CEBU INSTITUTE OF TECHNOLOGY UNIVERSITY

COLLEGE OF COMPUTER STUDIES

CIT-U MEDAL COUNT

Team Member:

John Vincent Capoy

Neil Anthony Canton

Noe Norman Galo

Alwynn Batucan

Adviser:

Adviser: Mario Silvano

ABSTRACT:

This project was develop in order to have an easy and accurate way of providing a vital information for CIT-U's students, staff and faculty members about the intramurals, and record list of winners in every department. This is to provide school's Tally sheet of our Intramurals event. Our client is Dr. Larmie feliscuzo she is the assistant chair of our department college of computer studies, Cebu Institute of Technology University.

Student's staff and faculty members of CIT-U are always behind of the new or updates about the events during Intramurals, to aid this problem we come up to an idea that will make use of website which is the "Medal Tally" that provides an information about all the winners in all games. So that all the students, staff and faculty members will be updated about the happenings during intramurals.

There are possible revision that could be made for its improvements such as on sport page the winning list that are categorize in Gold, silver, Bronze will no longer appear in Silver or in Bronze once it is already in Gold category.

Software Requirements Specifications

for

CIT-U Medal Tally

signature

Members	Roles	Signature
John Vincent Capoy	Documentation/Web Developer	
Niel Anthony Canton	Web Developer	
Noe norman galo	Web developer	
Alwynn batucan	tester	

Change History

Revision History

Name	Date	Reason For Changes	Version

Approval History

Version	Reviewed By	Reviewer Title	Date

Preface

This is the software requirements specifications (SRS) document for the CIT-U Medal Tally Website. In particular, the document details the features, functions and software specifications of the events. The intended users of the SRS are the faculty, staff, and students specifically the new students.

-

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1. Purpose

Develop a CIT-U Medal Tally Website allowing the students, staff, and faculty to update school information. Our main purpose of this project is to provide vital information for CIT-U's students, staff, and faculty members about the school events, and record lists of winner's, scores of the main event. This is to provide school's Tally Sheet of the upcoming Intramurals.

1.2. Scope

The purpose of this CIT-U Events Medal Tally Website is to:

- To provide a simple and fastest way to inform students, staff, and teachers about the updates and fresh information about the school events.

1.3. Definitions, Acronyms and Abbreviations

- **Website** is a location connected to the internet that maintains one or more pages on the World Wide Web.

1.4. References

- www.google.com
- IEEE 12207.2-1997 Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207)Standard for Information Technology - Software Life Cycle Processes - Implementation Considerations
- IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specifications
- [IEEE] The applicable IEEE standards are published in "IEEE Standards Collection," 2001 edition.

1.5. Overview

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Specific Requirements section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2. Overall Description

2.1. Product perspective

- This CIT-U Medal Tally Website will be used as a guideline tools for students.
- This CIT-U Medal Tally Website will be downloaded to an android phone and will be runnable to almost all mobiles and tablets.

2.2. Product functions

- Department
- medals
- Sports
- Gamelog
- School year

2.3. User characteristics

- Students, faculty and staff will attend the prescribed events.
    - Example for the events is intramurals. The students will participate the events.

2.4. Constraints

- Only the user and admin can add, update and edit/delete the gamelog or points of any department.

