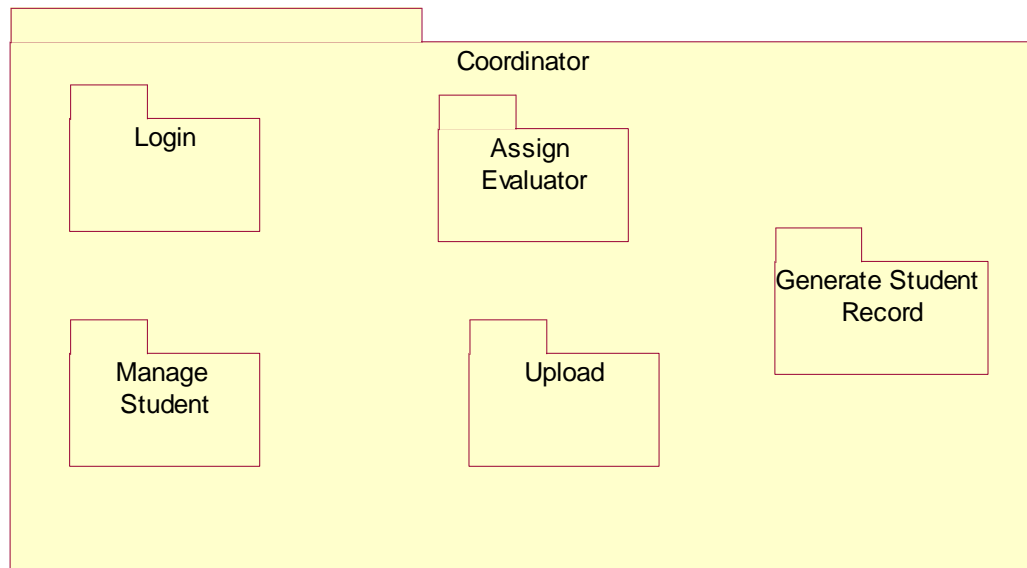
Figure 4.1 : Coordinator Module

4.1.1 Login Module

Table 4.1: Login Module

Responsibility	:	Allows coordinator to login		
Attributes	:	Username	:	Varchar
		Password	:	Varchar
		Category	:	Varchar
Methods	:	btnLogin_click	:	Verifies Username and password and creates session
		Page_load	:	Loads Coordinator home page

4.1.2 Manage Student Module

Table 4.2: Manage Student Module

Responsibility	:	Display student details		
Attributes	:	Matric_id	:	Varchar
		Password	:	Varchar
		Name	:	Varchar
		Category	:	Varchar
		title	:	text
		Sv_name	:	Varchar
		Ev1_name	:	Varchar
		Ev2_name	:	Varchar
		Date_p	:	Varchar
		Start_time_p	:	Varchar
		End_time_p	:	Varchar
		Location_p	:	Varchar
Methods	:	Page_load	:	Loads student details
	:	btnDelete_Click ( )	:	Delete Student
	:	btnAdd _click( )	:	Add New Student
	:	btnUpdate_click ( )	:	Update Student
	:	btnSearch_clik()	:	Search student

4.1.3　Assign Evaluator Module

Table 4.3: Assign Evaluator Module

Responsibility	:	Assign Evaluator details		
Attributes	:	Matric_id	:	Varchar
		Password	:	Varchar
		Name	:	Varchar
		Category	:	Varchar
		title	:	text
		Sv_name	:	Varchar
		Ev1_name	:	Varchar
		Ev2_name	:	Varchar
		Date_p	:	Varchar
		Start_time_p	:	Varchar
		End_time_p	:	Varchar
		Location_p	:	Varchar
Methods	:	Page_load	:	Loads student details
	:	delete_click ( )	:	Delete Student
	:	BtnAdd_click ( )	:	Add Student
		btnUpdate_click ( )		Update Student
	:	btnUpdateEvaluator_click ( )	:	Update Evaluator

4.1.4 Generate Student Record Module

Table 4.4: Assign Mark Module

Responsibility	:	Allows coordinator to generate report based on student marks.		
Attributes	:	Matric_id	:	Varchar
		Password	:	Varchar
		Name	:	Varchar
		Category	:	Varchar
		Title	:	Text
		Sv_name	:	Varchar
		Ev1_name	:	Vachar
		Ev2_name	:	Varchar
		Sv_mark1	:	Float
		Sv_mark2	:	Float
		Ev1_mark	:	Float
		Ev2_mark	:	Float
		Std_totalmark	:	Float
Methods	:	btnSubmit_click	:	Generate report in .xls file
		Page_load	:	Loads Coordinator home page

4.2.5   Upload Module

Table 4.5: Upload Module

Responsibility	:	Allows coordinator to upload related file		
Attributes	:	Id	:	Int
		Subject	:	Varchar
		Name_file	:	Varchar
		Name		Varchar
		Size_file		Varchar
		Type_file		Varchar
		Content		Longlob
		Category		varchar
Methods	:	btnUpload_click()	:	Upload file in database
		Page_load	:	Loads Coordinator upload page

