**FACULTY OF COMPUTING**

**BCS3153 SOFTWARE**

**EVOLUTION AND MAINTAINANCE**

**MINI PROJECT REPORT**

**Club Management System (CMS)**

**LECTURER:**

DR. HASAN KAHTAN KHALAF

**SECTION:**

01

**PREPARED BY:**


| | |
|---|---|
| CB16165 | Robiatul Adawiyah Binti Sapia'i |
| CB16164 | Nurfarzana Binti Rosli |
| CB16169 | Nurul Adilah Binti Suliman |
| CB16167 | Muhammad Danish Bin Abdul Wahab |
| CB15166 | Nabil Bin Lutfi |

## Contents

**Table of Figures**

## Checklist

**Classification Software Maintenance**

| Intention-Based Classification Software Maintenance | | Status | Person in Charge |
|---|---|---|---|
| Corrective Maintenance | | √ | Farzana, Danish |
| Adaptive Maintenance | | x | |
| Perfective Maintenance | | √ | Farzana, Danish, Adilah, Robiatul, Nabil |
| Preventive Maintenance | | x | |
| Activity-Based Classification Software Maintenance | | Status | |
| Correction Maintenance | | √ | Farzana, Danish |
| Enhancement Maintenance | | √ | Farzana, Danish, Adilah, Robiatul, Nabil |
| Evidence-based Classification Software Maintenance | | Status | |
| Business Rules | Enhancive Maintenance | √ | Adilah, Robiatul, Nabil |
| | Corrective Maintenance | √ | Farzana, Danish, Adilah, Robiatul, Nabil |
| | Reductive Maintenance | x | |

| | | | |
|---|---|---|---|
| Software Properties | Adaptive Maintenance | x | |
| | Performance Maintenance | x | |
| | Preventive Maintenance | x | |
| | Groomative Maintenance | x | |
| Documentation | Updative Maintenance | x | |
| | Reformative Maintenance | x | |
| Support Interface | Evaluative Maintenance | x | |
| | Consultive Maintenance | x | |
| | Training Maintenance | x | |

**GitHub Operation**

| Task | Status |
|---|---|
| Create repository | √ |
| Add collaborator (team member) | √ |
| Delegate module accordingly | √ |
| Create branch | √ |
| Make changes to code | √ |
| Proposed change request | √ |
| Review change and approve/reject change | √ |
| Create Workflow (CI file) | √ |
| Merge file | √ |
| Update dashboard | √ |

# Chapter 1

# Introduction

## 1.1 Background

Software Evolution is a term which refers to the process of developing software initially, then timely updating it for various reasons for example, to add new features or to remove obsolete functionalities. The evolution process includes fundamental activities of change analysis, release planning, system implementation and releasing a system to customers. According to Lowell Jay Arthur, software evolution means a continuous change from lesser, simpler, or worse state to a higher or better state.

The cost an impact of these changes is accessed to find out how much system is affected by the change and how much it might cost to implement the change. If the proposed changes are accepted, a new release of the software system is planned. During release planning, all the proposed changes such as fault repair, adaptation, and new functionality are considered.

Software Maintenance is the process of modifying a software product after it has been delivered to the customer. The main purpose of software maintenance is to modify and update software application after delivery to correct faults and to improve performance. Based on the term used by Keith H. Bennett and Lie Xu, maintenance for all post-delivery support and evolution to those driven by changes in requirements.

Maintenance can be divided into several categories which is Corrective maintenance, Adaptive maintenance, Perfective maintenance and Preventive maintenance. Software Maintenance must be performed in order to correct faults and improve the design of the system. Besides that, implementation of enhancements is important to ensure that the modification works. Furthermore, to accommodate programs so that different hardware, software, system features, and telecommunications facilities can be used. Maintenance can also be used to migrate legacy software and retire a software.

The system that will be used in this project is Club Management System. There are two users are involved in this system which is the members and admin. Members can sign up and login to see the latest announcement and publish advertisement. Admin will manage the financial and the members in the system.

## 1.2 Objective

There are several objectives in this project:

1. To understand Software Evolution and Maintenance concepts
2. To implement Software Evolution and Maintenance concepts into the system
3. To use and implement the Software Evolution and Maintenance tools into Club Management System.

## 1.3 Scope

Table 1.1 Scope of Project

| **Scope** | **Description** |
|---|---|
| Users of the system | Member<br>Admin |
| Software and Maintenance Tool | GitHub<br>Git Bash<br>Sublime Text 3<br>Notepad++<br>Google Cloud Platform<br>XAMPP<br>PhpMyAdmin with Designer |
| System Platform | Windows 64 bit |

## 1.4 Project Overview

In this project consists of seven chapters including Introduction, Taxonomy, Evolution and Maintenance Models, Reengineering, Impact Analysis, Tools, and Result & Discussion.

Chapter 1 is the Introduction, this chapter is the introduction of this project that includes the background of the project, the project objectives, project scope and the project overview.

Chapter 2 is Taxonomy; this chapter describes on the taxonomy software evolution and maintenance concept that will be used in this project.

Chapter 3 is Evolution and Maintenance Models; this chapter is used to explain on the models that and Evolution Models related to software maintenance and evolution and describes models will be implemented in this project.

Chapter 4 is Reengineering; this chapter is a brief of the reengineering concept of software maintenance and evolution. The reengineering process involved in this project are described here.

Chapter 5 is Impact Analysis; this chapter is used to describe the impact analysis of this project which describes the impact of changes that will affect the project.

Chapter 6 is Tools; this chapter explains the tools used in software maintenance and evolution.

Chapter 7 is Result & Conclusion; this chapter covers the difference between the before and after modified system with the change of request.

# Chapter 2

# Taxonomy

## 2.1 Classification of Software Maintenance

### 2.1.1 Intention-based Classification of Software Maintenance

i. Corrective Maintenance

Corrective maintenance is about correcting processive and performance failures. Refer result in Chapter 7

ii. Perfective Maintenance

Perfective maintenance is about making a variety of improvements to the user experience, processing efficiency, and maintainability

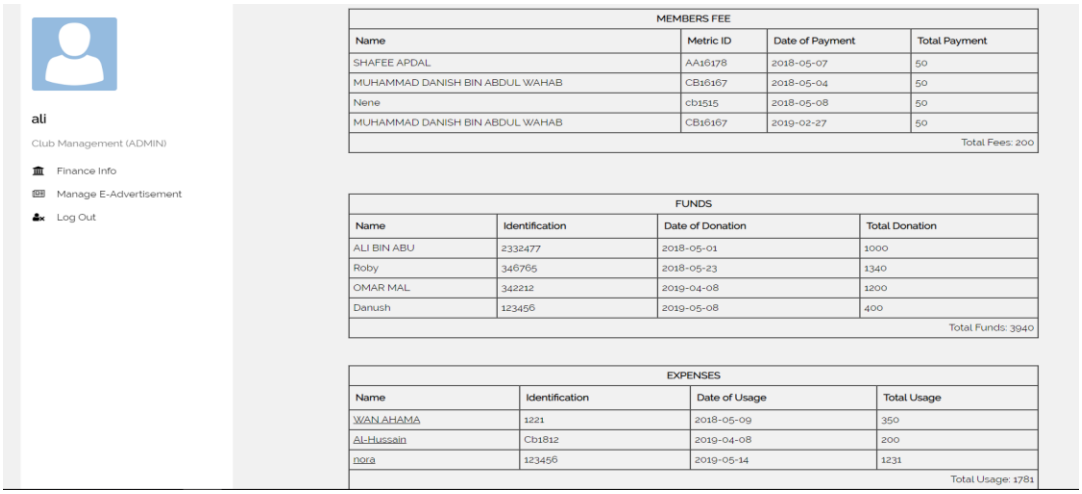


Figure 2.1 Before Enhancement

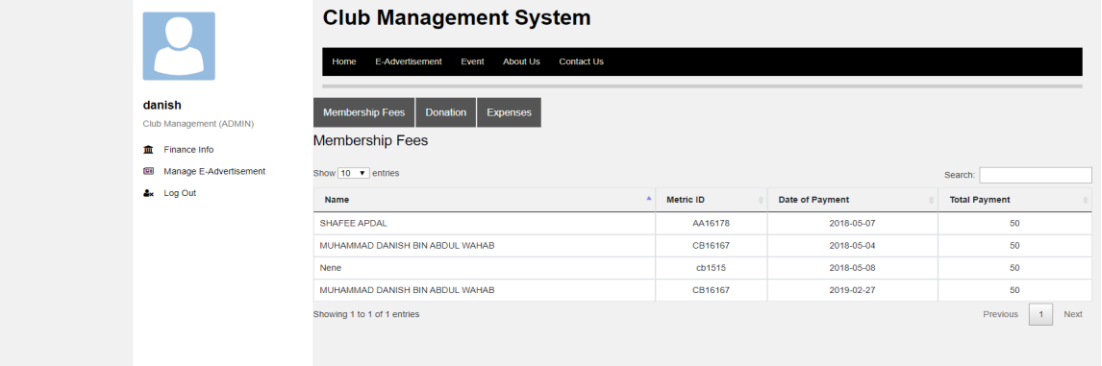


Figure 2.2 After Enhancement

### 2.1.2 Activity-based Classification of Software Maintenance

i. Corrections

Involves activities that are designed to fix defects in the system.

ii. Enhancements

Involves activities that are designed to effect changes to the system without modifying the behavior of the system.

### 2.1.3 Evidence-based Classification of Software Maintenance

#### 2.1.3.1 Business Rules

i. Enhancive

Involves adding and modifying business rules to enhance the system functionality.

ii. Corrective

Involves correcting identified bugs, adding defensive programming strategies and modifying exception handling.

#### 2.1.3.2 Software Properties

i. Groomative

Involves replacing components and algorithms with something more efficient and simpler.

## 2.2 Maintained Product

### 2.2.1 Product

The product to maintain for the Club Management System is the system functionality, system performance and the user interface of the system. Therefore, we maintain the system coding by fixing the bugs and remove unwanted coding. The example of product is shown in Chapter 4. Figure below shows the interface of the maintained product.

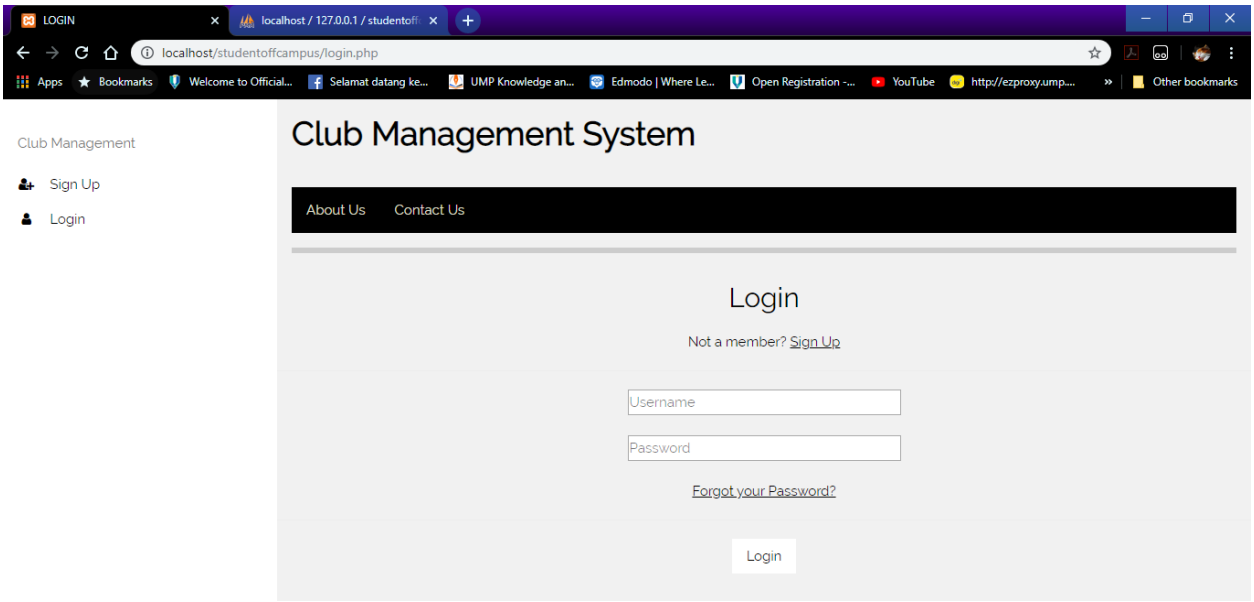


Figure 2.3 Homepage of CMS

### 2.2.2 Product Upgrade

There are two upgrades will be applied in Club Management System. We will upgrade the system functionality by adding two features in the system. Therefore, the functionality that we will add are the manage event and manage advertisement features.

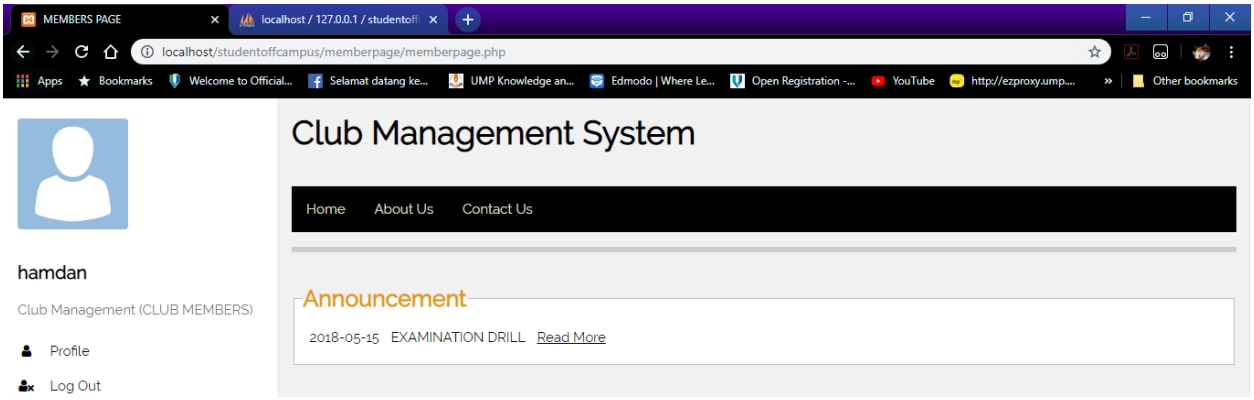


Figure 2.4 Before Adding Features

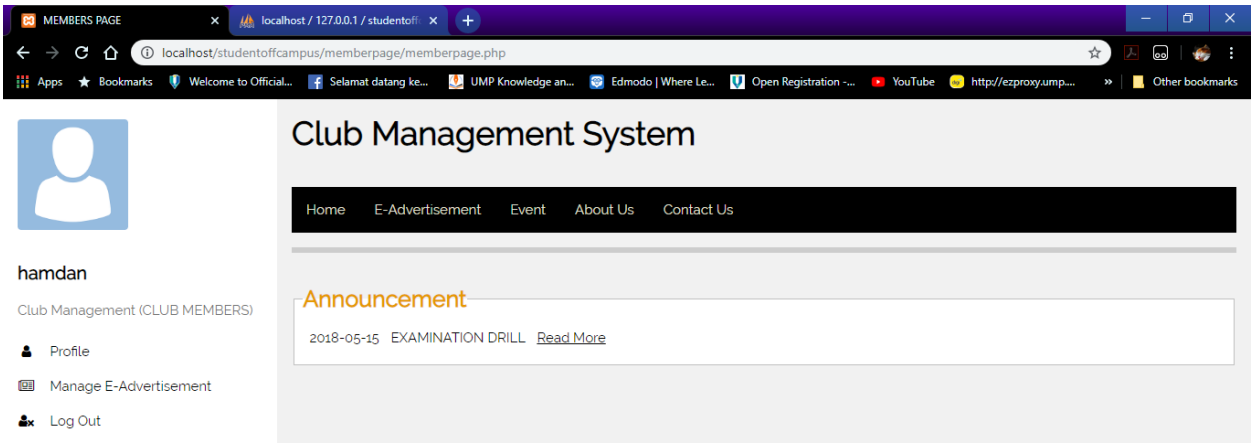


Figure 2.5 After Adding Features

### 2.2.3 Artefact

The artefacts that used to maintain the Club Management System is the CMS Software Requirement Specification (SRS) document and CMS Software Design Documentation (SDD) document.