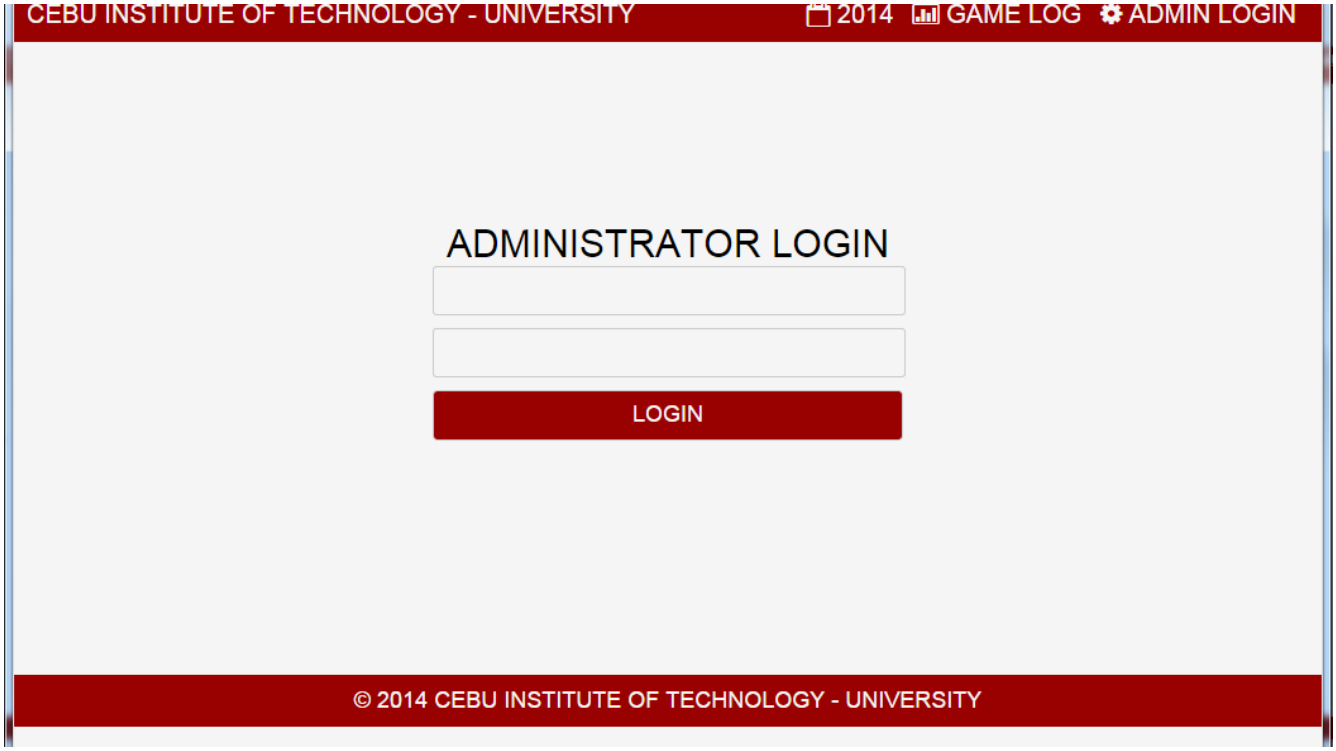
2.4. Assumptions and dependencies

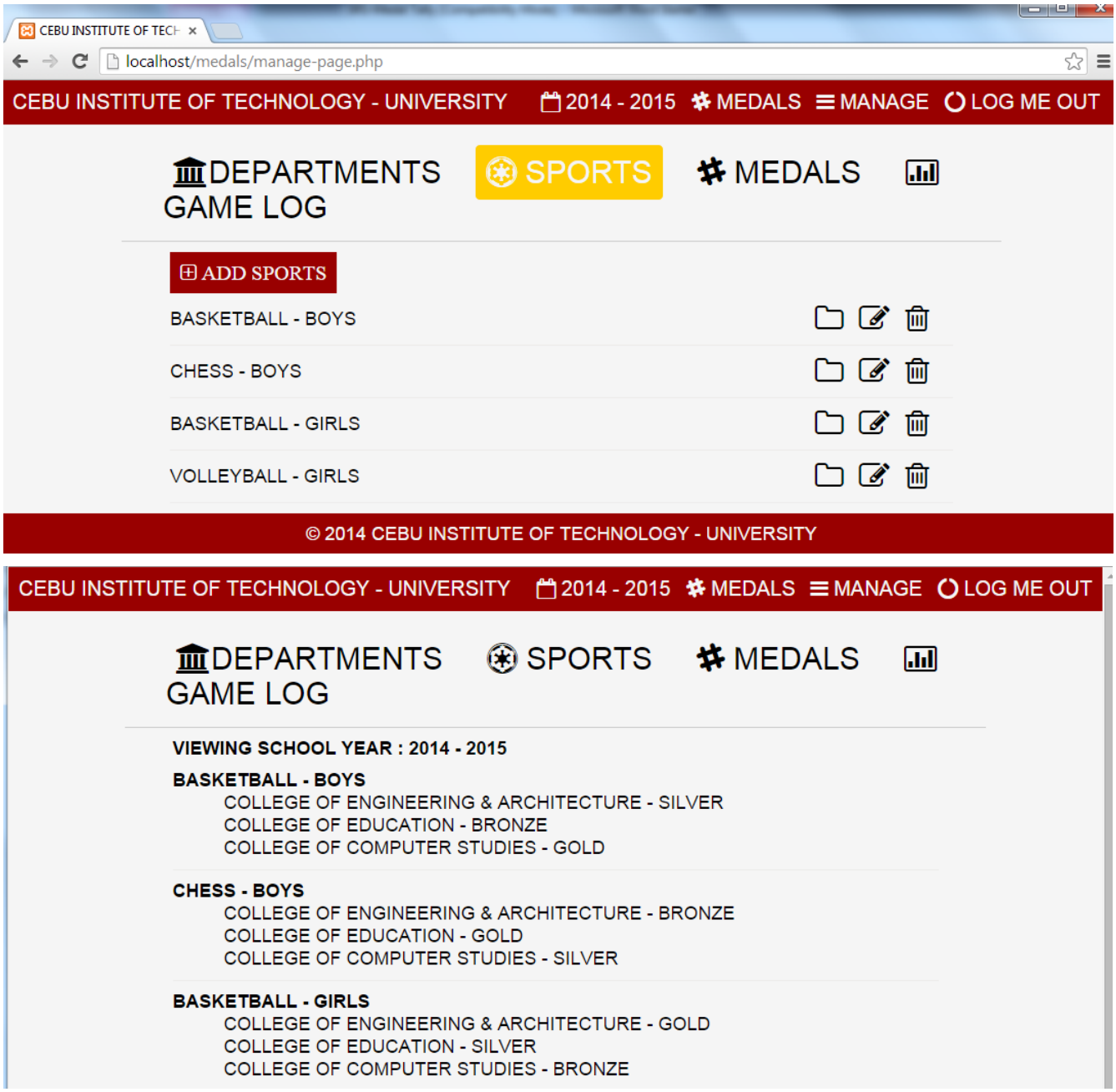
If the system will be used in Cebu Institute of Technology University, the process of which the management is done will be altered.it will be as easy to them to see the score and standing.