4.2   Supervisor Package
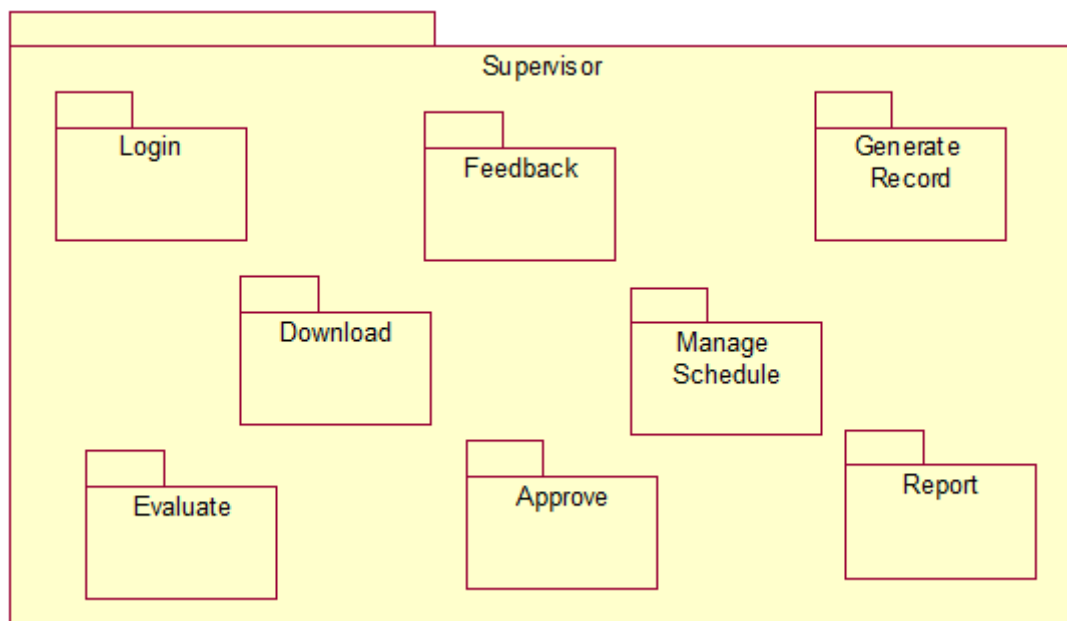
Figure 4.2 : Supervisor Module

4.2.1   Login Module

Table 4.6: Login Module

Responsibility	:	Allows coordinator to login		
Attributes	:	Username	:	Varchar
		Password	:	Varchar
		Category	:	Varchar
Methods	:	btnLogin_click	:	Verifies Username and password and creates session
		Page_load	:	Loads Supervisor home page. Supervisor able to choose project category for evaluate FYP project.

4.2.2   Approve Activity Module

Table 4.7: Approve Acitivity Module

Responsibility	:	Allows coordinator to login		
Attributes	:	Id	:	Varchar
		Matric_id	:	Varchar
		Name	:	Varchar
		Title		Title
		Date		Varchar
		Week		Int(2)
		Start_time		Varchar
		End_time		Varchar
		Progress		Varchar

		Comment		Varchar
		Expect		Varchar
		Status		Varchar
		upload		Varchar
Methods	:	View_click()	:	View student logbook
	:	Action_click()	:	Approve student activity
	:	btnSubmit_click()	:	Insert Comment in database

4.2.3 Evaluate Module

Table 4.8: Evaluate Module

Responsibility	:	Allows Supervisor to evaluate marks based on rubric		
Attributes	:	Matric_id	:	Varchar
		password	:	Varchar
		Name	:	Varchar
		category		varchar
		title		Varchar
		sv_name		varchar
		ev1_name		Varchar
		ev2_name		Varchar
		sv_name		Varchar
		ev1_name		Varchar
		ev2_name		Varchar
		sv_mark1		float
		sv_mark2		float
		ev1_mark		float

		ev2_mark		float
		std_totalmark		float
Methods	:	btnCalculate_click()	:	Calculate the total mark
	:	btnSubmit_click()	:	Insert Comment in database

4.2.4 Manage Schedule Module

Table 4.9: Manage Schedule Module

Responsibility	:	Allows supervisor to set schedule for meeting with their students.		
Attributes	:	Id	:	Int
		Matric_id	:	Varchar
		Name	:	Varchar
		Title		text
		Sv_name		Varchar
		date		varchar
		week		int
		start_time		Varchar
		end_time		Varchar
		location		Varchar
Methods	:	btnAdd_click()	:	Add time meeting
	:	btnDelete_click()	:	Delete time meeting

4.2.5   Generate Record Module

Table 4.10: Generate Report Module

Responsibility	:	Allows supervisor generate record of student marks		
Attributes	:	Matric_id	:	Varchar
		Password	:	Varchar
		Name	:	Varchar
		Category		varchar
		Title		text
		Sv_name		Varchar
		Ev1_name		Varchar
		Ev2_name		float
		Sv_mark1		float
		Sv_mark2		float
		Ev1_mark		float
		Ev2_mark		float
		Std_totalmark		float
Methods	:	btnSubmit_click()	:	Generate report in .xls format

4.2.6   Feedback Module

Table 4.11: Feedback Module

Responsibility	:	Allows supervisor view student progress and able to gives feedback by category		
Attributes	:	Matric_id	:	Varchar
		Name	:	varchar
		Title	:	text
		Sv_name		
		Feedback1		text
		Feedback2		text
		Feedback3		text
		Feedback4		text
		Feedback5		text
		Feedback6		text
		Feedback7		text
		Feedback8		text
		Feedback9		text
		Feedback10		text
		Feedback11		text
		Feedback12		text
		Feedback13		text
Methods	:	Name	:	Insert and update feedback into database

4.2.7 Report Module

Table 4.12: Report Module

Responsibility	:	Allows supervisor download student report		
Attributes	:	Id	:	Int
		Matric_id	:	varchar
		Name_student	:	Varchar
		title		varchar
		sv_name		varchar
		subject		Varchar
		week		int
		date		varchar
		name_file		varchar
		name		varchar
		size_file		bgint
		type_file		varchar
		content		loglob
		category		varchar
Methods	:	Download_click()	:	Download file from database

4.2.8   Upload and Download Module

Table 4.13: Upload and Download Module

Responsibility	:	Allows supervisor download file that uploaded by coordinator		
Attributes	:	Id	:	In
		subject	:	Varchar
		Name_file	:	Varchar
		Size_file		bgint
		Type_file		varchar
		content		longlob
		category		Varchar
Method	:	page_load	:	Load download page . Download file from database.

4.3   Student Package
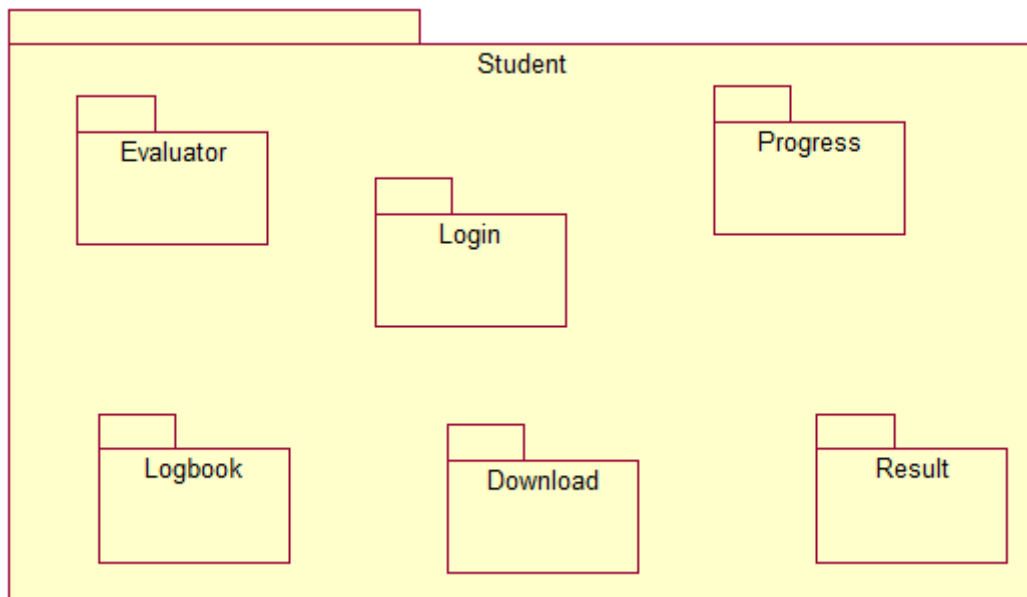
Figure 4.3 : Student Module

4.3.1 Login Module

Table 4.14: Login Module

Responsibility	:	Allows student to login		
Attributes	:	Username	:	Varchar
		Password	:	Varchar
		Category	:	Varchar
Methods	:	btnLogin_click	:	Verifies Username and password and creates session
		Page_load	:	Loads Student home page. Student is able to view their status and view the schedule.