# Chapter 3

# Evolution & Maintenance Model

## 3.1 Change Mini-Cycle Model

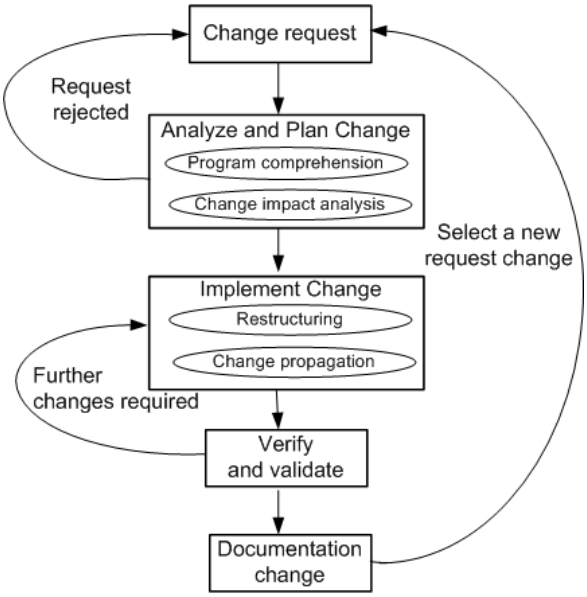


Figure 3.1 Change Request Flow

A system may have changed after being deployed during the development. The system may need to change according to the user expectation. There are five stages of change request which are change request, analyses and plan change, implement change, verify and documentation. Software Configuration Management (SCM) is tasked on changing the system. In Club Management System, the developer will have a change request to add new features or further enhance the system. The team follow this workflow where the leader alert the issue of what needs to be change in GitHub and other team members will analyse and plan on what is the best way to make the changes requested. Developer implement the changes into the Club Management System and verify it with the whole team. After successfully making the changes towards the system, the team will need to document the changes for future reference.

**Change Request:** For the initial stage of the Change Mini-Cycle model, the team receive the change through a form called the Change Request Form. The form must be filled in with the name of whom request the change, the priority of the change whether it should be priorities or not. Besides that, they will also have to select the category and fill in the reason for the change. The request is then will be reviewed by the team. The request form can be view in Chapter 7 Result and Discussion.

**Analyze the impact of change:** After the request has been made, the leader will request the Change Request Form in order to analyze the impact of change and the cost for the change. If the request does not meet the criteria of the system, then the team leader could reject the request.

**Implement Change:** Once the request has been analyzed by the leader, the team can proceed by making the changes. When the changes are complete, a Pull Request is made within the GitHub to implement the changes of the system. The figure below shows the Pull Request within GitHub for implementation of a feature for Club Management System.

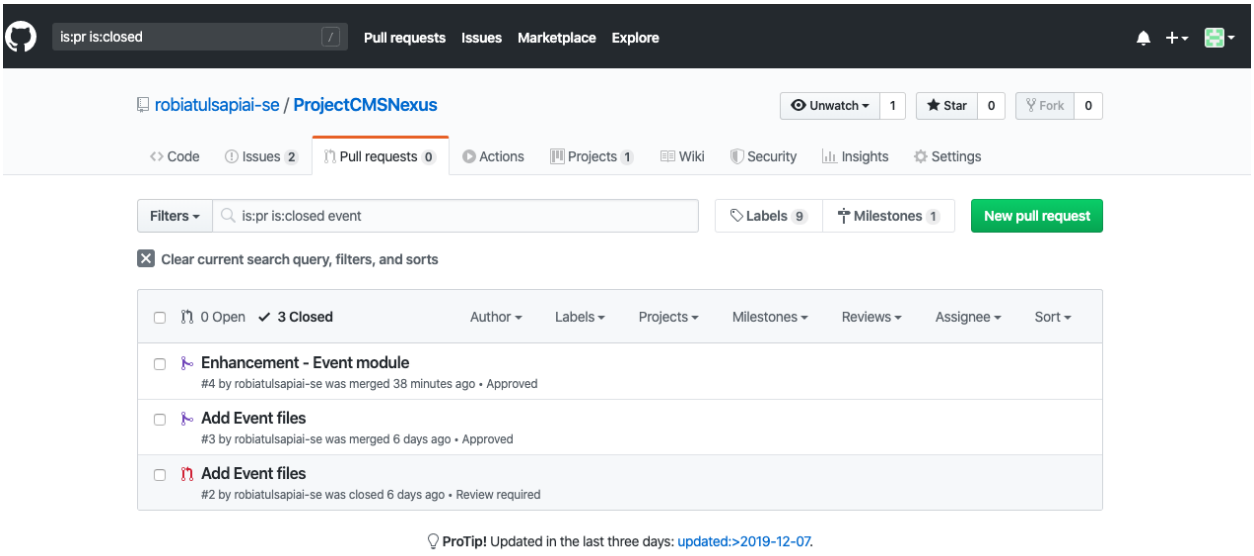


Figure 3.2 Pull Request within GitHub

**Verify and Validate:** After the pull request has been made, the leader will then verify the pull request by reviewing the commit or changes in the commit history to ensure that the implementation of the changes is done correctly. After ensuring that the implementation will work, and no problem will occur. The figure below shows the list of changes committed within GitHub.

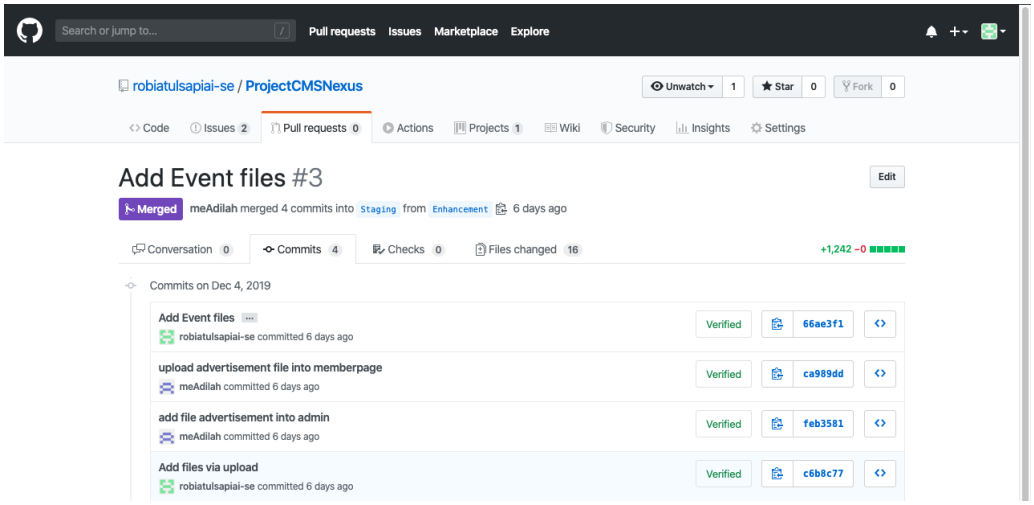


Figure 3.3 Changes Committed within GitHub

The leader will then submit a review to approve the change request and conduct testing for the system to ensure that the changes are working as intended. The figure below shows the review that will be submitted by the leader to approve the request.

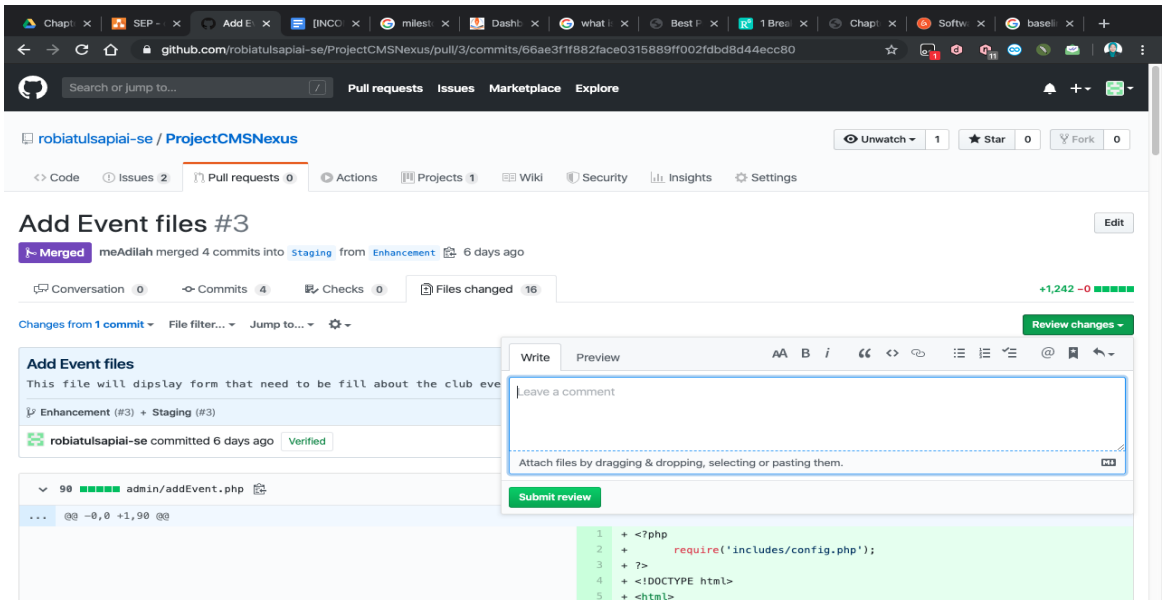


Figure 3.4 Review Submitted within GitHub

## 3.2 Maintenance Process

### 3.2.1 Problem Identification Stage

In this phase, the user submits their feedback after using the system. This is where a modification request will be sent to the team. After receiving the modification request, the team will identify and estimate the resources needed for the change to take place. The modification request could be accepted or rejected in this phase.

One of the feedbacks received from the user is within the financial report which is the table displaying the members fees, donations and expenses are not design in a good way for the user experience. Furthermore, the data organization is becoming longer and longer as the data is inputted in the system. The user requests the design of the table to be more organized for the user experience. The team decided to accept the modification request in order to make the user interface according to the user expectation. The figure below shows the problem that the user request to modify.

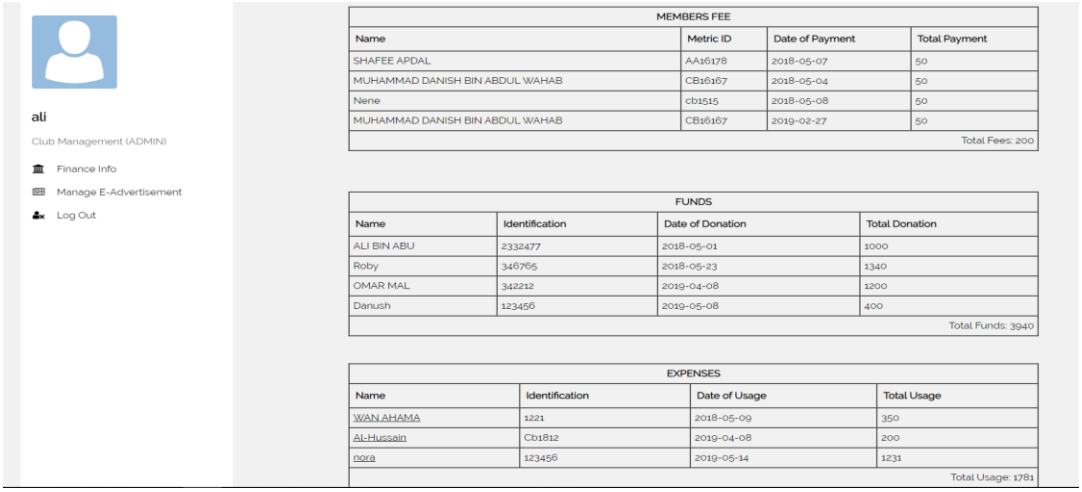


Figure 3.5 Problem Identified within the System

### 3.2.2 Analysis Phase

There are two major activities in this phase which are:

**Feasibility analysis:** The focus of this analysis is more to the business. The team will identify the impact of changes towards it will affect the system. The estimation cost in short-term and long-term will be calculated to ensure the worthiness to implement this change into the system. Figure the possible solutions to do the changes.

**Detailed analysis:** It will focus more on the technical where the team will evaluate how the changes will impact the system. Check the modification requirement and the component that involved towards the changes and need to have a testing strategy and implementation plan to ensure the changes implemented correctly.

Based on the problem as stated above, the team analyze the modification request of the tables. The first activity the team did is the Feasibility Analysis. From the change request form we analyze the impact of changes towards it will affect the system. Based on the analysis, the team come out with a result which is the modification request is just an interface enhancement that will not affect the system. The cost of improvement will only take a small amount to conduct.

The second activity conducted is the Detailed Analysis where the technical team create a plan to implement the modification of the table in the financial report. Besides that, the testing strategy is also planned to ensure that the modification is done correctly.

### 3.2.3 Design Phase

In this phase the team will first identify software component that will be affected in this change and design the modification changes. The involved component is important to prevent any error or problem during the progress of modification. The modification changes are then documented in the artefact and the team needs to create test suite for this new change that has been done and select test cases for regression testing.

For the modification of the table, we analyze the code. Most of the table are connected to the database which display all the financial data in the financial report. Thus, this may cause a problem during the modification as there is a possibility that some errors in displaying the data may occur. A test suite is created to conduct the test of the modification. The figure below shows the coding that requires modification that involves the database.

