**BIT302**

**SOFTWARE ENGINEERING**

**Assignment 1**

**Project Proposal**

**“Web based Academic Information System”**

Team Leader: Lovely (E1500367)

Contact Details: lovelyzheng98@gmail.com

Member: I Wayan Denny Harry Caka (E1500365)

Contact Details: dennycaka@gmail.com

**Table of Contents**

[Introduction 3](#_Toc498518228)

[Project Background 4](#_Toc498518229)

[Project Aims 5](#_Toc498518230)

[Project Objectives 5](#_Toc498518231)

[Project Scope 6](#_Toc498518232)

[Project Schedule 8](#_Toc498518233)

[Work Breakdown Structure 10](#_Toc498518234)

[Milestones 10](#_Toc498518235)

[Gantt Chart 11](#_Toc498518236)

[Development Platform 13](#_Toc498518237)

[Demonstration Platform 14](#_Toc498518238)

[Risk Management Plan 15](#_Toc498518239)

[References 17](#_Toc498518240)

# Introduction

In the last decade, the Internet and World Wide Web has been developed rapidly and gained so much popularity due to its usefulness in all aspect of our life. Almost all businesses nowadays are web – enabled to improve and enhance their operations. By taking advantage of web technologies, an organization can reach out to customers and provide them with not only general information about its products or services but also the opportunity of performing interactive business transactions [1].

Information system is an example of business needs that have migrated to the web as a web based application. A web based application is a software package that can be accessed through the web browser. The software and database reside on a central server rather than being installed on the desktop system and is accessed over a network [2].

Information system itself can be described from two different perspectives: one that relates to the system’s purposes; the other relates to the system’s structure. It can be said that an information system is a technologically implemented medium for the purpose of recording, storing, and disseminating linguistic expressions as well as for the supporting of inference making. Structure-wise, an information system consists of a collection of people, processes, data, models, technology and partly formalized language, forming a cohesive structure which serves some organizational purpose or function [3].

# Project Background

As what has been mentioned before, many organizations are using the web to improve their services to the customers. School is an example of organization that often uses web technologies. Many schools have a website for visitors to see information or news related to the school programs and activities. However, there aren’t many schools that has their own academic information system for the students.

In most schools, mark list of students is recorded in very simple form, for example in Microsoft Excel. Students can only see their marks through report cards that given to them only once every semester. Using Excel is still important to put formula and calculate final marks for students, but a report card is very traditional and cannot be accessed any time.

Because of that reason, we have chosen to create a web based academic information system for a junior high school. Through the system, we are hoping that students can check their marks for any subject anytime anywhere. Students can also see the summary of their marks and from previous years. For example, a student is now in 9th grade, he/she can see marks details for the current semester, but can only see the summary of marks when he/she was in 7th and 8th grade.

Additionally, students can check for recent announcement through the web application, so that they will not miss any information about the school upcoming activities. Students can also check their schedule and teacher information. Teacher information that can be viewed include address and phone number, to make it easier for students or parents to contact the teachers if there is any issue.

To build this system, we have to know what subjects are generally taught in junior high school. We also have to know how many tests and marks are given to a student in a semester, and what is the formula to calculate the final mark. This academic system will also show personal information of the student, so we have to know what kind of information will be included (student ID, student name, student address, etc.).

We will put the data of marks and personal information in a database and it will be handled by an admin. The admin will be given password to access the system to add, delete and update the data. Each student will get a user ID and password to see their own data and information, and the contact details of teachers in their school. The desired outcome will be a web – based academic information system that allows student to see their information anytime.

# Project Aims

* To change the traditional way of showing marks to students from paper – based report card to online information system
* To ease communication between students and teachers, by providing them with teacher contact details
* To provide students with latest announcement from the school so that they will not miss any school’s activity
* To design a web based application which connect database to the website so that data can be updated anytime and information can be viewed anytime anywhere

# Project Objectives

1. Conducting a research about how many and what subjects are generally taught in junior high school
2. Determining how many tests are being given to student per subject per year to know how many marks will be entered in the database
3. Select tools and programming language that are most suitable to develop an information system
4. Produce documents that are necessary for the completion of this project
5. Decide how the user interface will be designed and determine the color scheme
6. Creating database and input all the data that are needed
7. Integrating database and web design to produce a complete application