4.3.2   Logbook Module

Table 4.15: Logbook Module

Responsibility	:	Allows student to submit logbook activity and generate logbook		
Attributes	:	Id	:	Varchar
		Matric_id	:	Varchar
		Name	:	Varchar
		Title		Text
		Date		Varchar
		Week		Int(2)
		Start_time		Varchar
		End_time		Varchar
		Progress		Varchar
		Comment		Varchar
		Expect		Varchar
		Status		Varchar
		Upload		Varchar
Methods	:	btnAdd_click()	:	Submit activity
	:	Delete_click()	:	Delete activity
	:	Generate_click()	:	Generate logbook activity in .pdf file.

4.3.3   Progress Module

Table 4.16: Progress Module

Responsibility	:	Allows student submit report by chapter and view feedback from their supervisor.		
Attributes	:	Matric_id	:	Varchar
		title	:	text
		sv_name	:	Varchar
		category		varchar
		type_report		varchar
		date		Varchar
		week		int
		Abstract		text
		Chapte1_introduction		text
		Chapter2_literature_review		text
		Chapter3_methodology		text
		Chapter4_design		text
		Chapter5_implementation		text
		Chapter6_result_discussion		text
		Chapter7_conclusion		text
		Part1_introduction		text
		Part2_report_body		text
		Part3_conclusion		text
		Appendices		text
		Refer		text
Methods	:	btnSubmit_click()	:	Insert and update feedback into database

	:	btnUpload_click()	:	Upload full report into database

4.3.4   Evaluator Module

Table 4.17: Evaluator Module

Responsibility	:	Allows student view their evaluator before presentation		
Attributes	:	Matric_id	:	Varchar
		Password	:	varchar
		Name	:	Varchar
		Category		varchar
		title		text
		Sv_name		Varchar
		Ev1_name		varchar
		Ev2_name		varchar
		Date_p		varchar
		Start_time_p		varchar
		End_time_p		varchar
		Location_p		varchar
Method	:	page_load	:	Load evaluator page.

4.3.5 Result Module

Table 4.18: Result Module

Responsibility	:	Allows student view their result of FYP		
Attributes	:	Matric_id	:	Varchar
		Password	:	Varchar
		Name	:	Varchar
		Category		varchar
		Title		text
		Sv_name		Varchar
		Ev1_name		Varchar
		Ev2_name		float
		Sv_mark1		float
		Sv_mark2		float
		Ev1_mark		float
		Ev2_mark		float
		Std_totalmark		float
Method	:	page_load	:	Load result page.

4.3.6   Download Module

Table 4.18: Download Module

Responsibility	:	Allows student view their result of FYP		
Attributes	:	Id	:	In
		subject	:	Varchar
		Name_file	:	Varchar
		Size_file		bgint
		Type_file		varchar
		content		longlob
		category		Varchar
Method	:	page_load	:	Load download page . Download file from database.

5.   REQUIREMENT TRACEABILITY

Table 5.1 : Requirement Traceability

Requirement ID	Description
(SRS-REQ-PMS-2013-1-00) (SRS-REQ-PMS-2013-1-01)	**Login** • At the Home Page interface, user needs to login first by entering username, password and category (coordinator, Student and supervisor) in the Login section to open the Index Menu under his / her account. • PMS displays the Index menu page. • Before login, student must register their information.
(SRS-REQ-PMS-2013-2-00)	**Submit Activities** • Allow lecturer to approve/reject activities submitted by their supervisee • Student will submit their activities to supervisor and to get approval. • The student's logbook will be shown in .pdf.
(SRS-REQ-PMS-2013-3-00)	**Progress** • Student will submit their project progress and supervisor shall to review the student progress. • Student must submit their project progress by weekly. • Supervisor able to update their student progress if have any mistaken.

	• The progress also must sent by chapter and depends on schedule.
(SRS-REQ-PMS-2013-4-00)	**Feedback** • Student will get feedback from supervisor when submit their progress. • Student must submit their project progress by weekly. • Once student submit their progress, student will get feedback from supervisor. • When student submit their project, supervisor will check and update the problem as they arise it.
(SRS-REQ-PMS-2013-5-00)	**View Result** • Student must submit their project progress by weekly. • Supervisor should evaluate student based on rubric. • Student able to view their result after they are completed and submit their project progress.
(SRS-REQ-PMS-2013-6-00)	**Assign Mark** • Lecturer who play as supervisor and evaluator will assign marks to student based on the rubric. • The system displays list of approved students and a form marks rubric will displayed. • Lecturer choose to supervise or evaluate their students.
(SRS-REQ-PMS-2013-7-00)	**Manage Schedule** • Supervisor set the time and arrange it for

	meet their student.
(SRS-REQ-PMS-2013-8-00)	**Upload and Download** • Coordinator upload the format of technical report, thesis, rubric and others. • Student and supervisor able to download the uploaded files by coordinator.
(SRS-REQ-PMS-2013-9-00)	**Insert Student** • Coordinator insert all data student into database • The file should be in excel format which the extension .xls.
(SRS-REQ-PMS-2013-10-00)	**Assign Evaluator** • Coordinator will choose two lecturer to be the evaluator of a student. • Evaluator can't be same • Student able to check evaluator before they are present their project.
(SRS-REQ-PMS-2013-11-00)	**Manage Student** • Coordinator have authority in manage database. • Coordinator is able to add, delete and search student information.
(SRS-REQ-PMS-2013-12-00)	**Set Timeline** • Coordinator set the timeline project.
(SRS-REQ-PMS-2013-13-00)	**Generate Record** • Coordinator will able to generate all student

	records into excel in .xls format.
(SRS-REQ-PMS-2013-14-00)	**Approve** • Supervisor approves or rejects activities submitted by their students • Lecturer check student activities. • Student can view their status of activities submitted.