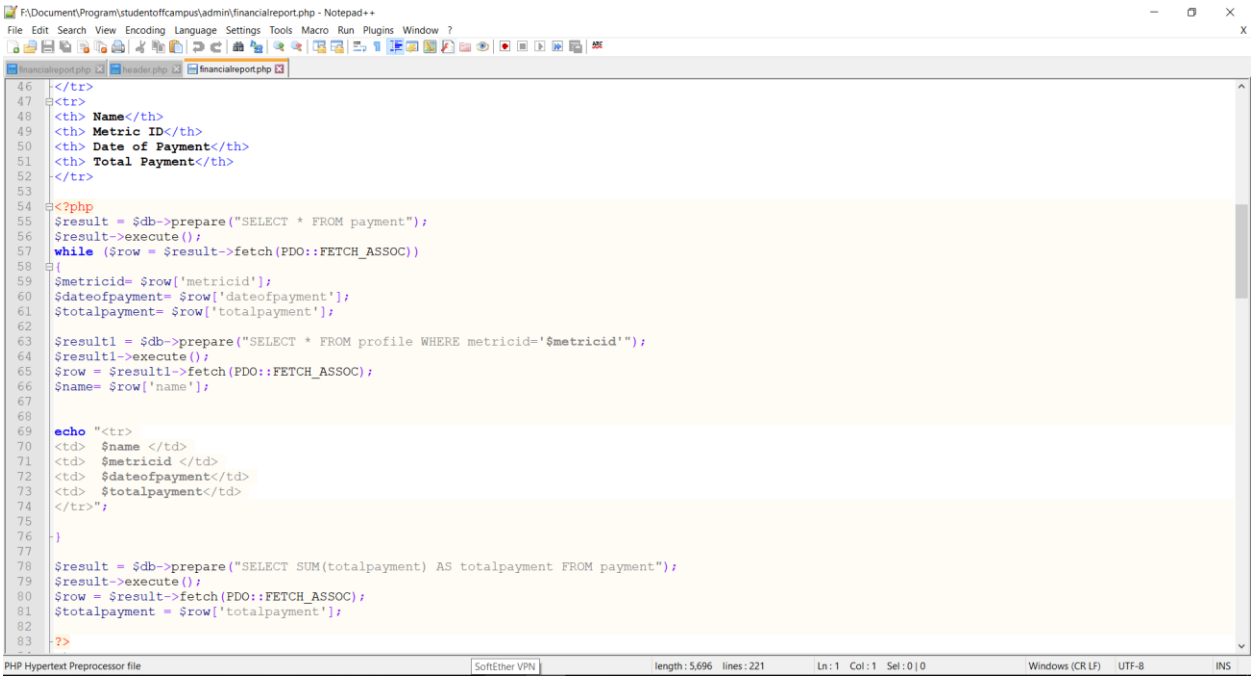


Figure 3.6 Code that is Analyze

### 3.2.4 Implementation Phase

Implementation of changes of code will take place in this phase. The changes made will be tested in unit testing whether it will work or not. The modified code will be integrated into the system and testing will be conducted. Risk analysis and review will need to be conducted for the system for test-readiness.

In the phase, the team implemented the changes that was requested. The modification links the system with few Bootstraps, JQuery and DataTable to improve the tables. Besides that, we include few interface modifications in the codes to make the interface more organized. The figures below show the code modifications of the system.

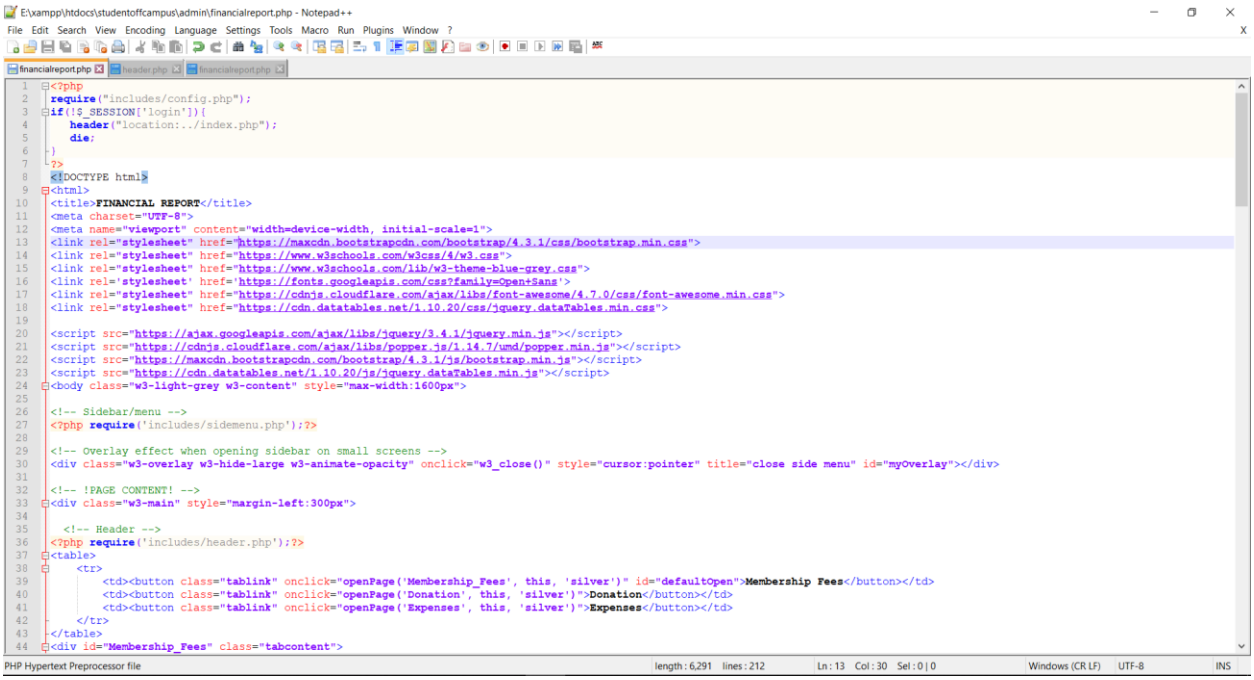


Figure 3.7 Modification of Financial Report 1

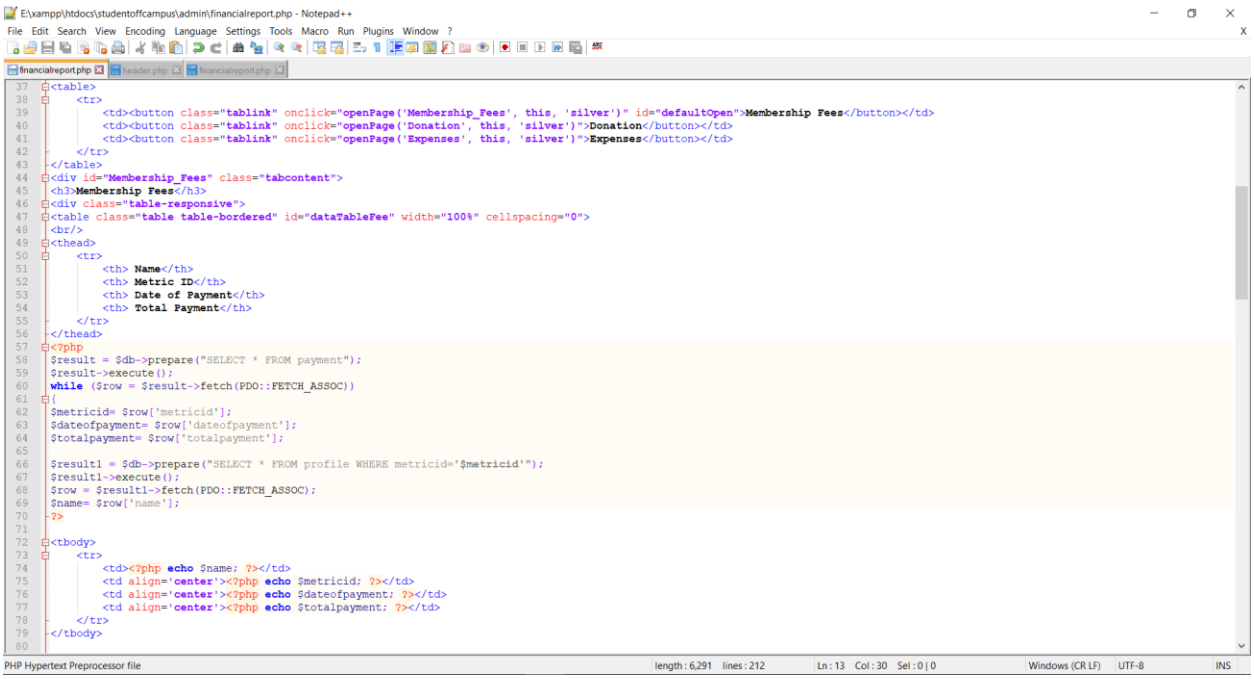


Figure 3.8 Modification of Financial Report 2

### 3.2.5 System Test Phase

System test phase will help conduct the testing which are functionality testing, robustness testing, stability testing, load testing, performance testing, security testing and regression testing.

We conduct testing for the modification of the system. So far, the testing went as intended and the modification is successfully implemented based on the modification request. The figure below shows the result of the modification that was requested.

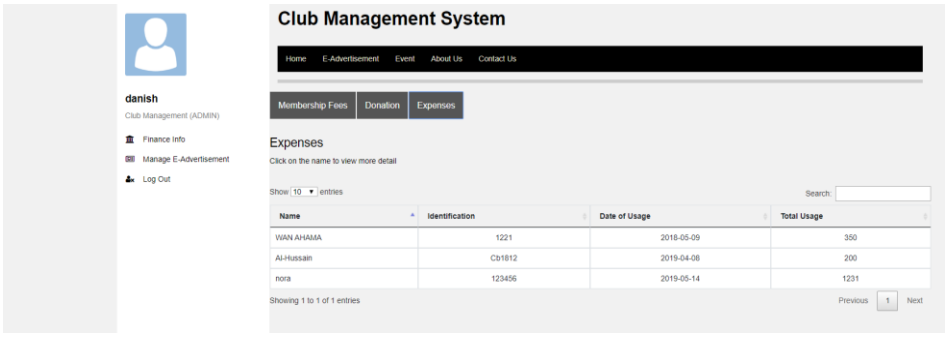


Figure 3.9 Result of Modification 1

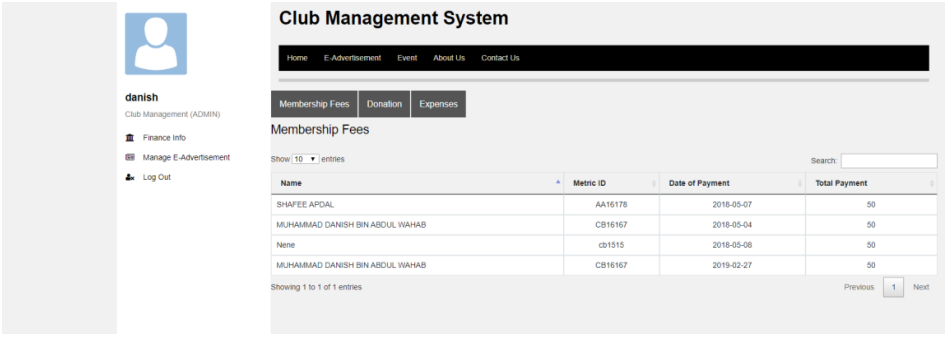


Figure 3.10 Result of Modification 2

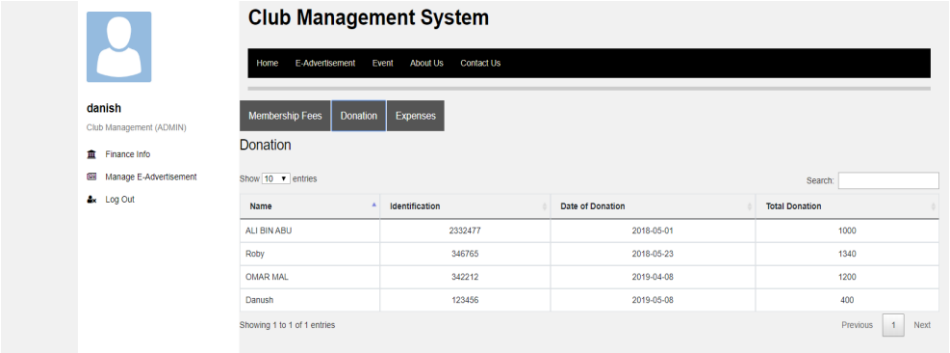


Figure 3.11 Result of Modification 3

# Chapter 4

# Reverse Engineering

## 4.1 Implementation

The system was developed by using PHP, HTML and JavaScript languages and CSS as the stylesheet. This system uses XAMPP web server as a platform to connect with the MySQL database, which is phpMyAdmin. Figure below shows an example of coding implementation that adding the new features.

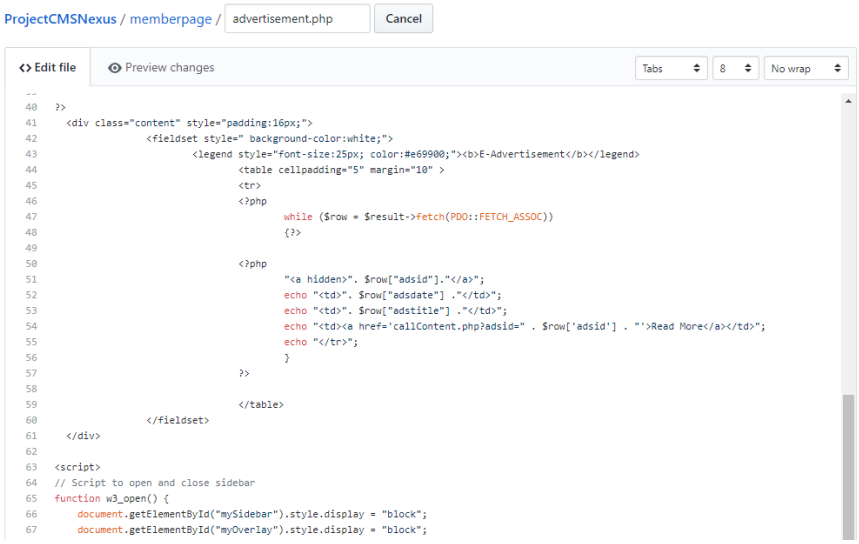


Figure 4.1 Coding Implementation

## 4.2 Design

We design the database of the system and the characteristics of the system. The Club Management System is a structured system. At first, the system only consists of six tables. Then, we add new features, which are advertisement and event table. Therefore, there are eight tables in the CMS database, which are members, profile, advertisement, announcement, donation, event, expenses and payment.