# Project Scope

|  |
| --- |
| **Project Title:** Web based Academic Information System  **Date:** 27th October 2017  **Prepared by:** Lovely, Project Team Leader, lovelyzheng98@gmail.com |
| **Project Summary and Justification:**  This information system is made to help students in junior high school to see teachers’ information, mark details, schedule and announcements. Through the system, we are integrating database system and website design. The system will be managed by an admin to add, delete and update data. As a result, students don’t have to wait until the end of the semester to see their marks in form of paper – based report cards. Students also do not have to depend on hard – copied schedule to check their subject time table. |
| **Product Characteristics and Requirement:**  1. Do some research on junior high schools that have applied web based academic information system for their students  2. Providing content that suitable to solve problems that students have. The web based application should be informative and helpful by showing all data that are students need to see.  3. The web application will be tested with different browsers to make sure it is accessible and does not have any display problem.  4. All the link will be tested, to see if the link work properly or not. It will be tested once a week. |
| **Summary of Project Deliverables**  **Project management-related deliverables:** Project aims, project objectives, scope statement, WBS, schedule, cost baseline, requirements specification document, design and testing documentation, web based academic information system that is useful and can work, final project presentation, and other documents required to manage the project.  **Product-related deliverables:**   1. Web based academic information system that can be accessed by any registered student in a school 2. The content of the web allows student to know the marks that have obtained for the current semester and parents can see their child’s marks. 3. Ability to manage relationship with clients, which in this project are school and its students by answering their questions related to the system, to avoid confusion and misunderstandings. |
| **Project Success Criteria:** Our goal is to complete this project within three months. The project will be considered successful if it meets all of the product characteristic and requirement listed above, and does not misaligned with the project scope. The project team will succeed if they can follow team contract and stay on track of WBS and Gantt Chart that has been created. If the information system for junior high school is successful and very helpful for students, similar system can be developed and applied for primary and high schools too. |

# Project Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schedule** | **Start Date** | **End Date** | **Estimate Days** | **Responsible** |
| **Initiating** | | | | |
| Identifying Topic | Mon 16/10/17 | Mon 16/10/17 | 1d | All |
| Conducting Research | Tue 17/10/17 | Fri 20/10/17 | 4d | All |
| Identifying Project Aims and Objectives | Mon 23/10/17 | Tue 24/10/17 | 2d | All |
| Identifying Non – Functional and Functional Requirements | Wed 25/10/17 | Thu 26/10/17 | 2d | All |
| Complete Initiating Task | Thu 26/10/17 | Thu 26/10/17 | 0d | All |
| **Planning** | | | | |
| Develop Scope Statement | Fri 27/10/17 | Mon 30/10/17 | 2d | Lovely |
| Develop WBS | Tue 31/10/17 | Tue 31/10/17 | 1d | Denny |
| Create Project Schedule | Wed 01/11/17 | Wed 01/11/17 | 1d | Denny |
| Baseline Gantt Chart | Thu 02/11/17 | Thu 02/11/17 | 1d | Denny |
| Determine Development and Demonstration Platform | Fri 03/11/17 | Mon 06/11/17 | 2d | All |
| Develop Risk Management Plan | Tue 07/11/17 | Tue 07/11/17 | 1d | Lovely |
| Develop Use Case Diagram and Use Cases | Wed 08/11/17 | Thu 09/11/17 | 2d | All |
| Develop Expanded Use Cases | Fri 10/11/17 | Mon 13/11/17 | 2d | Denny |
| Develop Analysis Class Diagram | Tue 14/11/17 | Tue 14/11/17 | 1d | Lovely |
| Develop Sequence Diagram | Wed 15/11/17 | Wed 16/11/17 | 2d | Lovely |
| **Executing** | | | | |
| Collecting Data | Thu 02/11/17 | Tue 07/11/17 | 4d | All |
| Inputting Data to Database System | Thu 02/11/17 | Tue 07/11/17 | 4d | All |
| Designing the Web Page | Wed 08/11/17 | Fri 24/11/17 | 13d | All |
| Integrating the Database with the Web Design | Mon 27/11/17 | Mon 18/12/17 | 16d | All |
| Produce Prototype #1 | Mon 18/12/17 | Mon 18/12/17 | 0d | All |
| Improve and Update the System | Tue 19/12/17 | Tue 26/12/17 | 4d | All |
| Produce Complete System | Tue 26/12/17 | Tue 26/12/17 | 0d | All |
| **Monitoring and Controlling** | | | | |
| Update Gantt Chart | Mon 27/11/17 | Mon 27/11/17 | 1d | Denny |
| Testing Prototype #1 | Tue 19/12/17 | Wed 20/12/17 | 2d | All |
| Testing the Complete System | Tue 26/12/17 | Wed 27/12/17 | 2d | All |
| **Closing** | | | | |
| Presentation and Demonstration for Prototype #1 | Thu 21/12/17 | Thu 21/12/17 | 0d | All |
| Presentation and Demonstration for the Complete System | Thu 28/12/17 | Thu 28/12/17 | 0d | All |