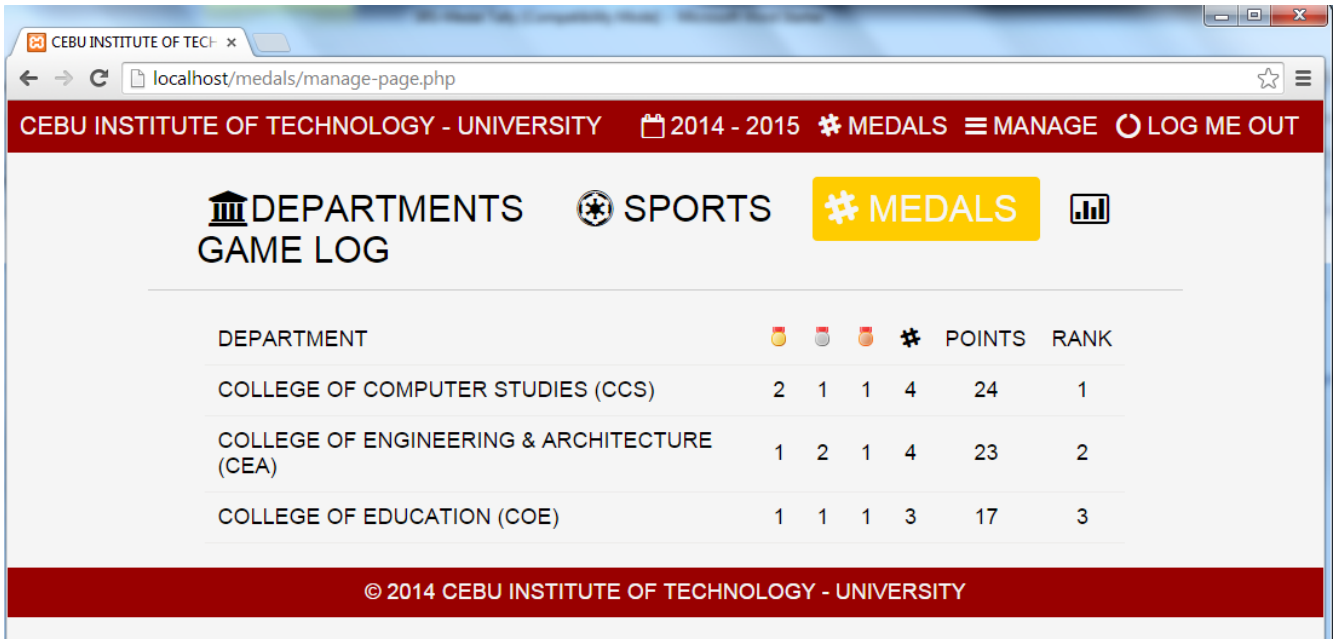
3. Specific Requirements

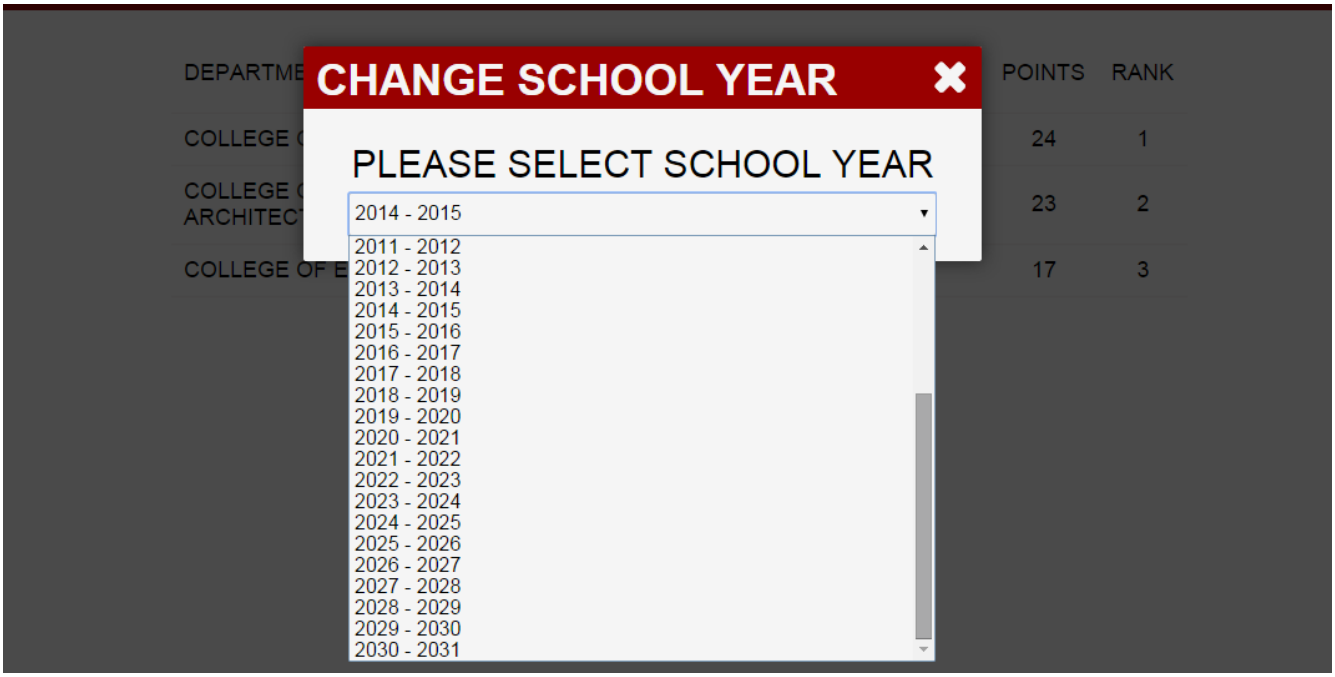
3.1. External interface requirements

3.1.1. User interfaces




CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
ADD SPORTS
BASKETBALL - BOYS
CHESS - BOYS
BASKETBALL - GIRLS
VOLLEYBALL - GIRLS
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
VIEWING SCHOOL YEAR : 2014 - 2015
BASKETBALL - BOYS
COLLEGE OF ENGINEERING & ARCHITECTURE - SILVER
COLLEGE OF EDUCATION - BRONZE
COLLEGE OF COMPUTER STUDIES - GOLD
CHESS - BOYS
COLLEGE OF ENGINEERING & ARCHITECTURE - BRONZE
COLLEGE OF EDUCATION - GOLD
COLLEGE OF COMPUTER STUDIES - SILVER
BASKETBALL - GIRLS
COLLEGE OF ENGINEERING & ARCHITECTURE - GOLD
COLLEGE OF EDUCATION - SILVER
COLLEGE OF COMPUTER STUDIES - BRONZE