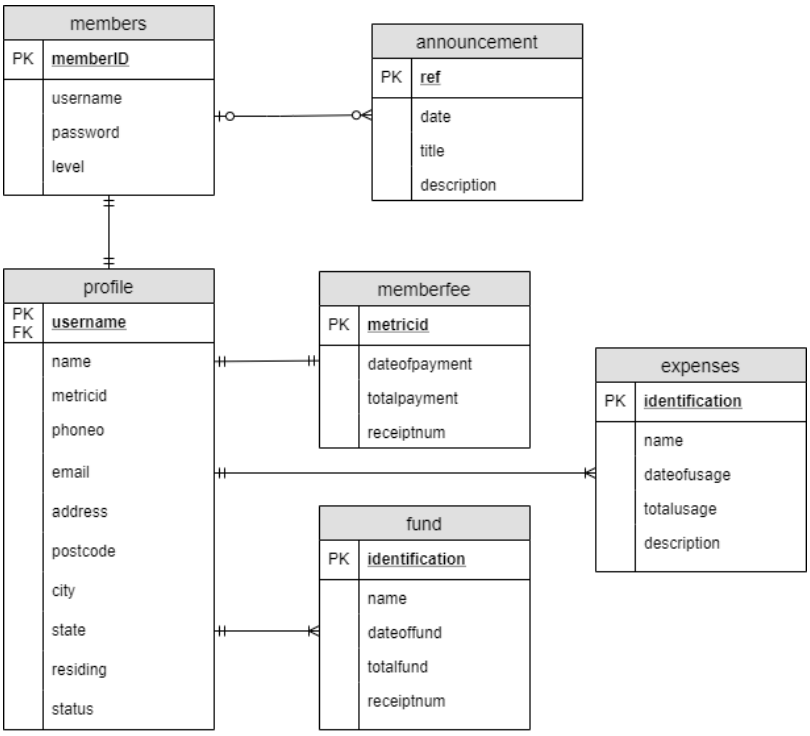


Figure 4.2 ERD Diagram Before

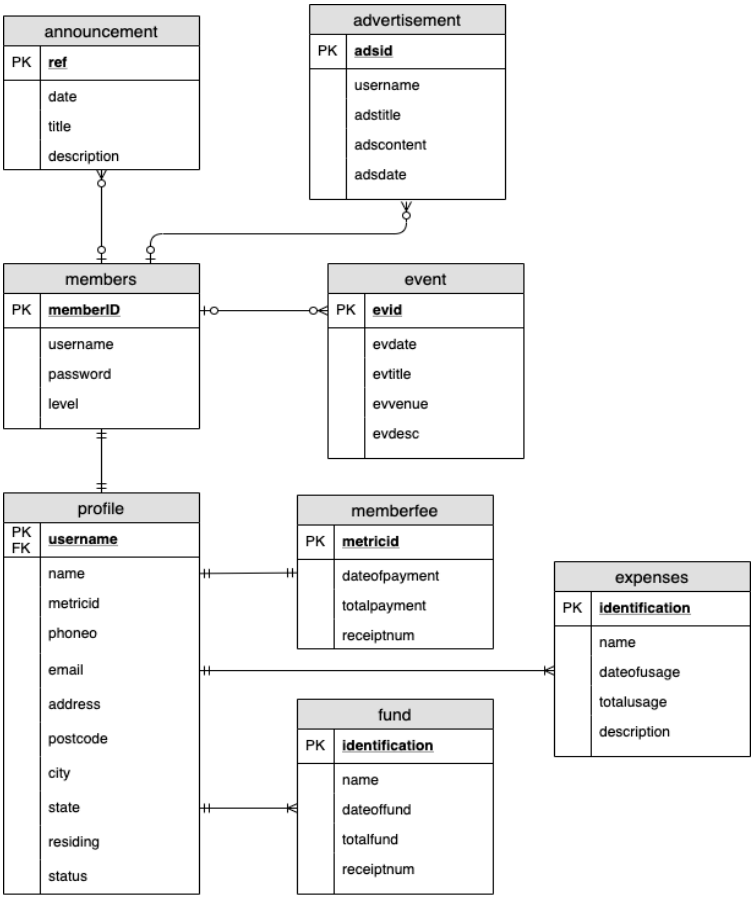


Figure 4.3 ERD Diagram After

## 4.3 Requirements

The features for the Club Management System is increasing. Therefore, the change request is applied, and changes is made at the system documentation. The new features are the advertisement module and event module. The module consists of CRUD properties which are the system able to create, edit, update and delete the advertisement and events.

## 4.4 Conceptual

The purpose of the changes is made to make the club management more organized by separating the advertisement and events from the announcement module.

# Chapter 5

## Impact Analysis

In this chapter, the Impact Analysis is to define the impact of changes in the deployed product or application. It gives the details about the areas of the system that may be affected due to the change in the section or features of the application. The Change Request (CR) are made to make a change to the system. The changes include add, delete or modify the functional or requirements of the system.

For CMS project, the form of Change Request will be performed by the developer of the system. Then, the impact analysis will be analyzing because there might be test cases, issues, and even source code impacted. In CMS project, the changes that has the highest impact towards the system are by adding new features which are Manage Event and Advertisement. This is because the new features will impact the Manage Announcement as the type of announcement is divided.

To perform the changes of system, a form of request is made to get the details of changes. This form contain impact analysis and Change Request of the system are shown in Table.

| Change Request Form | | | |
|---|---|---|---|
| Project Name | Club Management System (CMS) | | |
| Requested By | Nurfarzana | Date | 4/12/2019 |
| Name of Request | Description of announcement | Request ID | CMS_CR_01 |
| | | Priority | High |
| Change Category | □ Schedule<br>□ Cost<br>□ Scope<br>√ Requirements/Deliverables<br>√ Testing/Quality<br>□ Resources | | |
| Change Reason | - The textbox of Description should be in <textarea> input as the input is long.<br>- The description of announcement is not shown in table. | | |
| Impact of Change | □ Corrective Action<br>□ Preventative Action<br>Enhancive Action<br>□ Evolution<br>√ Defect Repair<br>□ Updates<br>□ Other | | |
| Proposed Action | Project Manager accepted the change request as it will enhance the system's module. There is no risk toward the changes. | | |
| Status | In Review | Approve | Rejected |
| | | / | |
| Approval Date | 6/12/2019 | | |
| Approved By | Robiatul Adawiyah | | |

| Change Request Form | | | |
|---|---|---|---|
| Project Name | Club Management System (CMS) | | |
| Requested By | Nurul Adilah | Date | 4/12/2019 |
| Name of Request | Add feature advertisement in the system | Request ID | CMS_CR_02 |
| | | Priority | High |
| Change Category | □ Schedule<br>□ Cost<br>□ Scope<br>√ Requirements/Deliverables<br>□ Testing/Quality<br>□ Resources | | |
| Change Reason | To create a medium for the committee member to promote their advertisement | | |
| Impact of Change | □ Corrective Action<br>□ Preventative Action<br>□Enhancive Action<br>√ Evolution<br>□ Defect Repair<br>□ Updates<br>□ Other | | |
| Proposed Action | Project Manager accepted the change request as it will enhance the system's module. There is no risk toward the changes. | | |
| Status | In Review | Approve | Rejected |
| | | / | |
| Approval Date | 6/12/2019 | | |
| Approved By | Robiatul Adawiyah | | |

| Change Request Form | | | |
|---|---|---|---|
| Project Name | Club Management System (CMS) | | |
| Requested By | Muhammad Danish | Date | 4/12/2019 |
| Name of Request | Change the tables to be more organized for the fees, donations and expenses. | Request ID | CMS_CR_03 |
| | | Priority | Low |
| Change Category | □ Schedule<br>□ Cost<br>□ Scope<br>√ Requirements/Deliverables<br>□ Testing/Quality<br>□ Resources | | |
| Change Reason | The tables will be more organized and better for the user experience. | | |
| Impact of Change | □ Corrective Action<br>□ Preventative Action<br>√ Enhancive Action<br>□ Evolution<br>□ Defect Repair<br>□ Updates<br>□ Other | | |
| Proposed Action | Project Manager accepted the change request as it will enhance the system's module. There is no risk toward the changes. | | |
| Status | In Review | Approve | Rejected |
| | | / | |
| Approval Date | 6/12/2019 | | |
| Approved By | Robiatul Adawiyah | | |

| Change Request Form | | | |
|---|---|---|---|
| Project Name | Club Management System (CMS) | | |
| Requested By | Robiatul Adawiyah | Date | 4/12/2019 |
| Name of Request | Add new feature which is manage event | Request ID | CMS_CR_04 |
| | | Priority | High |
| Change Category | □ Schedule<br>□ Cost<br>□ Scope<br>√ Requirements/Deliverables<br>□ Testing/Quality<br>□ Resources | | |
| Change Reason | To make it more efficient to differentiate between type of announcement which may include event or advertisement | | |
| Impact of Change | □ Corrective Action<br>□ Preventative Action<br>√ Enhancive Action<br>□ Evolution<br>□ Defect Repair<br>□ Updates<br>□ Other | | |
| Proposed Action | Project Manager accepted the change request as it will enhance the system's module. There is no risk toward the changes. | | |
| Status | In Review | Approve | Rejected |
| | | / | |
| Approval Date | 6/12/2019 | | |
| Approved By | Robiatul Adawiyah | | |

# Chapter 6

# SCM Practice and Tools

## 6.1 SCM Practice

There are four SCM Practice that were applied in this project. Which are:

1. Configuration Identification: Configuration identification is a method of determining the scope of the software system. Activities during this process:

    a. Identification of configuration items such as source code modules and requirements specification. Figure 6.1 below shows the source code that were uploaded in the GitHub. The codes were categorized into folders which separate the files for members and admin for CMS. Figure 6.2 shows the SRS document as our reference for system requirement changes.

    b. The process starts with basic objects which are grouped into aggregate objects. Details of what, why, when and by whom changes in the test are made. This part is shown in Chapter 5 (Change Request).

    c. Identify tools to be used in this project. The summary of the tool is depicted as follows:

Table 6.1 Tools Used in the Project

| Tool | Function |
|---|---|
| GitHub | A software that act as the version control to manage and maintain the code |
| Atom | A software that act as IDE that is used to code and maintain the system |
| XAMPP | A software that act as a server to host the system |
| phpMyAdmin | Software tool written in PHP, intended to handle the administration of MySQL over the Web |

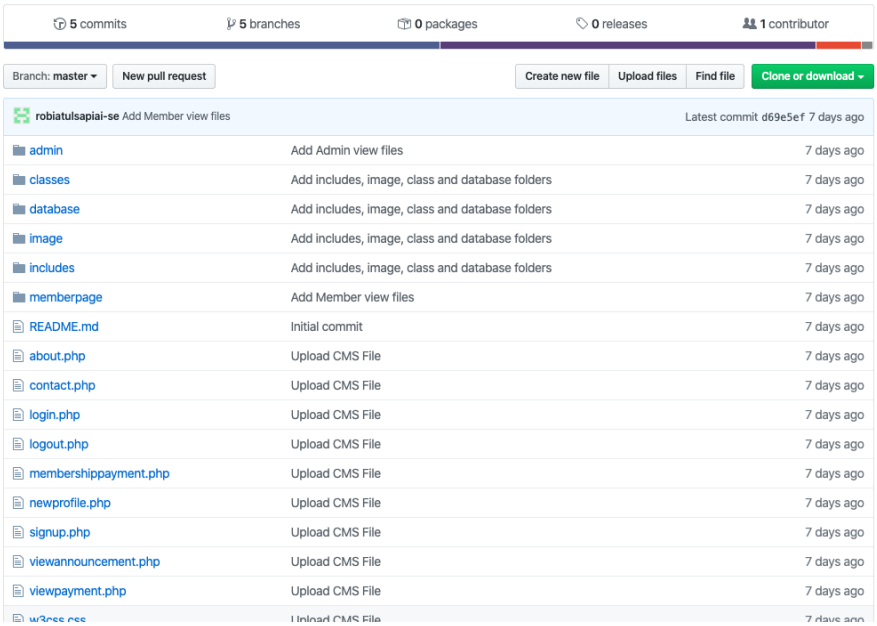


Figure 6.1 Codes were Divided into Folders.


Figure 6.2 SRS of CMS

2. Change Control: Change control is a procedural method which ensures quality and consistency when changes are made in the configuration object. In this step, the change request is submitted to project leader. Activities during this process:

   a. Control ad-hoc change to build stable software development environment. Changes are committed to the repository. Few branches were created to cater ad-hog changes. Figure below shows the branches that were used in this project and its function.

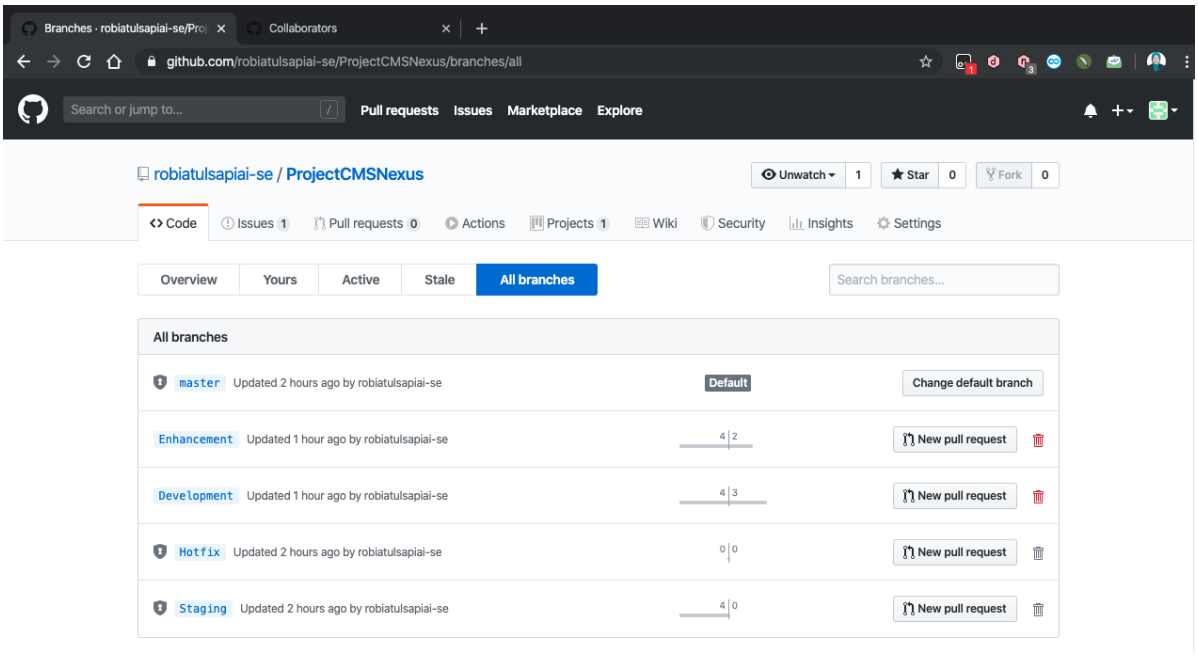


Figure 6.3 Project Branches

   b. The request will be checked based on the technical merit, possible side effects and overall impact on other configuration objects. the pull request for certain branches is compulsory to be reviewed and approved. This can be shown in figure below.