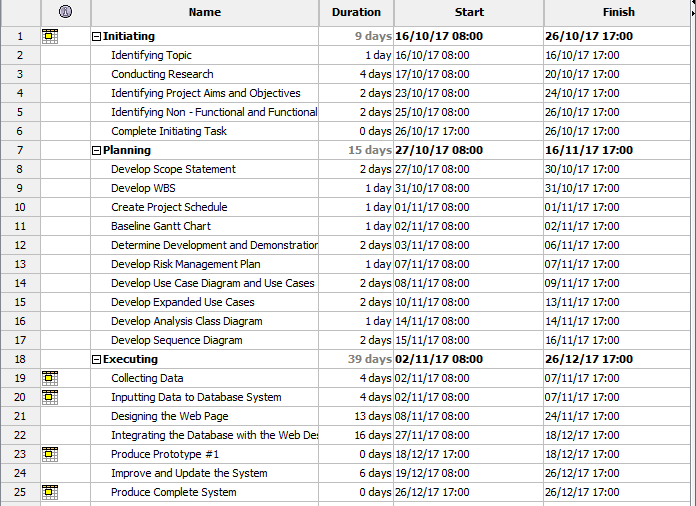
# Work Breakdown Structure

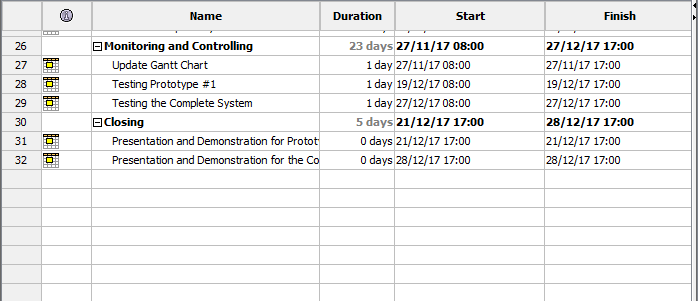
1. Initiating Tasks
   1. Identifying Topic
   2. Conducting Research
   3. Identifying Project Aims and Objectives
   4. Identifying Non – functional and Functional Requirements
   5. Complete Initiating Task
2. Planning Tasks
3. Scope Statement
4. WBS
5. Project Schedule
6. Baseline Gantt Chart
7. Development and Demonstration Platform
8. Risk Management Plan
9. Use Case Diagram and Use Cases
10. Expanded Use Cases
11. Executing Tasks
12. Collecting Data
13. Inputting Data to Database System
14. Designing the Web Page
15. Produce Prototype #1
16. Produce Complete System
17. Monitoring and Controlling Tasks
18. Update Gantt Chart
19. Testing Prototype #1
20. Testing the Complete System
21. Closing
22. Presentation and Demonstration for Prototype #1
23. Presentation and Demonstration for the Complete System