3.1.2. Hardware interfaces

- None: This is a Web Applications
- PC or MAC

3.1.3. Software interfaces

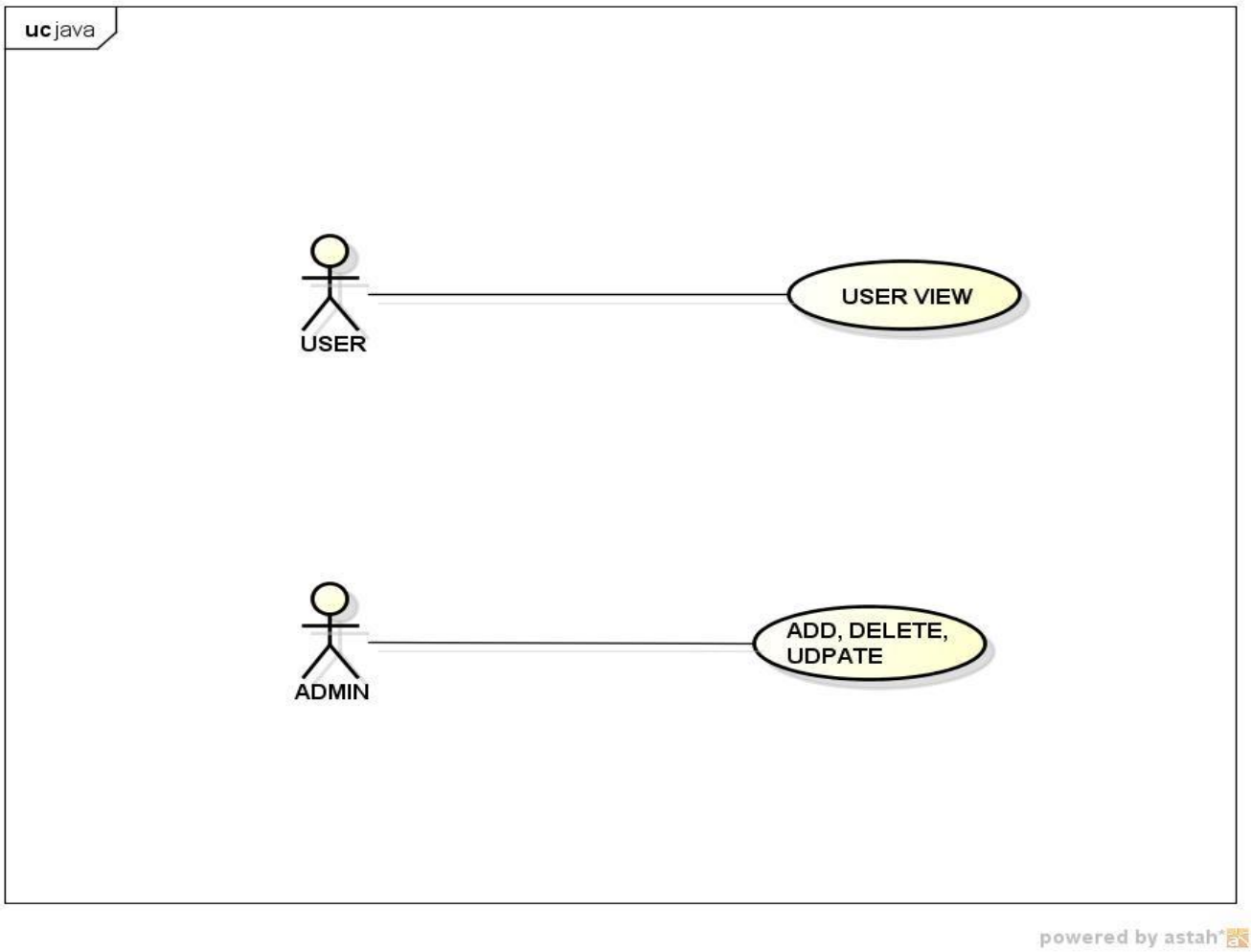
- Database
- Web Applications

3.1.4. Communications interfaces

None.

3.2. Functional requirements

3.2.1. Use case 1



User View:

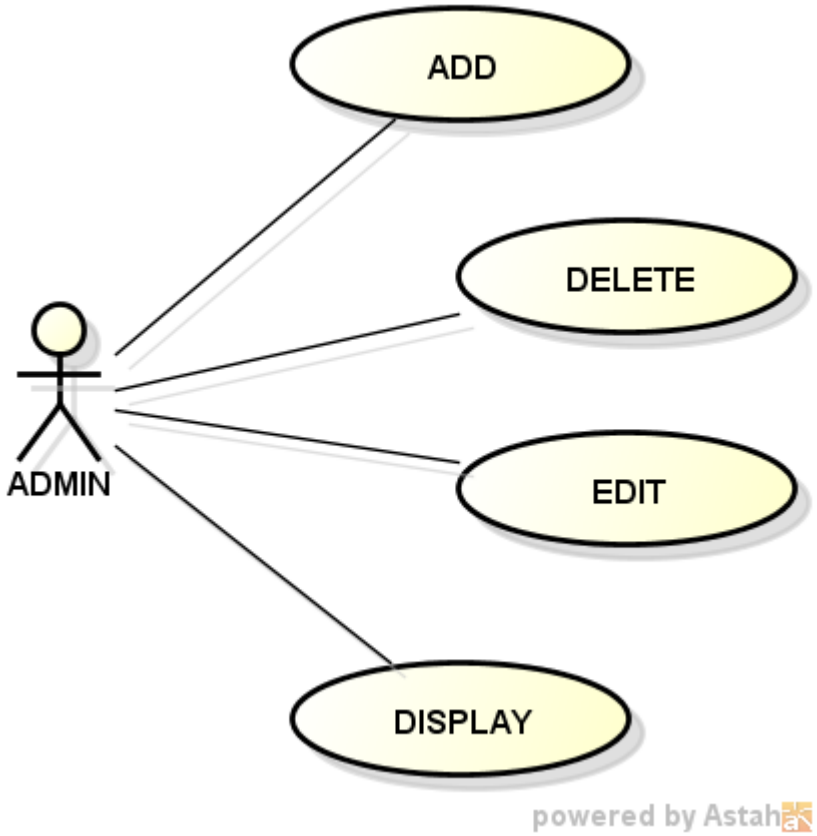
- User can view events, scores and schedules of the game in the web.