6. ABBREVIATION

Table 6.1 : Abbreviation

Item	Definition
FK	Foreign Key
PK	Primary Key
REQ	Requirement
SDD	Software Design Document
SQL	Structure Query Language
SRS	Software Requirement Specification
UMP	Universiti Malaysia Pahang
PMS	Progress Monitoring System for Student Final Year Project

APPENDIX A

SDD EVALUATION FORM

SDD CHECKLIST

Implementation Identification   :   _____

Document Name   :   _____

Evaluation Date   :   _____

Put (√) if quality metric has been fulfilled or (X) if not

Document Content Completion		
a) Chapter 1 and Sub-chapters		
b) Chapter 2 and Sub-chapters		
c) Chapter 3 and Sub-chapters		
d) Chapter 4 and Sub-chapter		
e) Chapter 5		
Standard Documentation Format		
Cover Page		
Table of Content		
Page Number		
Line Spacing		
a) Paragraph (1.0 Lines)		
b) Label for Tables and Figures		
Justification / Alignment		

Numbering and Indentation		
Font (Type & Size)		
Header and Footer		

Internal Consistency		
Contradictory statement in a document		
Only listed acronym & abbreviation used in document		
Understandability		
Right use of grammar, punctuation, symbols		
Standard Abbreviation		
All terms used in forms are defined		
Tables and figures are labeled properly		
Traceability To Other Documents		
Terms, acronym & abbreviation means the same thing in the existing documents		
Requirement contradictory		

Consistency With Other Documents		
Contradictory of items in a document		
Same acronyms & abbreviation used in existing documents		

Verified by   :   _____

Post   :   _____

Date Verified   :   _____

Time   :   _____

Venue   :   _____

APPENDIX C

STR

2013

SOFTWARE TEST REPORT (STR)

PROGRESS MONITORING SYSTEM FOR
STUDENT FINAL YEAR PROJECT (FYP)

SITI NURHIDAYAH BT MOHAMAD KHAZALI
CB11008
900821035942
To be submitted to the Undergraduate Project
Bachelor of Computer Science (Software Engineering)

DOCUMENT APPROVAL

	Name	**Date**
Verified by: _____ Project Manager	Siti NurHidayah Bt Mohamad Khazali	
Authenticated by: _____ Supervisor	Miss Azlina Binti Zainuddin	
Authenticated by: _____ Client	Coordinator Undergraduate Project	

Software : IBM Rational Software Architecture (RSA), Microsoft Office 2007

Archiving Place : D:\PMS\DOCUMENTATION\SRS\

Copies Available : doc, docx,pdf

TABLE OF CONTENTS

LIST OF TABLES

1. INTRODUCTION

1.1 Purpose

The intention of this document is to state the important phases of software development life cycle is testing phase. Software testing is a vital area that is being considered and important in a changing technology. If product is tested, the quality of the system will be increase.

1.2 Abbreviations and Glossary

Table 1.1 : Abbreviations and Glossary

Abbreviation		
PMS	:	Progress Monitoring System for Student Final Year Project
IEEE	:	The Institute of Electrical and Electronic Engineers (IEEE) is an international non-profit, profesional organization for the advancement of technology related to electricity.
FYP	:	Final Year Project
Glossary		
Administrator	:	The person who manages and maintains the web based system
Supervisor	:	The person who is charge of grading student's weekly progress based on rubric.
Coordinator	:	A person who is charge of assigning students to supervisors in she/he faculty.
Student	:	The user will be updating the logbook and send their progress by weekly report.

1.3   Document Overview

This document is the Software Test Report (STR) of the Progress Monitoring System for Student Final Year Project. It contains the results of tests, which were executed during the testing phase. It is a record of the qualification testing performed in a software system or subsystem. Black box testing has being choose to test PMS.

1.4   References

This section contains the references used in producing this document.

i.   IEEE.*IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.*IEEE Computer Society, 1998.

ii.   http://share.pdfonline.com/2a295f5c17d24edfb35ba48c43b9a2b5/cmp.htm

iii. Addison Wesley, "Software Engineering", Eighth Edition, Ian Soomerville, 2007

iv.   Description about how to write Software Requirement Specification (SRS)

2. OVERVIEW OF TEST RESULTS

2.1 Test Log

Progress Monitoring System for Student Final Year Project will develop using web based application, Adobe Dreamweaver 8 and PHP scripting language, and interact with MySQL Server. PMS was tested and focus on black box testing.

2.2 Rational for decision

- Pass : The test result is set to "Pass" when all the steps are in correct state and the real result is compliant to the expected results.

- Fail : The test result is set to "Fail" when all the steps are in correct state and the real result is not compliant to the expected result.

2.3 Overall Assessment of Test

All the tests with interfaces are passed but the Graphical User Interface (GUI) is not optimized and fully utilize for screens of all the types of browser platform or version.

2.4 Impact of Test Environment

First, setting up a test environment to evaluate the system is very important practice for successful deployments. When perform a complete product evaluation and a scenario -based test plan, generally we will gain different knowledge of a product and how it works in different situations. So, tester should have ability to take the point of view from customer and intermediate knowledge.

3.   DETAIILED TESTS RESULTS

3.1   Unit Testing

Unit testing is used to test individual part of coding where the test plan is design based on the specific module. Any error that found in unit testing will be fixed immediately by developer when the error is found. Unit testing can also ensure that all input data is in correct format and no error when passing the data within the database.

3.1.1   Login Function

Table 3.1 : Coordinator Login

No	Event	Attribute and Value	Expected Result	Result
1.	Verify login user after the correct input data is submit on login form	Username: COOR_UMP Password: 12345 Category : Coordinator	Successful login to admin home page	Pass
2.	Verify login user after the null value is submit on login form	Username: Password:	Message box displayed request user to input the empty field	Pass
3.	Verify login user after the invalid value is submit on login form	Username: COOR_UMP Password: password	Login fail and the page redirect back to the home page.	Pass

Table 3.2 : Supervisor Login

No	Event	Attribute and Value	Expected Result	Result
1.	Verify login user after the correct input data is submit on login form	Username: azlina Password: azlinaz@ump.edu.my Category : Supervisor	Successful login to supervisor home page	Pass
2.	Verify login user after the null value is submit on login form	Username: Password:	Message box displayed request user to input the empty field	Pass
3.	Verify login user after the invalid value is submit on login form	Username: azlina Password: password	Login fail and the page redirect back to the home page.	Pass

Table 3.3 : Student Login

No	Event	Attribute and Value	Expected Result	Result
1.	Verify login user after the correct input data is submit on login form	Username:CB11008 Password:CB11008 Category : Student	Successful login to student home page	Pass
2.	Verify login user after the null value is submit on login form	Username: Password:	Message box displayed request user to input the empty field	Pass
3.	Verify login user after the invalid value is submit on	Username: CB11008	Login fail and the page redirect back to	Pass

	login form	Password: password	the home page.	