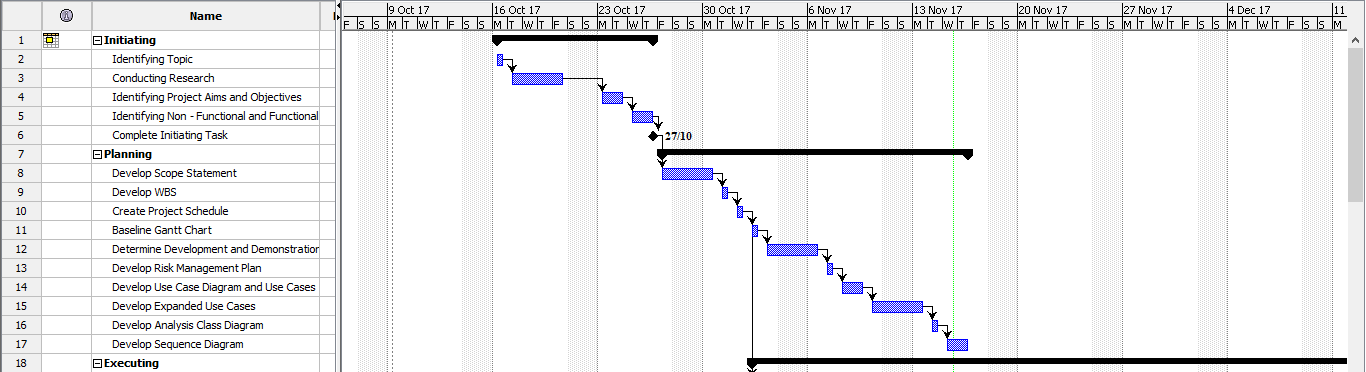
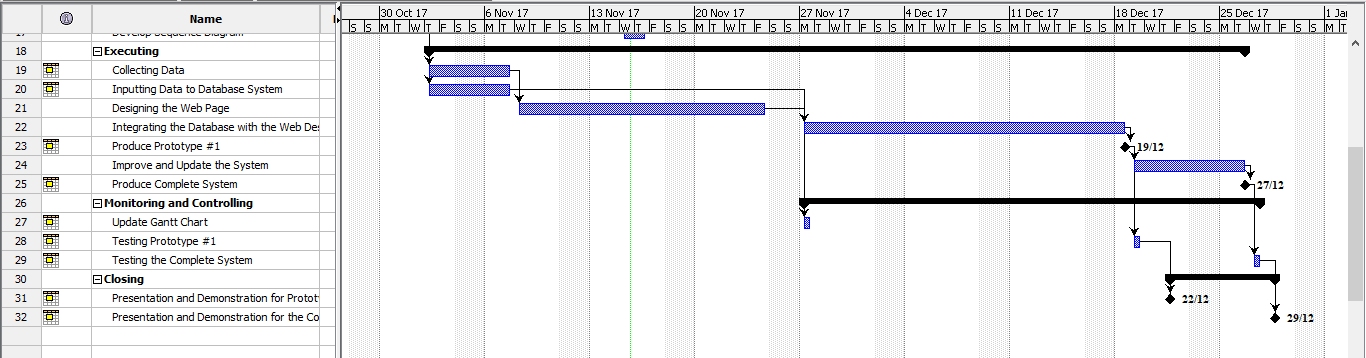
# Milestones

* Complete Initiating Task
* Produce the Prototype #1
* Presentation and Demonstration for Prototype #1
* Produce the Complete System
* Presentation and Demonstration for the Complete System

# Gantt Chart







# Development Platform

**Software/tools:**

1. **Microsoft Excel**

We will use Excel to store data and formula that we have collected before inputting them to database in MySQL.

1. **Microsoft Word**

We will use Word to produce documents and reports related to this project.

1. **Microsoft Power Point**

Power Point will be used to create presentation to show the design of our system and what went right or wrong during the development of the system.

1. **StarUML**

This is an open – source tool for Unified Modelling Language diagrams and modelling. We will use this tool to create use case diagram, class diagram and sequence diagram.

1. **MySQL**

We choose MySQL as our relational database management system because it is open – source and has many features to help us managing our database.

1. **Notepad++**

Notepad++ will be used to write our html codes to design the interface for our web application.

1. **phpMyAdmin**

**We choose this tool because it is open – source and it can be connected to our database in MySQL. phpMyAdmin** provides a convenient graphical user interface to work with and it also has all common functions that we need to develop a MySQL-based application or website.

1. **Adobe Photoshop**

Photoshop is suitable to manipulate and edit images that will be used in our web application design.

1. **Project Libre**

Project Libre is an open – source project management software that we use to create Gantt chart which help us in scheduling works for this project.

**Hardware:**

1. **Laptop**

All the for developing this application will be done using laptops that run on Windows operating system.