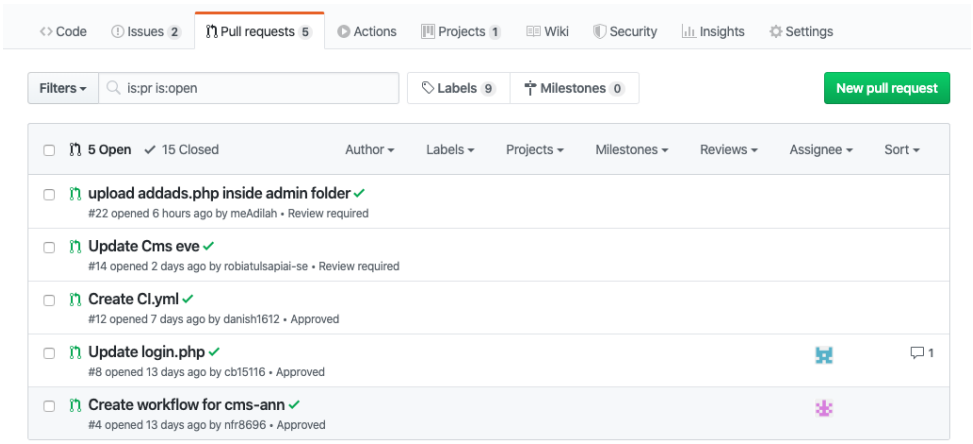


Figure 6.4 Open Pull Request that need Review Approval

3. Configuration Status Accounting: Configuration status accounting tracks each release during the SCM process. This stage involves tracking what each version has and the changes that lead to this version. Activities during this process:

   a. Monitor status of change requests by viewing the list of pull request if its open or close. the figure below shows there are 5 open pull request and 15 closed pull requests.

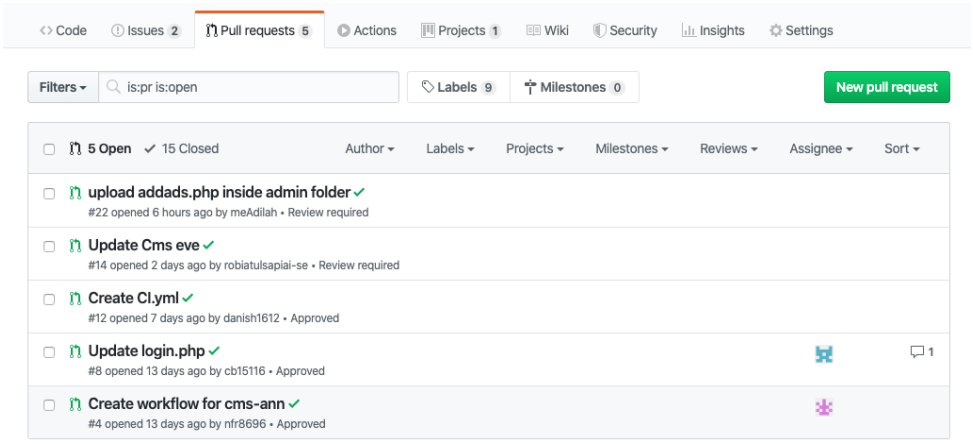


Figure 6.5 Open Pull Request

b. Do complete listing of all changes by retrieve it in the commit history to track changes as shown in the figure below.

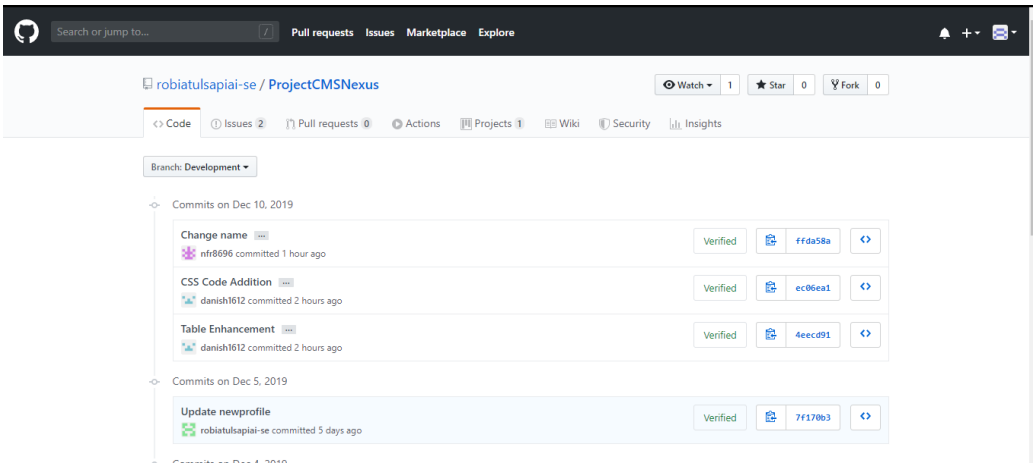


Figure 6.6 Commit History

c. Allows tracking of progress which can be viewed in project dashboard as shown in the figure below. the details about the dashboard will be explain in subchapter 6.1.5

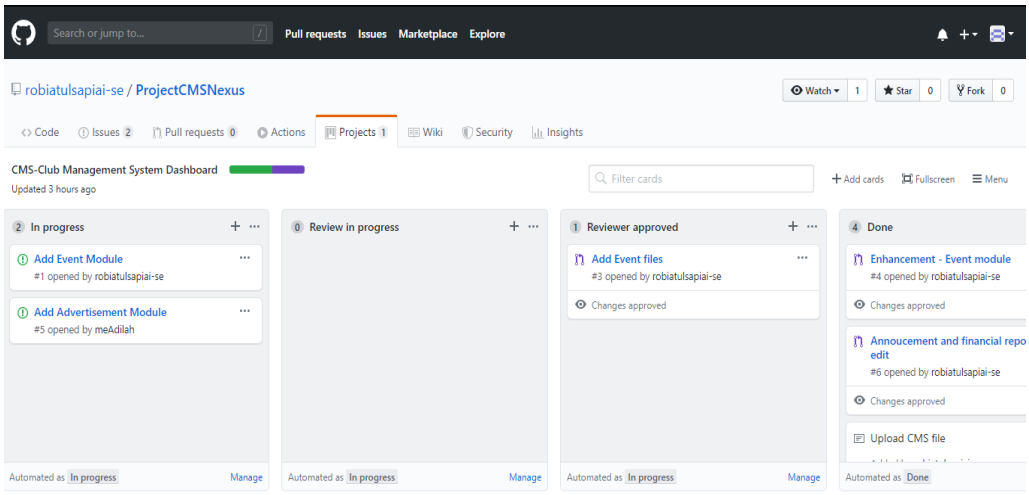


Figure 6.7 Dashboard

d. Allows to check previous releases/versions. By using GitHub, the changes made can be reverted. this can be shown in the figure below.

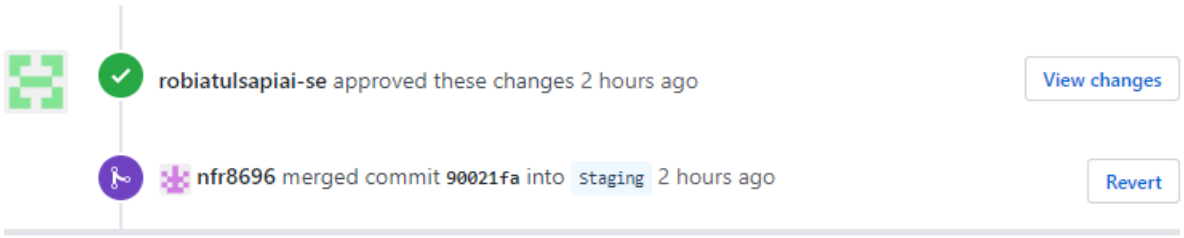


Figure 6.8 Release/Versions

4. Configuration Review Audit: Software Configuration audits verify that all the software product satisfies the baseline needs. It ensures that what is built is what is delivered. Activities during this process:

   a. Configuration auditing is conducted by leader by checking that defined processes are being followed and ensuring that the goals are satisfied. The goal is to systematically manage, organize, and control the changes in the documents, codes, and other entities. (refer Milestone)

   b. SCM audits also ensure that traceability is maintained during the process.

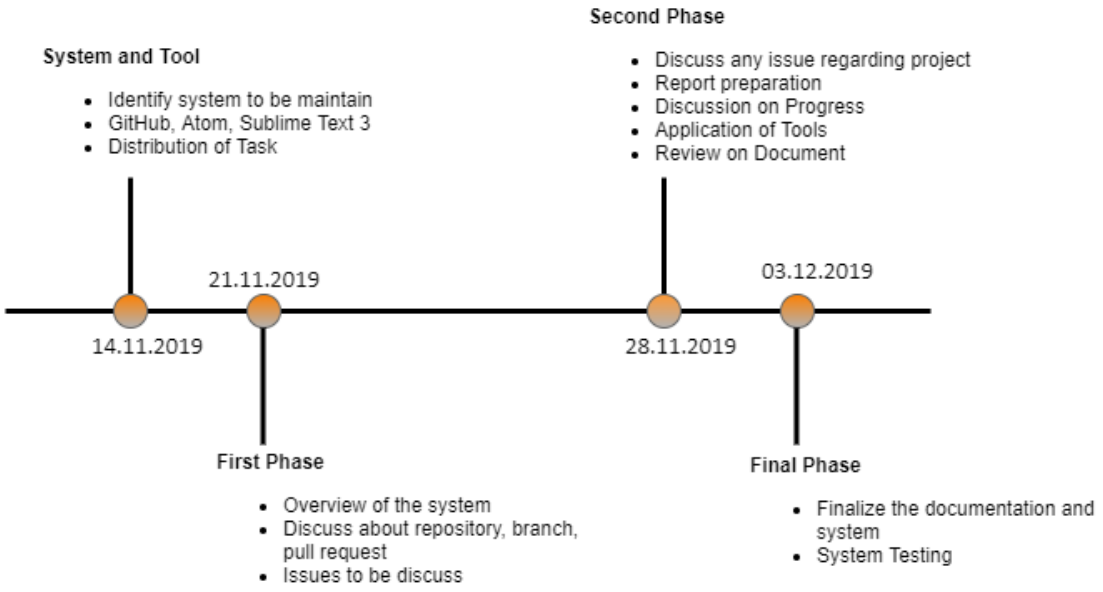


Figure 6.9 Milestone

## 6.2 SCM Tools

In this chapter, all the tools that were used in completing this project will be explained in detail. All the project repositories, branches, dashboard and other activity also will be shown in this chapter as well as the collaborator. The tool that is used is GitHub.

### 6.2.1 Repository

GitHub provides a web-based graphical interface for project management and code versioning system as well as a social network platform made for developers. It also provides access control and several collaboration features, such as wikis and basic task management tools for every project. Figure below shows our GitHub repository for Club Management System which is called as ProjectCMSNexus.

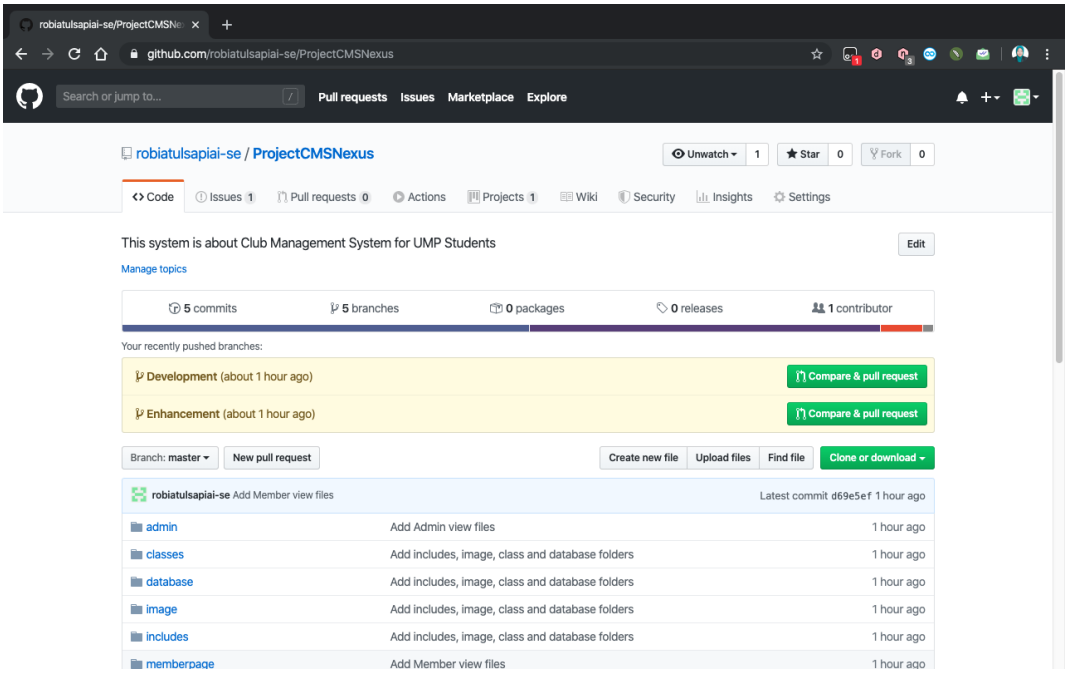


Figure 6.10 GitHub Repository CMS

### 6.2.2 Collaborator

There are six members who involved in this project. The group leader will invite the five members as collaborator once the repository is made. Figure 6.11 below shows the list of the collaborator who will involve in this project.

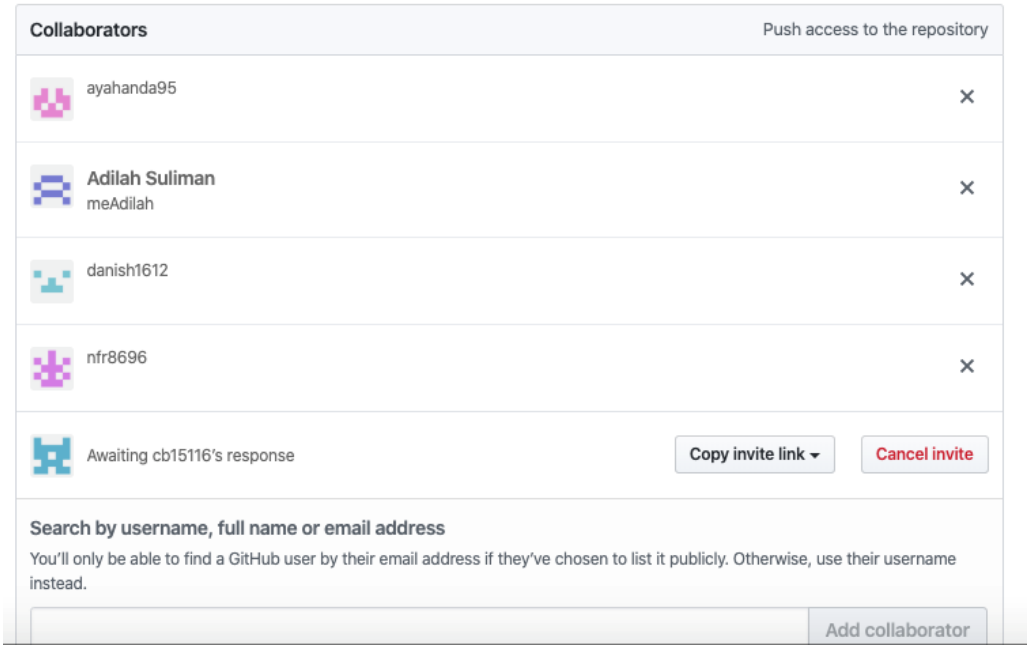


Figure 6.11 Collaborator Involved in this Project

All six members will be given different module to be managed and maintained as well as module to be evolve. Table 2 below show the details of the task delegation given to the team members:

Table 6.2 Task Delegation

| **Person in charge** | **Module** |
|---|---|
| CB15166 | Manage profile |
| CB16167 | Manage Finance |
| CB16164 | Manage Announcement |
| CB16169 | Manage Advertisement |
| CB16165 | Manage Event |

### 6.2.3 Branches

There are five branches that is created to ease the work for managing, maintaining and evolve this project. The five branches are:

Table 6.3 Branches

| Branch | Description |
|---|---|
| Master (Protected) | This is the default branch which is the first version of the CMS. It requires review approval to make pull request in this branch. |
| Staging (Protected) | This branch is created from master branch. Its function is to be the release branch from development branch. This branch can be merged into master branch once the system is agreed to be deployed. |
| Development | This branch is created from staging branch. All the error, remove codes, enhance interface can be done in this branch. This branch can be merge into staging branch. |
| Enhancement | This branch is created from staging branch. Its function is to cater the evolution of CMS such as adding new module into CMS. This differentiate between development branch which only cater maintenance activities. |
| Hotfix (Protected) | This branch is created from master branch. Its function is used when the CMS is already merge into master branch and when there is a bug that needs to be fixed. |

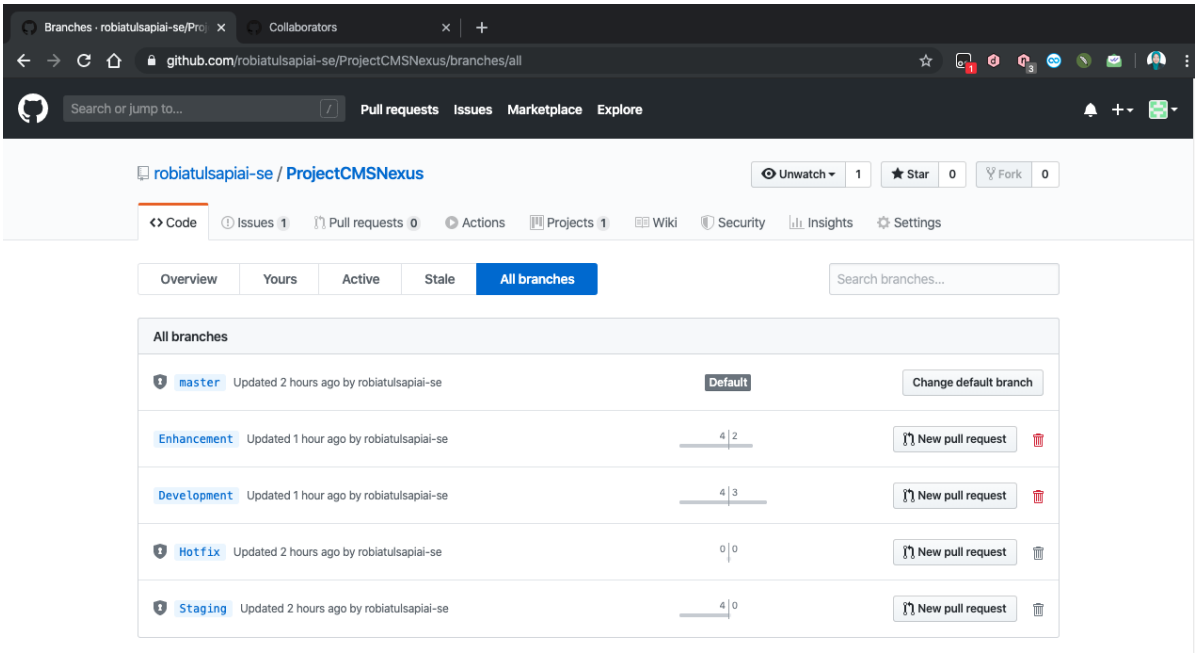


Figure 6.12 Branches in CMS Repository

### 6.2.4 Commit

All the activities history will be displayed and stored in the commit section where every single action such as merge will be displayed here. This can be shown in the figure below:

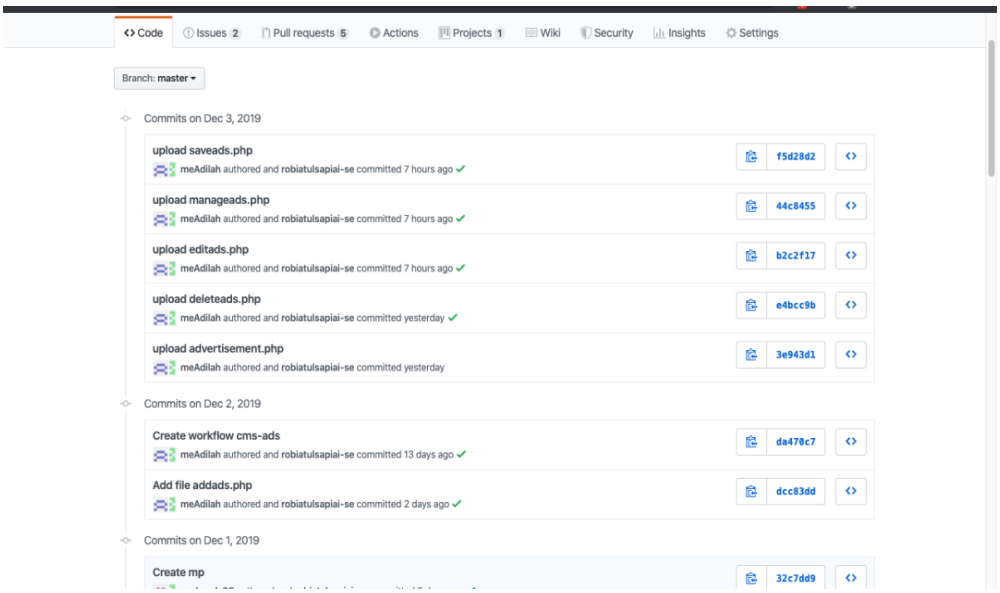


Figure 6.13 Commit Happen in Master Branch

### 6.2.5 Dashboard

Every project is crucial to have a dashboard or baseline to keep track everything such as what things to do, in progress, completed task, issues arise and many more. By using GitHub, we can keep track every activity status happen in CMS repository as shown in Figure 6.14 below.

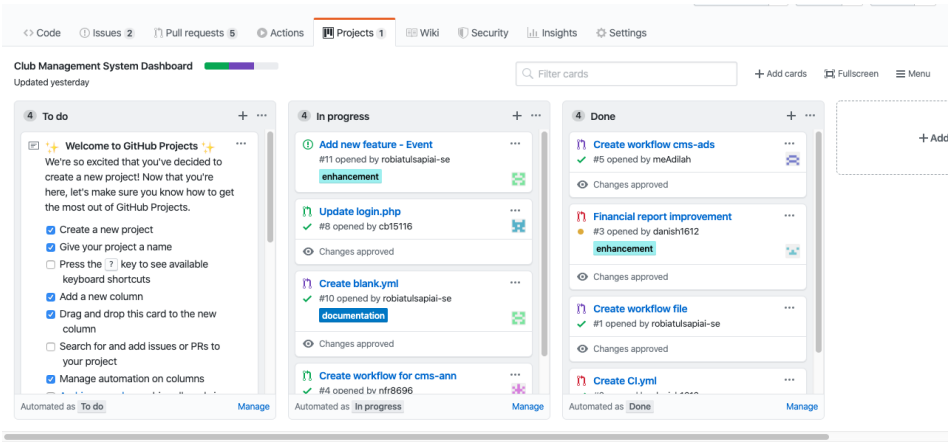


Figure 6.14 CMS Dashboard

### 6.2.6 Pull Request

All the change request from collaborator will be display in the pull request section. The open pull request need review approval from the leader. The leader has the rights to approve the pull request or change it. Figure below shows the pull request section for CMS.

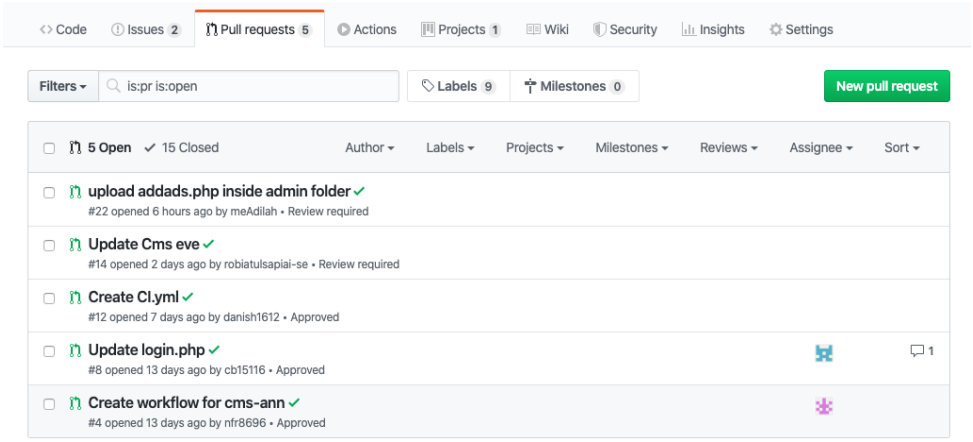


Figure 6.15 Open Pull Request that need Review Approval

## 6.3 Summary

Based on all the information explained in this chapter, there are two things that can be summarized when using the SCM tool called GitHub. Which the first one is how to do maintenance (refer Figure 6.16) and how to create new feature – evolution (refer Figure 6.17).

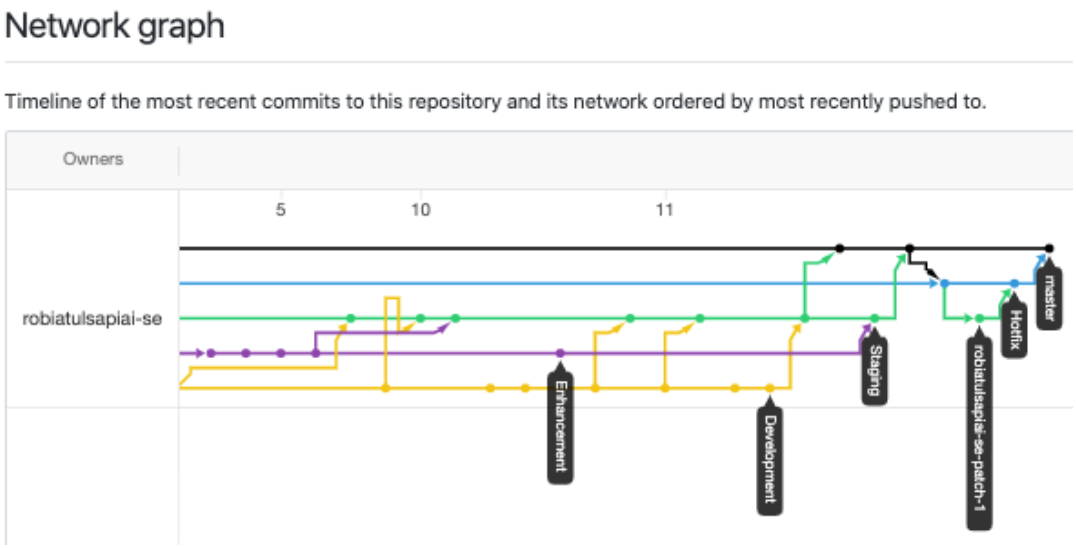


Figure 6.16 Maintenance Activities Flow represent in the purple line

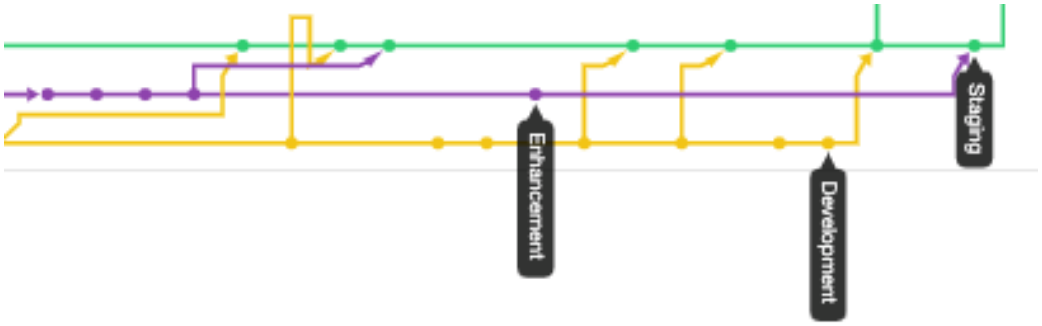


Figure 6.17 Evolution (i.e. Add Feature) Flow represent in the yellow line

# Chapter 7

# Result and Discussion

The results are discussed based on the change request that has been proposed by developer in Chapter 5. The output on the system will be shown and the conclusion of the output will be discussed.

## 7.1 Result of Change Request 1 (CMS_CR_01)

Change Request CMS_CR_01 request to change the type of input box to the description element for adding new announcement. Before the change request, the type of input box only <input> tag which is not proper. This is because description need to provide bigger text box for the administrator key in the details. Hence, by using <textarea> tag, it is much better. The interface (before change) is shown in Figure and after change is shown in Figure.

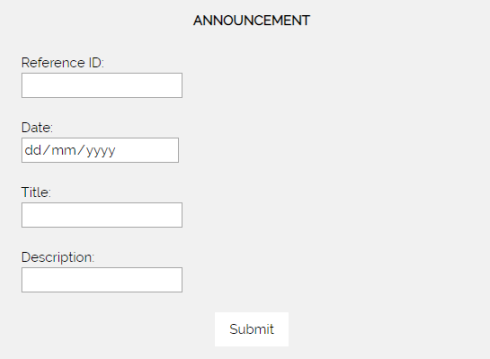


Figure 7.1 Before Change Request 1

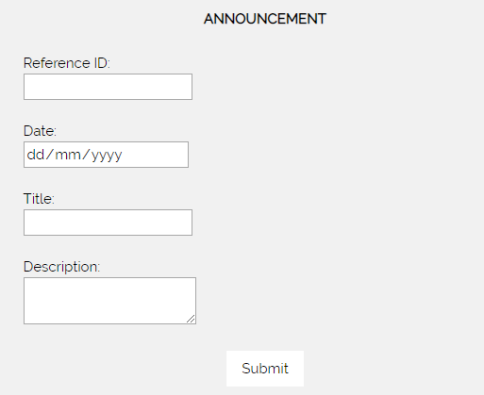


Figure 7.2 After Change Request 1

The other update for announcement is the description. The description of each title of announcement is not shown in announcement list table. So, the update needs to do as it is high priority to Manage Announcement module. The figure shows the interface before and after the changes.