3.1.2 Coordinator: Import Student List

Table 3.4 : Import Student List

No	Event	Attribute and Value	Expected Result	Result
1.	Verify file type after click on the Import button on import student list form with selected excel file		File uploaded successfully and the records are save into database	Pass
2.	Verify file type after click on the Import button on import student list form without a file		Error message is displayed and request user to select a file	Pass
3.	Verify file type after click on the Import button on import student list form with not an excel file		Error message displayed and request user to select only an excel file	Pass

3.1.3 Coordinator : Search Student

Table 3.5 : Search student

No	Event	Attribute and Value	Expected Result	Result
1.	Insert Matric Number	Matric_id:CB11008 Category:Matric ID	Display user information	Pass
2.	Insert student name	Name : hidayah Category: name	Display user information	Pass
3.	Search name in category Matric ID	Name : Hidayah Category : Matric ID	Error message displayed and request user to insert others value. "SORRY, BUT WE CAN NOT FIND AN ENTRY TO MATCH YOUR QUERY"	Pass

3.1.4 Coordinator : Assign Evaluator

Table 3.6 : Assign Evaluator

No	Event	Attribute and Value	Expected Result	Result
1.	Insert category and Supervisor name for assigning Evaluator	Category : PSM2 Select : AZLINA BT ZAINUDDIN	Display student list	Pass
2.	Select student for assign evaluator	Matric ID : CB11008	Page assign evaluator will display.	Pass
3.	Verify evaluator name after click on Update on select/assign evaluator form with correct input	First Evaluator: MOHD HAFIZ BIN MOHD HASSIN	Successful update and the record will save into database	Pass

		Second Evaluator: ROHANI BINTI ABU BAKAR		
4.	Verify evaluator name after click on Update on select/assign evaluator form with same input value	First Evaluator: MOHD HAFIZ BIN MOHD HASSIN Second Evaluator: MOHD HAFIZ BIN MOHD HASSIN	Error message displayed stated both evaluator cannot be the same "Evaluator Cannot be Same"	Pass
5.	Verify evaluator name after click on Update on select/assign evaluator form with more than 8 same input value	First Evaluator: MOHD HAFIZ BIN MOHD HASSIN Second Evaluator: ROHANI BINTI ABU BAKAR Condition: ZALILI BINTI MUSA >8	Error message displayed stated an evaluator cannot manage more than 8 students	Pass

3.1.5 Coordinator : Generate Student's Record

Table 3.7: Generate student's Record

No	Event	Attribute and Value	Expected Result	Result
1.	Verify the selection made by user with correct value	All records	Successful generate all students record into excel file	Pass
2.	Verify the selection made by user with null value		Error message is displayed and request user to choose a category to generate record. "Please Choose a Category"	Pass

3.1.6 Coordinator : Timeline

Table 3.8: Set Timeline

No	Event	Attribute and Value	Expected Result	Result
1.	Insert Close Date, Week and Task	Input : 16/12/2013 Select : 14 Task : "Presentation at Astaka"	Successful save into database	Pass
2.	Verify the selection made by user with null value		Error message is displayed. "Please Enter The date For the Close Date" "Please Enter The Week to Submit"	Pass

| | | | "Please Enter The Task To Inform" | |
| | | | | |

3.1.7 Coordinator : Upload

Table 3.9: Upload

No	Event	Attribute and Value	Expected Result	Result
1.	Insert File name, File category, Subject / Others and Choose file	File name : Example Category : Format Technical Report Subject : New Format File : .pdf, .ppt,.xls,.doc, .zip	Successful save into database	Pass
2.	Verify the selection made by user with null value		Error message is displayed. "Enter File Name" "choose the right category for right purpose" "Enter the Subject" "Choose file to upload"	Pass

3.1.8 Supervisor : Approve Activities

Table 3.10 : Approve Activities

No	Event	Attribute and Value	Expected Result	Result
1.	Verify action of approval after click on Submit button on approve logbook activities form with correct input values	Status: Approve Comment: Good test technique Expected Progress: More test data run	Successful insert the action into database and redirect back to home page of approve activities	Pass
2.	Verify action of approval after click on Submit button on approve logbook activities form with null input values	Status: Comment: Expected Progress:	Error message is displayed and request user to insert the missing field	Pass

3.1.9   Supervisor : Evaluate Marks

Table 3.11 : Evaluate Marks

Test Case ID	Event	Attribute and Value	Expected Result	Result
1.	Verify total marks given to student after click on Submit button on different category of project with correct input values	Matric No: CB11008	Successful insert the marks into database	Pass
2.	Verify total marks given to student after click on Submit button on different category of project with null input values	Matric No: Total:	Error message is displayed and request user to insert the missing field	Pass

3.1.10 Supervisor : Feedback

Table 3.12 : Feedback

No	Event	Attribute and Value	Expected Result	Result
1.	Verify student progress. If student choose Technical Report.	Feedback : Good	Successful insert comment	Pass
2.	Verify student progress. If student choose Thesis	Feedback : Good	Successful insert comment.	Pass

3.1.11 Student : Submit Activities

Table 3.13 : Submit Activities

No	Event	Attribute and Value	Expected Result	Result
1.	Verify action of approval after click on Submit button on approve logbook activities form with correct input values	Meeting Date: 12/12/2013 Meeting Time (Start): 2:00 pm Meeting Time (End): 3:30 pm Week: 3 Progress: Requirement	Successful insert activities to database and redirect to home page of log book	Pass
2.	Verify activities submitted after click on Add button on submit activities form with correct null values	Meeting Date: Meeting Time (Start): Meeting Time (End): Week: Progress:	Error message is displayed and request user to insert the missing field	Pass

3.1.12 Student : Submit Progress

Table 3.14 : Submit Progress

No	Event	Attribute and Value	Expected Result	Result
1.	Submit progress by weekly depends on type of report.	Type Report :Technical Report Type Report : Thesis	Successful insert progress into database	Pass
2.	Verify activities submitted after click on Submit button and user can update it.	Abstract : " The method will be approach for construct PMS is Rapid Application Development (RAD). "	Successful insert into database.	Pass

3.1.13 Student : Submit Report

Table 3.15 : Submit Report

No	Event	Attribute and Value	Expected Result	Result
1.	Verify file and submit .	Date : 15/12/2013	Successful insert progress into database	Pass
2.	Verify activities submitted after click on Submit button and user can update it.	Abstract : " The method will be approach for construct PMS is Rapid Application Development (RAD). "	Successful insert into database.	Pass

3.2 Functional Testing

After unit testing is completed, functional testing will be continued. Functional testing is to test the functionality of each module to ensure the requirements of user are met. Following is the text plan that prepared by developer for functional testing.