# Demonstration Platform

**Software:**

1. **Web browser – Google Chrome and Mozilla Firefox**

Because we are developing web based application, this application will be opened through a web browser. We choose Google Chrome and Mozilla Firefox to open our application because those are the most common browsers used nowadays. We also want to make sure our design and the data being displayed are consistent in both browsers.

**Hardware:**

1. **Laptop and PC**

Our application is intended to be opened through web browsers from laptop and personal computer.

# Risk Management Plan

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Risk Management Plan for Development of Web Based Academic Information System | | | | | | | | | | | |
| Prepared by: Lovely Date: Monday 30th October 2017 | | | | | | | | | | | |
| No | **Rank** | **Risk** | **Description** | **Category** | **Root Cause** | **Triggers** | **Potential Responses** | **Risk Owner** | **Probability (1 – 10)** | **Impact** | **Status** |
| 1 | 1 | Project would not be done on time | Team members could overwhelm by the tasks and might not finish them all on time | Process Risk | Not enough time to complete the project | One or more tasks might take longer than expected to complete | Follow the project schedule and do better time management to avoid tasks being unfinished after the due date | Lovely  Denny | 4 | High | This issue has not happened yet and so far tasks are being done on time. |
| 2 | 2 | There is possibility that the system will have some malfunction | The system might have some errors and crash, or cannot perform its function correctly | System Risk | There are errors during the development process | Team members do not check the system carefully during the testing stage | Test each function of the system thoroughly to ensure system crash would not happened | Lovely  Denny | 3 | High | This issue has not happened yet. |
| 3 | 3 | The team may not understand all the requirements to develop and complete the system | When requirements are not fully understood, the system that is produced may not as useful as what was planned | Process Risk | Lack of research and understanding about the subject | Not enough time to carry out research | Spare some times to conduct more research about similar application to learn about requirements that needed to be fulfill | Lovely | 3 | Medium | Team members have done more research and have understood what are the requirements to complete the system. |
| 4 | 4 | Lack of communication that may lead to misunderstanding between team members | Tasks that are not clearly divided and communicate may cause misunderstanding and conflict | People Risk | Each team member is busy with other activities | Have different schedules which make it difficult to have same spare time to meet and communicate about the project | Do online discussion if it is not possible to physically conduct a meeting, make a meeting schedule from the beginning of the project | Denny | 3 | Medium | This issue has not happened yet and so far team members communicate each other clearly. |

# References

[1] Worwa, K., Stanik, J. (2010, December). Quality of Web-based information systems. Journal of Internet Banking and Commerce, vol. 15, no.3. Retrieved from <http://www.icommercecentral.com/open-access/quality-of-webbased-information-systems.php?aid=38469>

[2] Web Based Application. Retrieved from <http://www.netsity.com/webbasedapplication.htm>

[3] Fuad, S. (2011). Information Systems: Definitions and Components [PDF]. Retrieved from <http://www.uotechnology.edu.iq/ce/Lectures/SarmadFuad-MIS/MIS_Lecture_3.pdf> (University of Technology – Iraq)

**BIT302**

**SOFTWARE ENGINEERING**

**Assignment 1**

**Requirements Specification Document**

**“Web-based Academic Information System”**

Team Leader: Lovely (E1500367)

Contact Details: lovelyzheng98@gmail.com

Member: I Wayan Denny Harry Caka (E1500365)

Contact Details: dennycaka@gmail.com

**Table of Contents**

[Functional Requirements 20](#_Toc498518191)

[Non – functional Requirements 20](#_Toc498518192)

[Use Case Diagram 21](#_Toc498518193)

[Use Cases 22](#_Toc498518194)

[Expanded Use Cases 24](#_Toc498518195)

[Analysis Class Diagram 35](#_Toc498518196)

[Tasks Division 36](#_Toc498518197)

# Functional Requirements

**User Requirements:**

1. Clear login menu with user ID and password column so that students can access the system without confusion, that will lead the students to student home page
2. Forget password option if a student forgot his/her password
3. The system should have menu with buttons/icons that students can choose from, e.g. a person icon that students can click to see his/her personal information
4. Button to change login password
5. Clear logout button for the student to exit the application