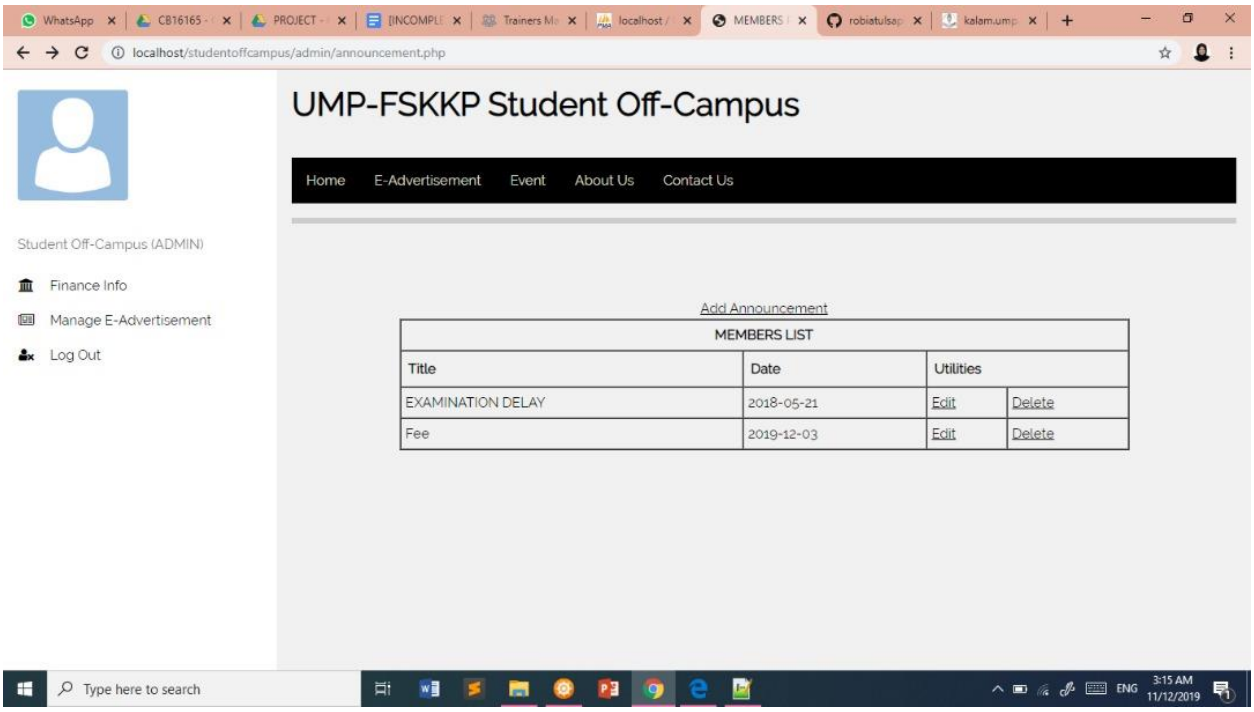


Figure 7.3 Before Change Request 1

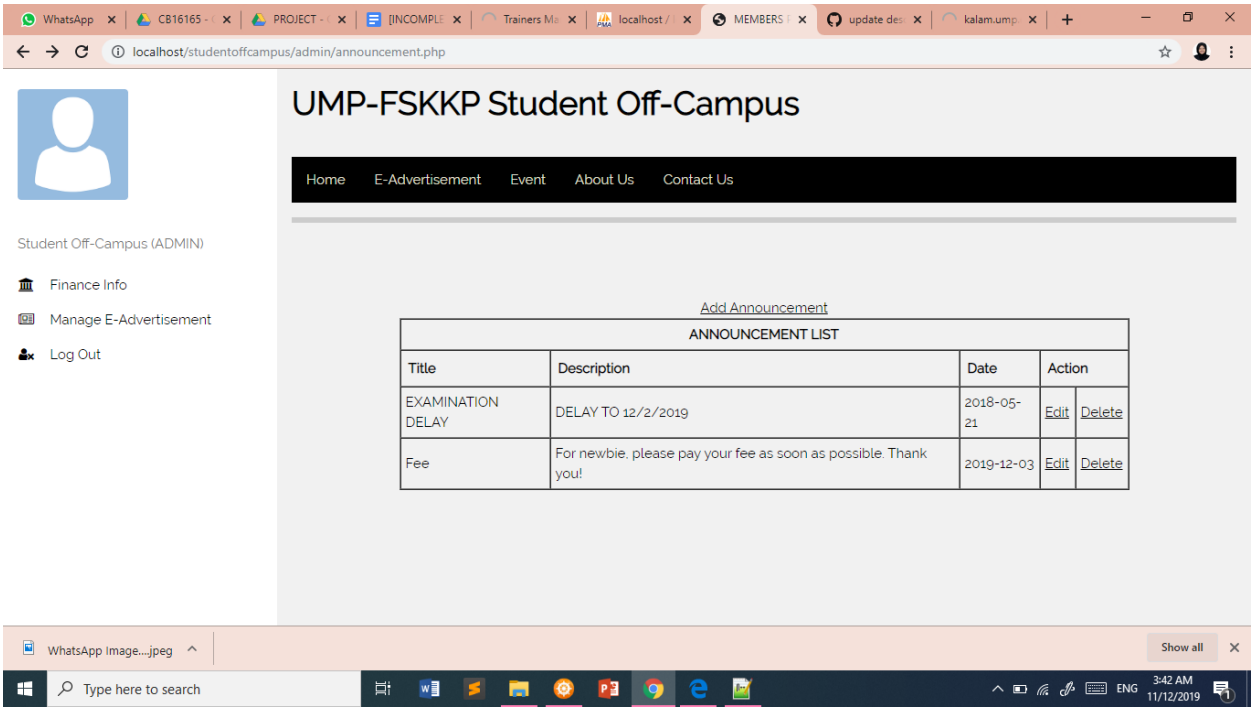


Figure 7.4 After Change Request 1

## 7.2 Result of Change Request 2 (CMS_CR_02)

Change Request CMS_CR_02 request to add new feature which is advertisement module in the system. Before the change request, the advertisement is included in the announcement. Therefore, for better club management, the advertisement module is created. The before interface is shown in Figure 7.5 and after change is shown in Figure 7.6.

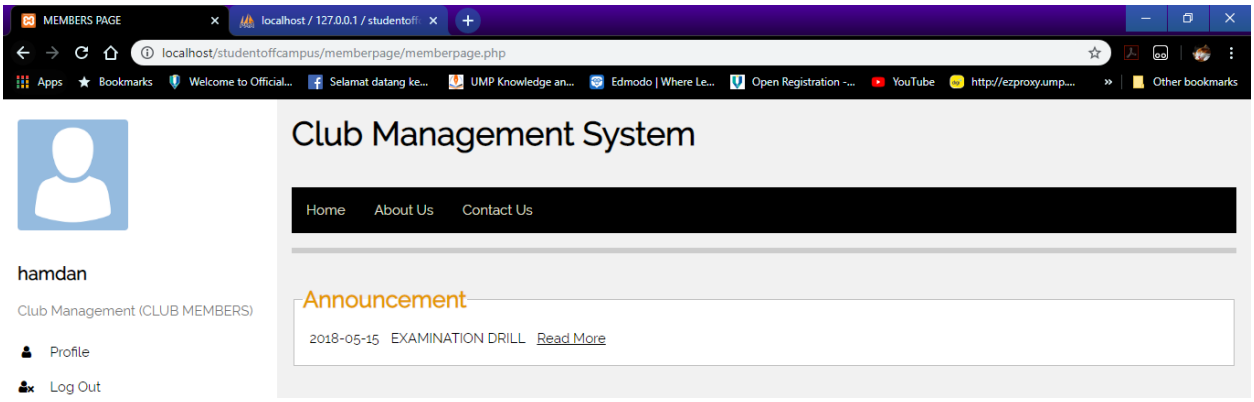


Figure 7.5 Before Change Request 2

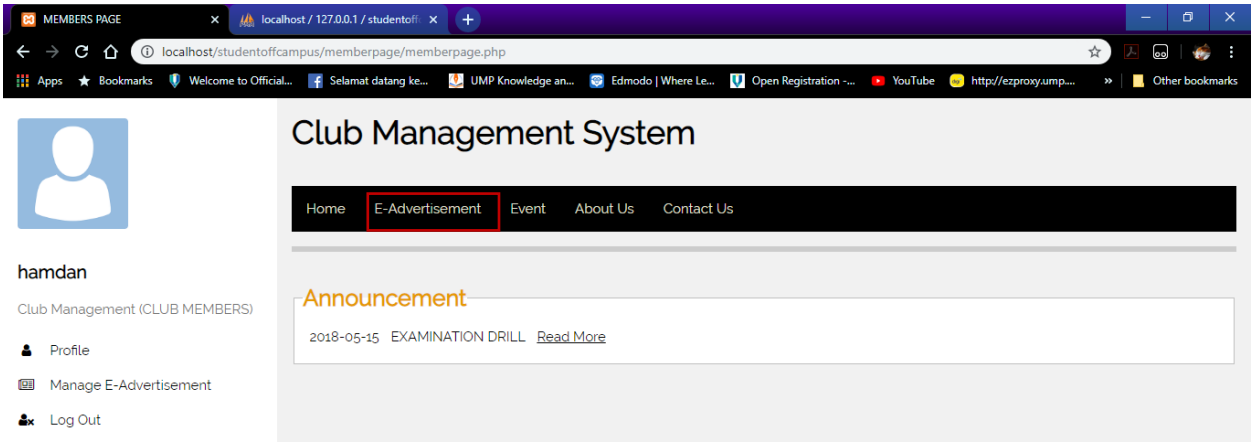


Figure 7.6 After Change Request 2

## 7.3 Result of Change Request 3 (CMS_CR_03)

Change Request CMS_CR_03 is to change the table in financial report to a more organized for the user experience. Before the change the 3 tables which are donation, expenses and usage are unorganized and very hard to. A proper table is needed so that it will be more organized and better for the user experience. Thus, few modifications of the code in the financial report is change in the table section to enhance the table as shown below.

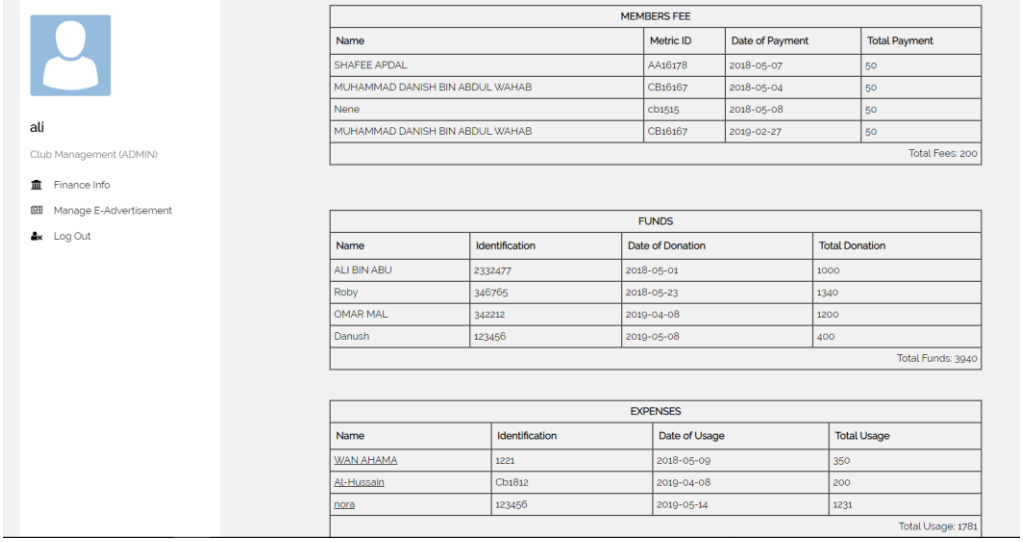


Figure 7.7 Before Change Request 3

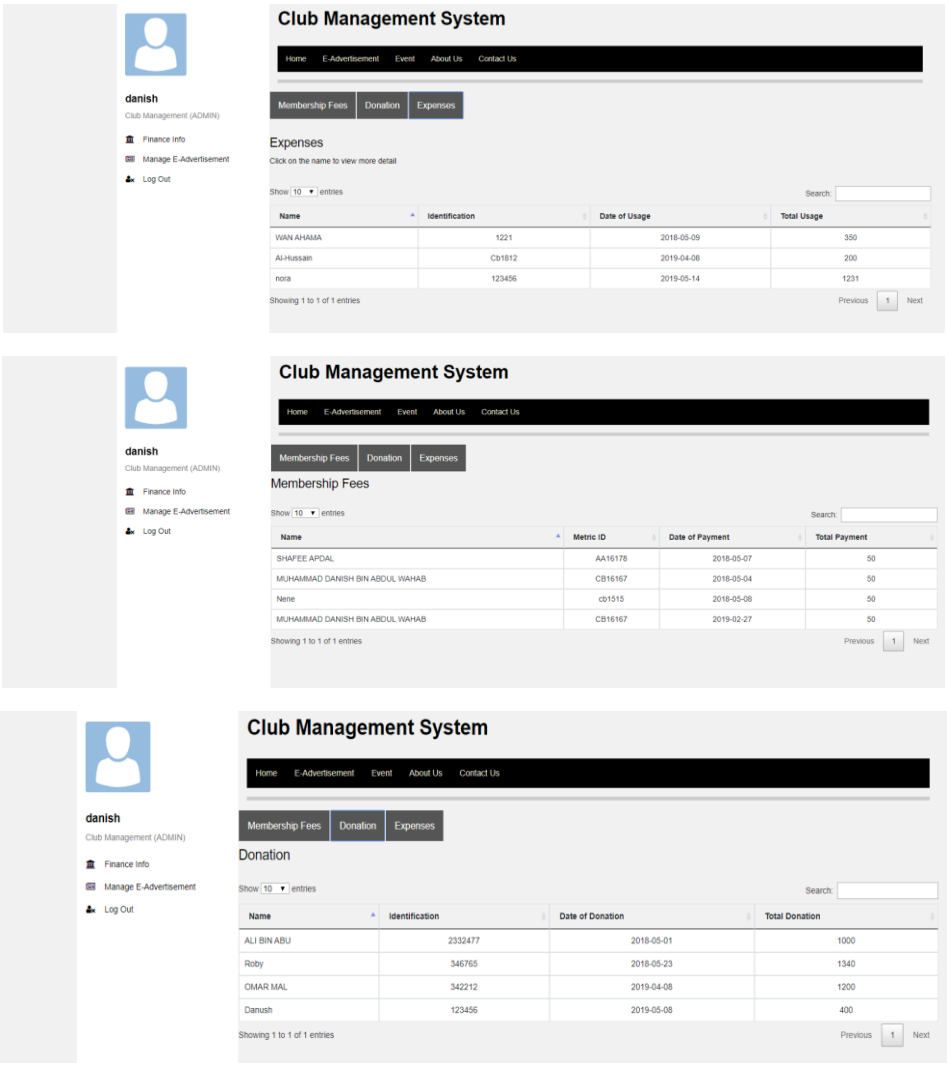


Figure 7.8 After Change Request 3

## 7.4 Result of Change Request 4 (CMS_CR_04)

Change Request CMS_CR_04 request to add new feature which is event module in the system. Before the change request, the event is included in the announcement. Therefore, for better club management, the event module is created. The before interface is shown in Figure 7.9 and after change is shown in Figure 7.10.