3.2.1 Login with different users

Table 3.16 : Login with different users

No	Event	Attribute and Value	Expected Result	Result
1.	Login as Student	Username: CB11008 Password: cb11008 Category: Student	Home page of student displayed	Pass
2.	Login as Lecturer	Username: azlina Password: azlinaz@ump.edu.my Category: Supervisor	Home page of lecturer displayed	Pass
3.	Login as Admin	Username: COOR_UMP Password: 12345 Category: Coordinator	Home page of admin displayed	Pass

3.2.2 Print excel records

Table 3.17 : Print excel Records

No	Event	Attribute and Value	Expected Result	Result
1.	Login as Coordinator	Username: COOR_UMP Password: 12345 Category: Coordinator	Home page of Coordinator displayed	Pass
2.	Login as Supervisor	Username: azlina Password: azlinaz@ump.edu.my Category: Supervisor	Home page of supervisor displayed	Pass
3.	Click on Record menu		A dropdown list is displayed to request coordinator and supervisor choose a categories	Pass
4.	Select a categories to generate the excel file	Select All Records	The list of all student's information will save into excel file.	Pass

3.2.3 Marks Process

Table 3.18 : Mark Process

No	Event	Attribute and Value	Expected Result	Result
1.	Login as Supervisor	Username: azlina Password: azlinaz@ump.edu.my Category: Supervisor	Home page of supervisor displayed	Pass
2.	Click on Evaluate menu and then choose student category	Category :PSM1 Category :PSM2 Category : PTA	Display student by categories - All list of student register with category - Evaluate form with progress 20% and 40% is shown .	Pass
3.	Select student Matric No, click calculate button after select the scale of mark.	Matric NO : CB11008 Total : 15	The mark will saved into database	Pass
4.	Click category on home page	Category :PSM1 Category :PSM2 Category : PTA	Display student by categories - All list of student register with category - Evaluate student that assigned from coordinator.	Pass
5.	Login as student to view the result	Username : CB11008 Password : CB11008 Category : student	Choose Result menu for view the result that assigned from supervisor and evaluator.	Pass

3.2.4 Assign Evaluator

Table 3.19 : Assign Evaluator

No	Event	Attribute and Value	Expected Result	Result
1.	Login as Coordinator	Username: COOR_UMP Password: 12345 Category: Coordinator	Home page of coordinator displayed	Pass
2.	Click on Evaluator menu.	-choose 1- Category :PSM1 Category :PSM2 Category : PTA Supervisor : AZLINA BINTI ZAINUDDIN	The list of student by supervisor.	Pass
3.	Choose a student and click on Edit link to insert evaluator.	Select EDIT	The list of particular students is displayed	Pass
4.	Select two lecturer to assign as evaluator for a single student.	First Evaluator: MOHD HAFIZ BIN MOHD HASSIN Second Evaluator: ROHANI BINTI ABU BAKAR	Both evaluator is assigned to the student.	Pass
5.	Login as Evaluator1 Login as Evaluator1	Username : mohdhafiz Password : hafizhassin@ump.edu.my Category : Supervisor Username : rohani Password : rohani@ump.edu.my Category : Supervisor	Home page of supervisor displayed.	
6.	Click on project categories at the home page and choose student categories.	-PSM1 - PSM2 - PTA	Three categories is shown	

7.	Click on PSM2	Select PSM2	- All list of student register for PSM2. - Evaluation form with PSM2 criteria is shown below the student list.	Pass
8.	Student can view their evaluator and venue before present.	Username : CB11008 Password: CB11008 Category : student	- Evaluator menu - Details of evaluator name and venue	Pass

3.3   User Acceptance Test

User Acceptance Test refers the final stage for testing stage of a system. When the test is done or is successful, it indicates the agreement to implement the system lives. Enhancement and some small changes may still need to be test, but the test shows the system is considered stable and able to process data according to requirements.

	Student
	Supervisor
	PSM / PTA Coordinator

Questions	Strongly Disagree 1	Disagree 2	Natural 3	Agree 4	Strongly Agree 5
I am able to complete my work quickly using this system					
Overall, I am satisfied with how easy it is use this system					
It was simple to use this system					
I was able to complete the tasks and scenarios quickly using this system.					
I felt comfortable using this system					
It was easy to learn to use this system					
The system gave error messages that clearly told me how to fix problems.					
Whenever I made a mistake using this system, I could recover easily and quickly					
It was easy to find the information that I needed					
The information was effective in helping me complete the tasks and scenarios					
The organization of information on the system screens was clear					
The interface of this system was pleasant					
This system has all the functions and capabilities that I expect it to have					
Overall, I am satisfied with this system.					

Figure3.1 : Survey on PMS
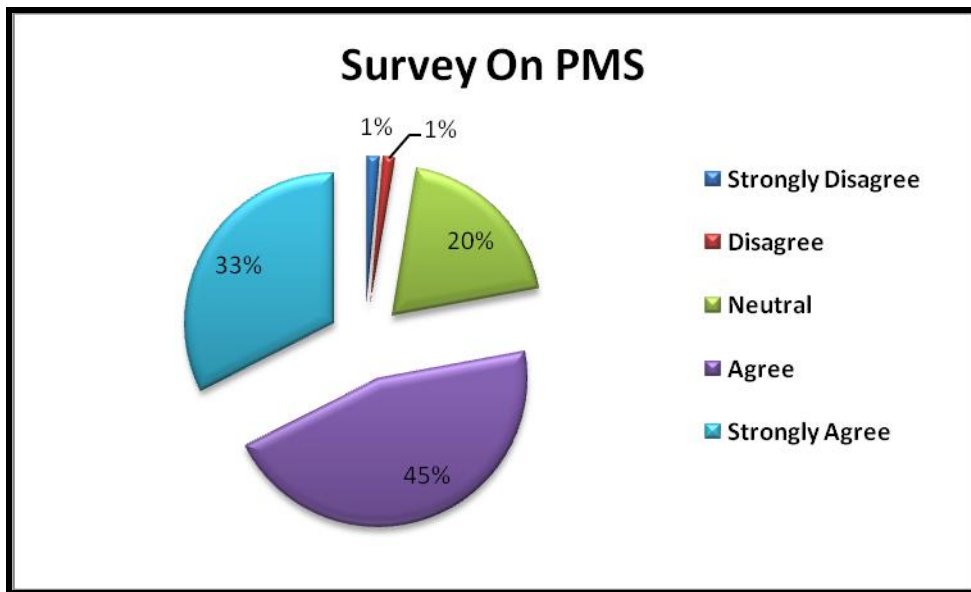
User acceptance test is conducted by using a questionnaire; the respondents consist of student, Supervisor, and coordinator. The results shows 98% of the respondents satisfies with the system.