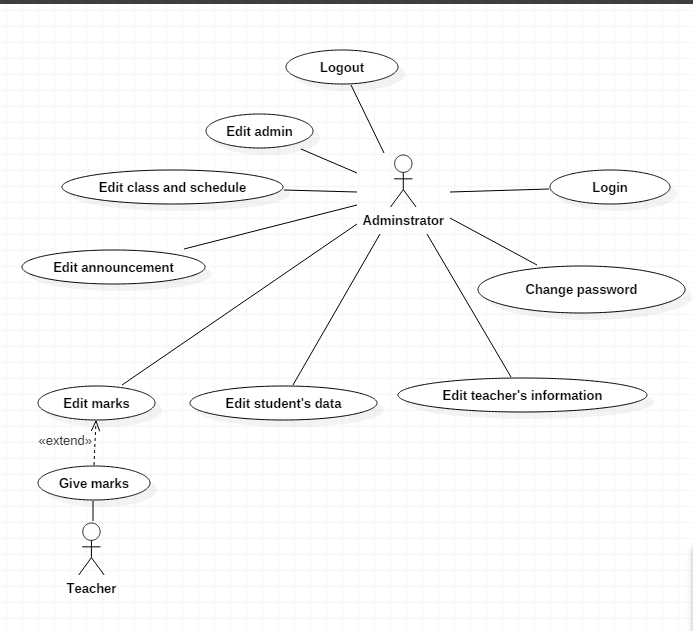
**Admin Requirements:**

1. Clear login menu with user ID and password column for admin to access the admin home page
2. Button to change login password
3. Admin page should have ‘edit menu’ to change data about students, teachers, marks and schedules
4. Admin should have ‘add menu’ to add new admin, students, teachers, marks, class and schedules
5. Admin should have ‘delete menu’ to delete an admin, student, teacher, announcement, mark and class
6. Menu to create announcement as a form that will be displayed in student home page

# Non – functional Requirements

1. Security requirement: using MD5 encryption for the password to make sure if someone without authorization can access our database, he/she can’t understand the password
2. Usability: the system should be easy to operate by admin and the features understood by students
3. Accuracy and precision: data entered by admin should be accurate and precise and free from error
4. Modifiability: data can only be modified by admin through admin page and students are only allowed to change his/her own password

# Use Case Diagram



# Use Cases

|  |  |  |
| --- | --- | --- |
| Requirement | Actor | Use Case |
| Login page has been created and functioned to allow admin and students to log in. | Student  Admin | Log in |
| Student and admin have been registered to the system and have had a login password | Student  Admin | Change password |
| The system provides a page for students to view their personal information and contact details | Student | View identity |
| The system has a page that shows contact details of all teachers in the school that students can view | Student | View teacher information |
| The system has a page that shows schedule based on the class of the student | Student | View schedule |
| A page for student to see their marks for home works, monthly tests, middle test, and final exam | Student | Check marks |
| Announcement that has been posted by admin will appear in student’s home page | Student | Check announcement |
| Teacher has given homework or test to students, and admin will input the marks to the system | Admin  Teacher | Edit marks |
| Students that have enrolled to the school will provide their personal data and contact details, and the information will be recorded in the system by admin | Admin | Edit student’s data |
| Information about teachers in the school will be recorded in the system by admin | Admin | Edit teacher’s data |
| Form to create announcement and it will be posted to student home page | Admin | Edit announcement |
| If there is a new class at the beginning of semester, new class and schedule will be created  If the class is already existed, the schedule will be updated at the beginning of the semester  If the class will no longer be needed, the class and schedule will be deleted at the beginning of the semester | Admin | Edit class and schedule |
| The system provides form to add or delete admin | Admin | Edit admin |
| The system provides log out button that will direct admin or student back to the login page | Admin  Student | Log out |

# Expanded Use Cases

1. Log in

|  |  |
| --- | --- |
| Use Case | Log in |
| Goal in Context | Allow admin and students to access the home page |
| Primary Actor  Secondary Actor | Admin and students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process begins when the student or admin enter their user ID and password, and presses the login button | 2. The system checks the information received |
|  | 3. The system displays the home page for the registered user or for admin. |
| Alternative Course | |
| Line 3: If the student or admin entered wrong ID or password, the system will show alert message to tell if the information being entered is incorrect and student or admin should enter their information again. | |