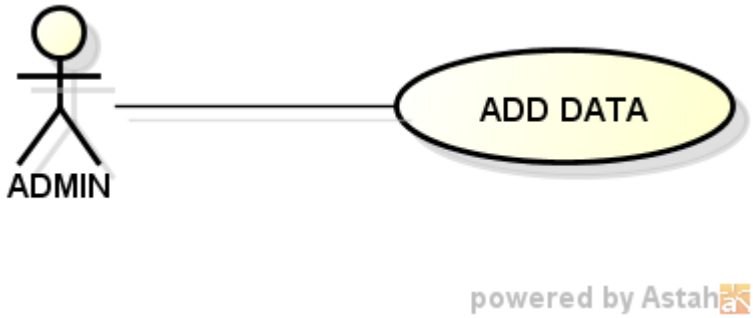
Admin View:

- Admin can edit, add, and update.

3.2.2. Use case 2

The Admin has the following sets of use cases





Use case: Add Data

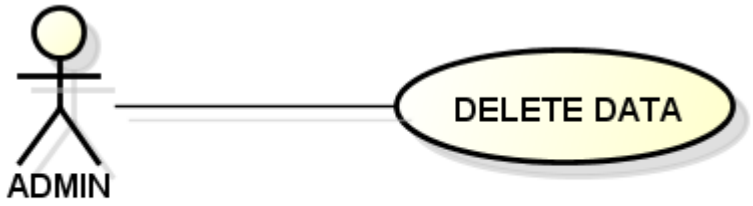
Brief Description

The Administrator will now add the collected information into the system's database

Initial Step-By-Step Description

1. The System now stores the information in the database

The Information will now be added into the system.



Use case: Delete Data

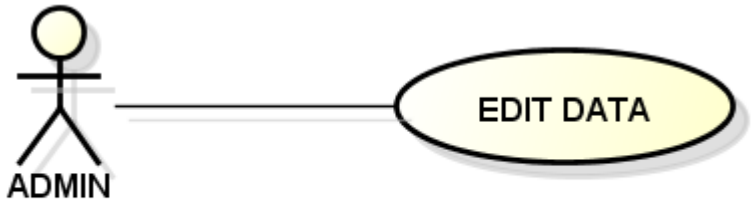
Brief Description

The Administrator may delete the data that was stored in the database.

Initial Step-By-Step Description

1. The System will delete it from the database, so it can input more schedule or add events.

The Information will not be found in the database.



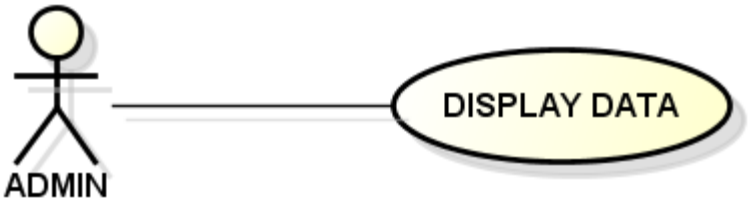
Use case: Edit Data

Brief Description

The Administrator can change the schedule/points of the game if they wanted to change plans.

Initial Step-By-Step Description

1. The System has the tool to update or change the reservation, and once they have changed they will need to see if there are available schedule.



Use case: Display Data

Brief Description

The Administrator can see the schedules and events.

Initial Step-By-Step Description

1. The Admin can display the schedule and see if there are some schedules which are to be updated or deleted.

3.3. Performance Requirements

- Any Computer device

3.4. Design constraints

3.5. Software system attributes

3.6. Other requirements

4. Appendixes

5. Index

Software Project Management Plan

for

CIT-U Medal Tally

Signature

Member Name	Position	Signature
John Vincent capoy	Documentation/web developer	
Neil Anthony canton	Web developer	
Noe norman galo	Web develop	
Alwynn batucan	Tester	

Change History

Name	Date	Reason For Changes	Version
John Vincent capoy	Aug. 1, 2014	Draft	0.10

Preface

The following Software Project Management Plan (SPMP) describes the proposed plan to be taken by our group, to contribute to the completion of the medal tally for CIT-U.