APPENDIX A

FORM QUESTIONNAIRE

A-1   User Acceptance Test from Coordinator

Universiti Malaysia PAHANG

FACULTY OF COMPUTER SYSTEM & SOFTWARE ENGINEERING

Survey on Progress Monitoring System for Student Final Year Project (PMS)

Select *

	Student
	Supervisor
✓	PSM / PTA Coordinator

Questions	Strongly Disagree 1	Disagree 2	Natural 3	Agree 4	Strongly Agree 5
I am able to complete my work quickly using this system				✓	
Overall, I am satisfied with how easy it is use this system				✓	
It was simple to use this system				✓	
I was able to complete the tasks and scenarios quickly using this system.				✓	
I felt comfortable using this system				✓	
It was easy to learn to use this system				✓	
The system gave error messages that clearly told me how to fix problems.				✓	
Whenever I made a mistake using this system, I could recover easily and quickly				✓	
It was easy to find the information that I needed				✓	
The information was effective in helping me complete the tasks and scenarios				✓	
The organization of information on the system screens was clear				✓	
The interface of this system was pleasant				✓	
This system has all the functions and capabilities that I expect it to have				✓	
Overall, I am satisfied with this system.				✓	

13/12/2018

DR. NORROZILA BINTI SULAIMAN
Senior Lecturer
Faculty Of Computer Systems & Software Engineering
Universiti Malaysia Pahang
Lebuhraya Tun Razak, 26300 Gambang, Kuantan, Pahang
Tel : 09-549 2119 Fax : 09-549 2144

A-2 User Acceptance Test from Supervisor

Universiti
Malaysia
PAHANG

FACULTY OF COMPUTER SYSTEM & SOFTWARE ENGINEERING

Survey on Progress Monitoring System for Student Final Year Project (PMS)

Select *

	Student
✓	Supervisor
	PSM / PTA Coordinator

Questions	Strongly Disagree 1	Disagree 2	Natural 3	Agree 4	Strongly Agree 5
I am able to complete my work quickly using this system				✓	
Overall, I am satisfied with how easy it is use this system				✓	
It was simple to use this system				✓	
I was able to complete the tasks and scenarios quickly using this system.					✓
I felt comfortable using this system					✓
It was easy to learn to use this system					✓
The system gave error messages that clearly told me how to fix problems.				✓	
Whenever I made a mistake using this system, I could recover easily and quickly				✓	
It was easy to find the information that I needed					✓
The information was effective in helping me complete the tasks and scenarios					✓
The organization of information on the system screens was clear					✓
The interface of this system was pleasant					
This system has all the functions and capabilities that I expect it to have					✓
Overall, I am satisfied with this system.					✓

Name :
Date : 13 / 12 / 2013

ROZLINA BINTI MOHAMED
LECTURER
FACULTY OF COMPUTER SYSTEMS
& SOFTWARE ENGINEERING
UNIVERSITI MALAYSIA PAHANG
LEBUHRAYA TUN RAZAK
26300 GAMBANG, KUANTAN, PAHANG
TEL. 09-5492131 FAX: 09-5492144

A-3   User Acceptance Test from Student

FACULTY OF COMPUTER SYSTEM & SOFTWARE ENGINEERING

Survey on Progress Monitoring System for Student Final Year Project (PMS)

Select *

/	Student
	Supervisor
	PSM / PTA Coordinator

Questions	Strongly Disagree 1	Disagree 2	Natural 3	Agree 4	Strongly Agree 5
I am able to complete my work quickly using this system				/	
Overall, I am satisfied with how easy it is use this system				/	
It was simple to use this system				/	
I was able to complete the tasks and scenarios quickly using this system.				/	
I felt comfortable using this system				/	
It was easy to learn to use this system				/	
The system gave error messages that clearly told me how to fix problems.				/	
Whenever I made a mistake using this system, I could recover easily and quickly				/	
It was easy to find the information that I needed				/	
The information was effective in helping me complete the tasks and scenarios				/	
The organization of information on the system screens was clear				/	
The interface of this system was pleasant				/	
This system has all the functions and capabilities that I expect it to have				/	
Overall, I am satisfied with this system.				/	

Name : NURUL ADIBAH BINTI JUMRI
Date : 15.12.2013

APPENDIX D

GANTT CHART

		Task Name	Duration	Start	Finish	Pre
1		Requirements Planning	26 days?	Thu 5/9/13	Thu 10/10/13	
2		Identify Scope and Problem	1 day	Thu 5/9/13	Thu 5/9/13	
3		Identify Purpose of System development	6 days	Fri 6/9/13	Fri 13/9/13	
4		Doing Research on Manual System	7 days	Mon 16/9/13	Tue 24/9/13	3
5		Identifiers Improvement Opportunities	8 days?	Wed 25/9/13	Fri 4/10/13	4
6		Develop a Concept For the new System	3 days	Mon 7/10/13	Wed 9/10/13	5
7		Submit Proposal	1 day?	Thu 10/10/13	Thu 10/10/13	6
8		User Design	15 days	Fri 11/10/13	Thu 31/10/13	
9		The Interior and external design for System	15 days	Fri 11/10/13	Thu 31/10/13	
10		Element that involves in system also were defined	14 days	Mon 14/10/13	Thu 31/10/13	
11		Design interface and System Progress	13 days	Tue 15/10/13	Thu 31/10/13	
12		Construction Phase	23 days?	Fri 1/11/13	Tue 3/12/13	
13		Construct System	23 days	Fri 1/11/13	Tue 3/12/13	
14		Program and Module Code is Designed	22 days?	Mon 4/11/13	Tue 3/12/13	
15		System test and solve problem	9 days?	Thu 21/11/13	Tue 3/12/13	
16		Cutover Phase	9 days?	Wed 4/12/13	Mon 16/12/13	15
17		Maintain and Upgrades the developes System	8 days?	Wed 4/12/13	Fri 13/12/13	
18		Verify all function in system	4 days	Tue 10/12/13	Fri 13/12/13	
19		System Launching	1 day	Mon 16/12/13	Mon 16/12/13	18