2. Change password

|  |  |
| --- | --- |
| Use Case | Change password |
| Goal in Context | Allow admin and students to change their login password |
| Primary Actor  Secondary Actor | Admin and students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process begins when the student or admin has been logged in |  |
| 2. Admin or student select the change password menu | 3. The system displays change password page with form to fill in the new password |
| 4. Admin or student has to fill in the form and type in the new password twice as validation |  |
| 5. Admin or student select the submit button | 6. The system will save the changes and update the database |
| Alternative Course | |
| Line 3: If two passwords being entered do not match, alert message will be shown and the new password cannot be submitted | |

3. View identity

|  |  |
| --- | --- |
| Use Case | View identity |
| Goal in Context | Allow students to access see their own personal information and contact details |
| Primary Actor  Secondary Actor | Students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Student has already accessed the home page and choose the personal information menu | 2. The system displays the personal information page |
| Alternative Course | |
| - | |

4. View teacher information

|  |  |
| --- | --- |
| Use Case | View teacher information |
| Goal in Context | Allow students to access teacher information page |
| Primary Actor  Secondary Actor | Students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Student has already accessed the home page and choose the teacher information menu | 2. The system displays the teacher information page |
| Alternative Course | |
| - | |

5. View schedule

|  |  |
| --- | --- |
| Use Case | View schedule |
| Goal in Context | Allow students to access schedule page |
| Primary Actor  Secondary Actor | Students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Student has already accessed the home page and choose the schedule menu | 2. The system displays the schedule page |
| Alternative Course | |
| - | |

6. Check marks

|  |  |
| --- | --- |
| Use Case | Check marks |
| Goal in Context | Allow students to check marks for home works or tests |
| Primary Actor  Secondary Actor | Students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Student has already accessed the home page and choose the marks menu | 2. The system displays the marks page |
| Alternative Course | |
| - | |

7. Check announcement

|  |  |
| --- | --- |
| Use Case | Check announcement |
| Goal in Context | Allow students to check announcement |
| Primary Actor  Secondary Actor | Students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Student has already accessed the home page and choose the announcement menu | 2. The system displays the newest announcement posted by admin |
| Alternative Course | |
| - | |

8. Edit marks

|  |  |
| --- | --- |
| Use Case | Edit marks |
| Goal in Context | Admin can input marks for students or edit the marks |
| Primary Actor  Secondary Actor | Admin  Teacher |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Teacher has checked a home work or test of students |  |
| 2. Admin select to the marks menu in the system | 3. The system displays the marks page |
| 4. Admin add new marks for student based on the subject and the type of the mark (homework or test or exam) | 5. The system saved the changes and update the information that will be displayed in the marks page |
| Alternative Course | |
| Line 4: Admin can choose to edit existing marks in the system | |

9. Edit student’s data

|  |  |
| --- | --- |
| Use Case | Edit student’s data |
| Goal in Context | Admin can input information of new students or edit information of existing students |
| Primary Actor  Secondary Actor | Admin  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Admin receives information about new students |  |
| 2. Admin select to the student menu in the system | 3. The system displays the student page |
| 4. Admin add new student to the database and his/her personal information and contact details | 5. The system saved the changes and update the information that will be displayed in the student page |
| Alternative Course | |
| Line 1: Admin receive information about existing student that want to change his/her data  Line 4: Admin edit the existing student’s data and update the database  Line 1: Admin receive information about student that will quit from the school  Line 4: Admin delete the student’s data and update the database | |

10. Edit teacher’s data

|  |  |
| --- | --- |
| Use Case | Edit teacher’s data |
| Goal in Context | Admin can input information of new teachers or edit information of existing teachers |
| Primary Actor  Secondary Actor | Admin  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Admin receives information about new teachers |  |
| 2. Admin select to the teacher menu in the system | 3. The system displays the teacher page |
| 4. Admin add new teacher to the database and his/her contact details | 5. The system saved the changes and update the information that will be displayed in the teacher page |
| Alternative Course | |
| Line 1: Admin receive information about existing teacher that want to change his/her data  Line 4: Admin edit the existing teacher’s data and update the database  Line 1: Admin receive information about teacher that will quit from the school  Line 4: Admin delete the teacher’s data and update the database | |