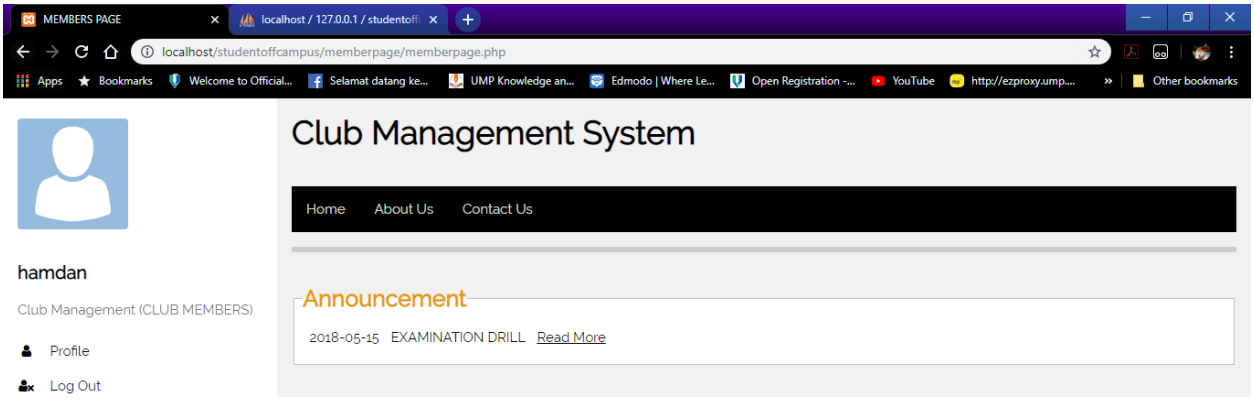


Figure 7.9 Before Change Request 4

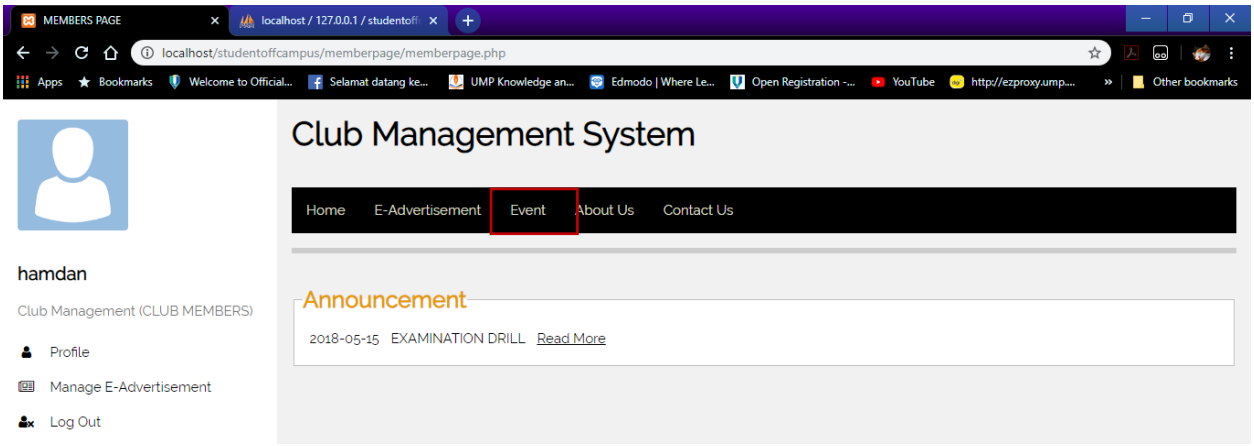


Figure 7.10 After Change Request 4

## 7.5 Hotfix

After merging with the master's branch, we conduct a test to use the system for each module including the new features and modifications. During the testing, a bug is found within the financial report specifically expenses table where the when click the data for more information. The result came out with variable. The figure below shows the bug occur.

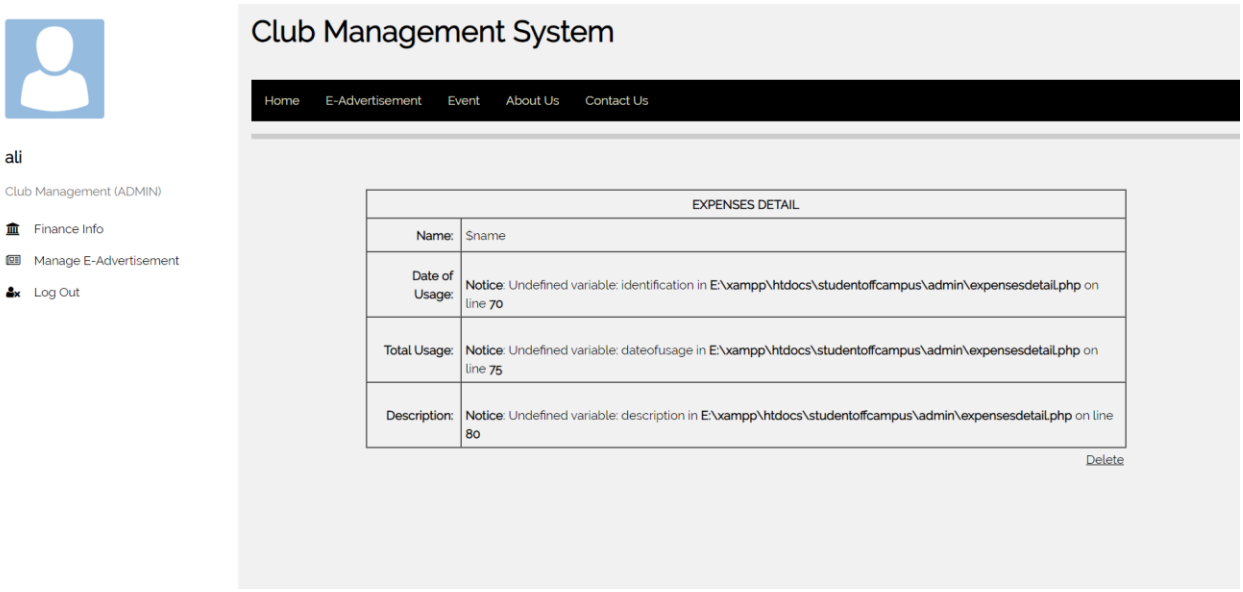


Figure 7.11 Bug Occur in System

The team analyze the error and found out that one of the variables is outside PHP Coding. Thus, a hotfix is conducted to fix the error in the master's branch. The figure below shows how the error is fixed.

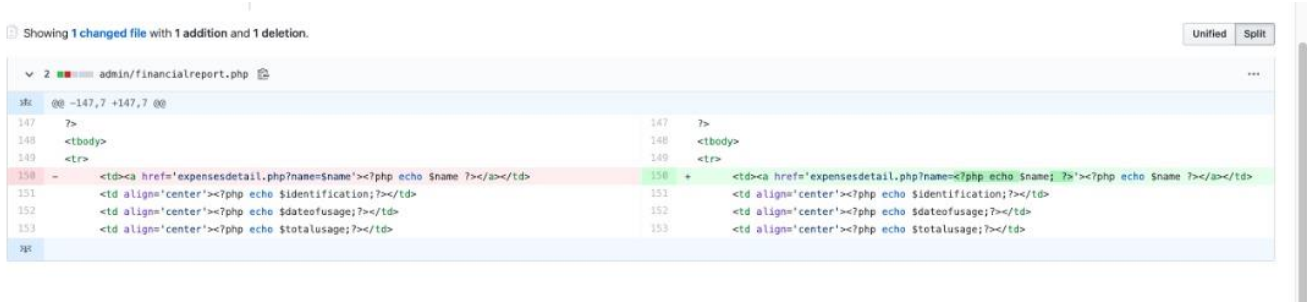


Figure 7.12 Fixing Bug

After the hotfix has been done, a testing is conducted to ensure that the bug is fix and free from any error. The figure below shows the result of the hotfix.

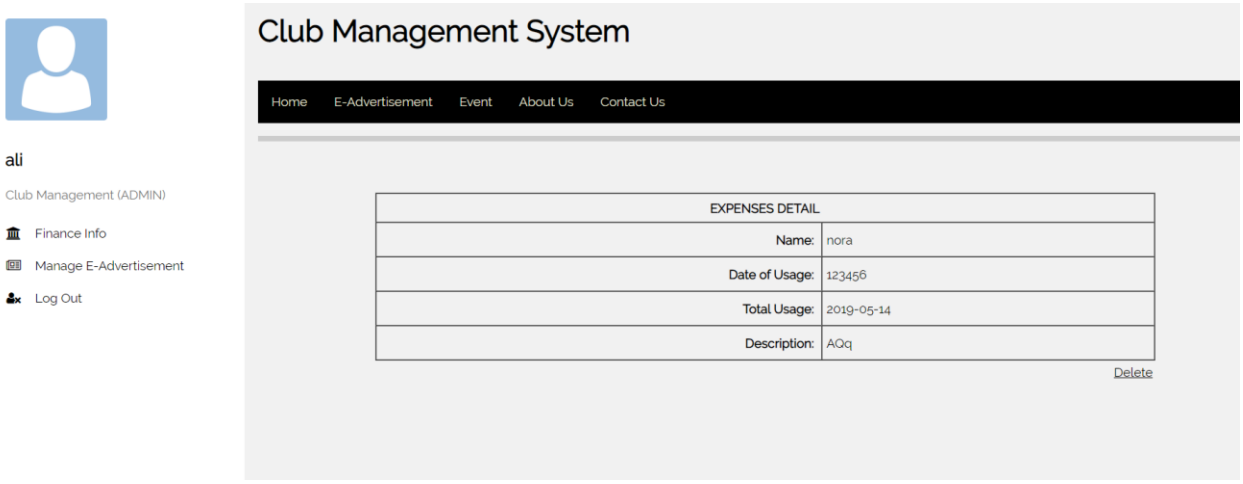


Figure 7.13 Result of Hotfix

# Chapter 8

# Meeting Minutes

MEETING MINUTES 1

| **PROJECT NAME** | SEM Mini Project | | |
|---|---|---|---|
| **DATE** | 14.11.2019 | **TIME** | 2.00pm |
| **MINUTES PREPARED BY** | Robiatul Adawiyah Binti Sapia'i | **LOCATION** | FSK27 |
| **MEETING OBJECTIVES** | | | |
| ● Choose system that will be maintain<br>● Decide SCM Tools to be used<br>● Task Delegation<br>● Explore GitHub and Git | | | |
| **ATTENDANCE** | | | |
| Robiatul Adawiyah Binti Sapia'i<br>Nurfarzana Binti Rosli<br>Nurul Adilah Binti Suliman<br>Muhammad Danish Bin Abdul Wahab<br>Nabil Bin Lutfi | | | |
| **AGENDA, NOTES, DISCUSSION AND ISSUES** | | | |
| **TOPIC** | | **OWNER** | **DURATION** |
| Discussion SCM tools<br>● The difference of Git, GitHub, Gitlab | | All Members | 2 Hours |

| | | | | | |
|---|---|---|---|---|---|
| Discuss which system to be maintained<br>● Decided to use our own project (Club Management System)<br>● Divide each module to each member | | | | | |
| **NEXT MEETING** | | | | | |
| **DATE** | 21.11.2019 | **TIME** | 2.00pm | **LOCATION** | FSK27 |
| **OBJECTIVE** | ● Discuss about repository, branch, pull request<br>● Report discussion | | | | |

MEETING MINUTES 2

| | | | |
|---|---|---|---|
| **PROJECT NAME** | SEM Mini Project | | |
| **DATE** | 21.11.2019 | **TIME** | 2.00pm |
| **MINUTES PREPARED BY** | Robiatul Adawiyah Binti Sapia'i | **LOCATION** | FSK27 |
| **MEETING OBJECTIVES** | | | |
| ● Discuss about repository, branch, pull request<br>● Report discussion | | | |
| **ATTENDANCE** | | | |
| Robiatul Adawiyah Binti Sapia'i<br>Nurfarzana Binti Rosli<br>Nurul Adilah Binti Suliman<br>Muhammad Danish Bin Abdul Wahab<br>Nabil Bin Lutfi | | | |
| **AGENDA, NOTES, DISCUSSION AND ISSUES** | | | |
| **TOPIC** | | **OWNER** | **DURATION** |
| Discuss about how to use GitHub<br>● Repository, branch and pull request<br>● Start to create dashboard<br>● Which part should be maintained<br>Technical report discussion | | All Members | 2 Hours |
| **NEXT MEETING** | | | |

| | | | | | |
|---|---|---|---|---|---|
| **DATE** | 28.11.2019 | **TIME** | 2.00pm | **LOCATION** | FSK27 |

| | |
|---|---|
| **OBJECTIVE** | ● Update on each module<br>● Any issues to be discuss |

MEETING MINUTES 3

| | | | |
|---|---|---|---|
| **PROJECT NAME** | SEM Mini Project | | |
| **DATE** | 28.11.2019 | **TIME** | 2.00pm |
| **MINUTES PREPARED BY** | Robiatul Adawiyah Binti Sapia'i | **LOCATION** | FSK27 |
| **MEETING OBJECTIVES** | | | |
| ● Update on each module<br>● Any issues to be discuss | | | |
| **ATTENDANCE** | | | |
| Robiatul Adawiyah Binti Sapia'i<br>Nurfarzana Binti Rosli<br>Nurul Adilah Binti Suliman<br>Muhammad Danish Bin Abdul Wahab<br>Nabil Bin Lutfi | | | |
| **AGENDA, NOTES, DISCUSSION AND ISSUES** | | | |
| **TOPIC** | | **OWNER** | **DURATION** |
| Update each module<br>● Module finance, manage announcement have some changes to the interface<br>● Module login and manage profile does not have request for change<br>New feature to be added<br>● Manage Event and manage advertisements are added from announcement module | | All Members | 2 Hours |

| | | | | | |
|---|---|---|---|---|---|
| | | | Adilah<br>Robiatul | | |
| **NEXT MEETING** | | | | | |
| **DATE** | 3.12.2019 | **TIME** | 9.00pm | **LOCATION** | WDK03 |
| **OBJECTIVE** | ● Update on project progress<br>● Discuss any issue regarding project<br>● Report preparation | | | | |

MEETING MINUTES 4

| | | | |
|---|---|---|---|
| **PROJECT NAME** | SEM Mini Project | | |
| **DATE** | 03.12.2019 | **TIME** | 9.00pm |
| **MINUTES PREPARED BY** | Robiatul Adawiyah Binti Sapia'i | **LOCATION** | WDK03 |
| **MEETING OBJECTIVES** | | | |
| ● Update on project progress<br>● Discuss any issue regarding project<br>● Report preparation | | | |
| **ATTENDANCE** | | | |
| Robiatul Adawiyah Binti Sapia'i<br>Nurfarzana Binti Rosli<br>Nurul Adilah Binti Suliman<br>Muhammad Danish Bin Abdul Wahab<br>Nabil Bin Lutfi | | | |
| **AGENDA, NOTES, DISCUSSION AND ISSUES** | | | |
| **TOPIC** | | **OWNER** | **DURATION** |
| Update on project progress<br>● Module manage event and manage advertisements in progress to finish<br>● Discuss on error problem when update finance<br>● Report preparation | | All Members | 2 Hours |