APPENDIX E

SAMPLE CODING

Generate logbook

```php
<?php
require('fpdf.php');
include('../student_auth.php');
class PDF extends FPDF
{

// Page footer
function Footer()
{
  // Position at 1.5 cm from bottom
  $this->SetY(-15);
  // Arial italic 8
  $this->SetFont('Arial','I',8);
  // Page number
        $this->Cell(0,10,'Page '.$this->PageNo().'/{nb}',0,0,'C');
        $this->Cell(0,10,'PSM LOG BOOK',0,0,'R');
}

function ChapterTitle()
{
  $this->Image('umplogo.gif',65,2,80,30);
        $this->Image('logbook.gif',30,40,150,100);
  // Arial bold 14
  $this->SetFont('Times','B',14);
  // Color RGB Green
        $this->SetTextColor(124,252,0);
  // Move to the right
  $this->Cell(16);
  // Title
  $this->Cell(30,80,'FACULTY OF COMPUTER SYSTEMS & SOFTWARE ENGINEERING',0,'C');
  // Line break
  $this->Ln(20);
          // Arial bold 34
  $this->SetFont('Times','B',34);
  // Color RGB Yellow
        $this->SetTextColor(255,255,0);
        // Move to the right
        $this->Cell(8);
        // Title
        $this->Cell(30,60,'UNDERGRADUATE PROJECT',0,'C');
        // Line break
  $this->Ln(20);
          // Times New Roman bold 52
  $this->SetFont('Times','B',60);
  // Color RGB Purple
        $this->SetTextColor(147,112,219);
        // Move to the right
        $this->Cell(45);
        // Title
        $this->Cell(30,210,'Log Book',0,'C');
        // Line break
  $this->Ln(120);
        // Arial 12
  $this->SetFont('Times','B',14);
  // Color RGB Yellow
        $this->SetTextColor(255,255,255);
        // Background color
  $this->SetFillColor(30,144,255);
  // Title
  $this->Cell(0,10,'STUDENT PROFILE',0,1,'C',true);
```

```php
        // Line break
    $this->Ln(8);
}

function Detail($data)
{
            // Arial 12
    $this->SetFont('Times','',14);
    // Color RGB Black
            $this->SetTextColor(0,0,0);
    // Title
            foreach ($data as $eachResult)
            {
            $this->Cell(0,6,'Name          : '.$eachResult["name"].'',0,1,'L');
            $this->Ln(4);
            $this->Cell(0,6,'Matric No     : '.$eachResult["matric_id"].'',0,1,'L');
            $this->Ln(4);
            //$this->Cell(0,6,$eachResult[""],0,1,'L');$this->Ln(4);
            $this->Cell(0,6,'Category      : '.$eachResult["category"].'',0,1,'L');
            $this->Ln(4);
            $this->Cell(0,6,'Project Title  : '.$eachResult["title"].'',0,1,'L');
            $this->Ln(4);
            $this->Cell(0,6,'Supervisor     : '.$eachResult["sv_name"].'',0,1,'L');
            $this->Ln(4);
            }
}
function Page2()
{
            $this->Image('page2.gif',30,20,150,150);
}


function Page3()
{
            $this->Image('reportsummary.gif',30,20,160,270);
}

function Page4($header,$data)
{
            //Header
            $this->SetFont('Times','B',14,'C');
            $this->SetFillColor(255,215,0);
    // Title
            $this->SetTextColor(255,250,250);
    $this->Cell(0,10,'UNDERGRADUATE PROJECT DIARY',0,1,'C',true);
            $this->Ln(5);

            $this->Image('page4.gif',10,40,170,60);
            $this->Ln(70);
            $this->SetFont('Times','B',14,'C');
            $this->SetTextColor(0,0,0);
            $this->Cell(10);
            $this->Cell(0,10,'REPORT ON MEETING DATE BETWEEN STUDENT AND SUPERVISOR',0,'C');

            $this->Ln(15);
            $this->SetFillColor(139,137,137);
            $this->SetFont('Times','B',12,'C');
            $this->Cell(35);
            $w=array(25,35,60);
            for($i=0;$i<count($header);$i++)
            $this->Cell($w[$i],7,$header[$i],1,0,'C',true);
            $this->Ln();
            $this->SetFont('Times','',12,'C');
            //Data
```

```php
            foreach ($data as $eachResult)
            {
                    $this->Cell(35);
                    $this->Cell(25,10,$eachResult["week"],1,0,'C');
                    $this->Cell(35,10,$eachResult["date"],1,0,'C');
                    $this->Cell(60,10,$eachResult["start_time"]. ' - ' .$eachResult["end_time"] ,1,0,'C');
                    $this->Ln();
            }


}

function Page5()
{
            $this->SetFont('Times','B',28);
    // Color RGB
            $this->SetTextColor(0,0,0);
            // Move to the right
            $this->Cell(35);
            // Title
            $this->Cell(0,10,'PROJECT GANNT CHART',0,'C');
            $this->Ln(15);
            $this->SetFont('Times','',12);
            $this->Cell(35);
            $this->Cell(0,10,'Please paste your project Gantt Chart here:',0,'C');
}

//Load data
function LoadData($file)
{
            //Read file lines
            $lines=file($file);
            $data=array();
            foreach($lines as $line)
            $data[]=explode(';',chop($line));
            return $data;
}

//Simple table
function BasicTable($header1,$data)
{
            //Header
            $this->SetFont('Times','B',14,'C');
            $this->SetFillColor(255,215,0);
            $this->SetTextColor(255,250,250);
    // Title
    $this->Cell(0,10,'PROJECT PROGRESS SUMMARY',0,1,'C',true);
            $this->Ln(5);
            $this->SetFont('Times','B',12,'C');
            $this->SetFillColor(255,255,0);
            $this->SetTextColor(0,0,0);
            $w=array(25,55,55,55);
            //Header
            for($i=0;$i<count($header1);$i++)
            $this->Cell($w[$i],15,$header1[$i],1,0,'C',true);
            $this->Ln();
            //Data
            foreach ($data as $eachResult)
            {
```

APPENDIX F

TURN IT IN RESULT

turnitin

--% SIMILAR

-- OUT OF 0

Originality | GradeMark | PeerMark

report

BY SITI NURHIDAYAH KHAZALI

undergraduate's subject in university. This way, it helps to track student in they are

have problem to finish their project. The step for approach student totally different for

each supervisor. So, the best way to keep track student is through web based system.

Before this, data and information for FYP is recorded as manually. Staffs need to

record all data and information and keep all information in files. Recorded as manually

will make the data and information lost. This is not consistent in order to find old record

No Service Currently Active