11. Edit announcement

|  |  |
| --- | --- |
| Use Case | Edit announcement |
| Goal in Context | Admin can input new announcement or edit an existing one |
| Primary Actor  Secondary Actor | Admin  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Admin receives information about new announcement |  |
| 2. Admin select to the announcement menu in the system | 3. The system displays the announcement page with form that admin can fill in |
| 4. Admin fill in the form and press the submit button | 5. The system will post the announcement to the student home page and admin announcement page |
| Alternative Course | |
| Line 1: Admin receives information to change some information from announcement that previously has been posted  Line 4: Admin delete the existing announcement before fill in the form and create a new one | |

12. Edit schedule

|  |  |
| --- | --- |
| Use Case | Edit class and schedule |
| Goal in Context | Admin can input new schedule at the start of the semester or edit the schedule if there is mistake when adding |
| Primary Actor  Secondary Actor | Admin  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Admin receives information about new schedule for existing class |  |
| 2. Admin select to the schedule menu in the system | 3. The system displays the schedule page |
| 4. Admin add new schedule based on the class | 5. The system saved the changes and update the information that will be displayed in the schedule page |
| Alternative Course | |
| Line 1: Admin receives information that a new class is to be created with its schedule  Line 4: Admin add new class and add new schedule based on the new class  Line 1: Admin receives information to delete a class and its schedule  Line 4: Admin delete the class and its schedule | |

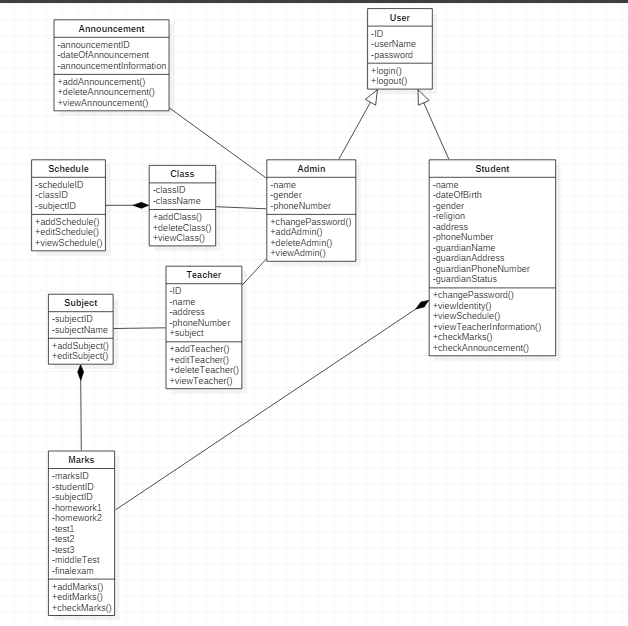
13. Edit admin

|  |  |
| --- | --- |
| Use Case | Edit admin |
| Goal in Context | New admin can be added to the system or delete existing one |
| Primary Actor  Secondary Actor | Admin  - |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. Existing admin receives information about new admin |  |
| 2. Admin select the add admin menu in the system | 3. The system displays the new admin page |
| 4. Admin add new admin to the database and his/her ID and password | 5. The system saved the changes and update the information that will be displayed in the existing admin page |
| Alternative Course | |
| Line 1: Admin receive information about admin that will quit from the school  Line 4: Admin delete the admin’s data and update the database | |

14. Logout

|  |  |
| --- | --- |
| Use Case | Log out |
| Goal in Context | Allow admin and students to exit the home page |
| Primary Actor  Secondary Actor | Admin and students  **-** |
| Typical Course of Events | |
| Actor Actions | **System Response** |
| 1. The process begins when the student or admin have finished what they need to do with the system |  |
| 2. Student or admin select the log out button | 3. The system displays login page |
| Alternative Course | |
| - | |

# Analysis Class Diagram



# Tasks Division

|  |  |
| --- | --- |
| Required Behaviour | Member’s Name |
| HTML and CSS script for designing the website menu and navigation | Lovely |
| PHP script for login | Lovely |
| PHP script for add new admin, delete and admin | Lovely |
| PHP script to change password | Denny |
| HTML and CSS script to design how data will be displayed when being viewed | Denny |
| PHP script for edit student, teacher, marks, class and schedules | Lovely and Denny |
| PHP script for search column in admin home page | Lovely |
| PHP script for posting announcement from admin to student home page | Denny |