The document will address the work completed by the group Letter in concise language detailing the usages of the software. The intended audience of the SRS are users, system consultants and as per request the end-user.

Table of Contents

List of Figures

List of Tables

6. Overview

6.1. Project Summary

1.1.1. Purpose, scope and objectives

This project concerns with implementing the topic Software Project Management Plan in to the Inventory management system project. It involves defining appropriate procedures to provide detailed guidance for preparing and updating of SPMPs based on standards and ensuring that these are followed, to develop a project plan where management is seen as everyone's responsibility and every important details about the said topic should be place into this matter for the better learning that may serve as benefit for the students to learn more.

The scope of this project is to develop an a medal tally system for CIT-U. To focus with serving this project with the best assurance of a good process and product quality of the tools or materials.

The purpose of the software is to help the student and faculty teacher to fasten access to the events.

The objective of this system is to give the end-user the flexibility in production scheduling, maintain independence of operations. The system must meet a wide variation in product deman and provide a safeguard for variation in raw material delivery time.

1.1.2. Assumptions and constraints

The project shall be finished before oct. 2, 2014, the end of the 2nd semester of the Software Engineering (SE) course. All codes shall be written in html5,php language. The software shall be working at any Desktop (PC) Computers. The codes written shall be in html language. Upon deployment of the system to the client, the system should be a standalone in pc environment with its own UI with the database on MS Access DB. The estimated duration for the completion of the project is for one semester only. On the system deployment, it should be usable on all computers and should be finished before the semester ends.

The end-user such as clerks will be our main focus in this software. in our case, we will try to understand what the clerks really need to maximize production while providing safeguard for variation in raw material, that would take advantage of economic purchase-order size.

1.1.3. Project deliverables

As part of the project, the Group will deliver the following documents and requirements to the client.

- Software Requirement Specification (SRS) document including use case maps, list of features within the
  scope of the project and the Quality Assurance requirements.
- Software Project Management Plan document.
- Software Test Documentation document.
- Software Design Description document.
- Status report of the software.
- All other documents and requirements that the group might have generated that add value to the final

  deliverable.

1.1.4. Schedule and budget summary

No further budget needed, resources and other requirements are attainable without spending any amount.

[Table 1.1.5] Schedule Allocation Plan

Milestone	Date (initiation / completion)
Proposal	July 11, 2014

System Requirements Specifications	July 26, 0214
Software Project Management Plan	August 2, 2014
Software Design Document	August 9, 2014
Software Test Document	August 16, 2014
Increment 1	September 8, 2014
Increment 2	September 29, 2014
Increment 3	October 9, 2014
Increment 4	October 10, 2014
Software Output Presentation	September 20, 2014
Final Documents	October 2, 2014

6.2. Evolution of plan

[Table 1.2.1] Evolution Plan

Version	Primary Author(s)	Description of Version	Date Expected
Draft	John Vincent capoy	Initial draft created for distribution and review comments	July 23, 2014
Preliminary	TBD	Second draft incorporating initial review comments, distributed for final review	July 24, 2014
Prerelease	TBD	Third draft that is about to be released to the customer	July 24, 2014
Final	TBD	First complete draft, which is placed under change control	July 25, 2014
Revision 1	TBD	Revised draft, revised according to the change control process and maintained under change control	TBD
etc.	TBD	TBD	TBD

7. References

IEEE Std 1058-1998 IEEE standard for software project management plans

Karl E. Wiegers SOFTWARE REQUIREMENTS, 2003 Microsoft

Printed Resources:

- Java Concepts: Early Objects
  By; Horstman, Cay

- Java Programming: from the ground
  By: Bravaco, Ralph

- Operating System Concepts - Seventh Edition
  By: Silberschatz, Galvin and Gagne

Online Resources:

http://www.ehow.com/about_6506466_inventory-system-definition.html

http://www.lighthousesystems.com/inventory.htm

http://stackoverflow.com/questions/13891046/database-structure-for-inventory-system

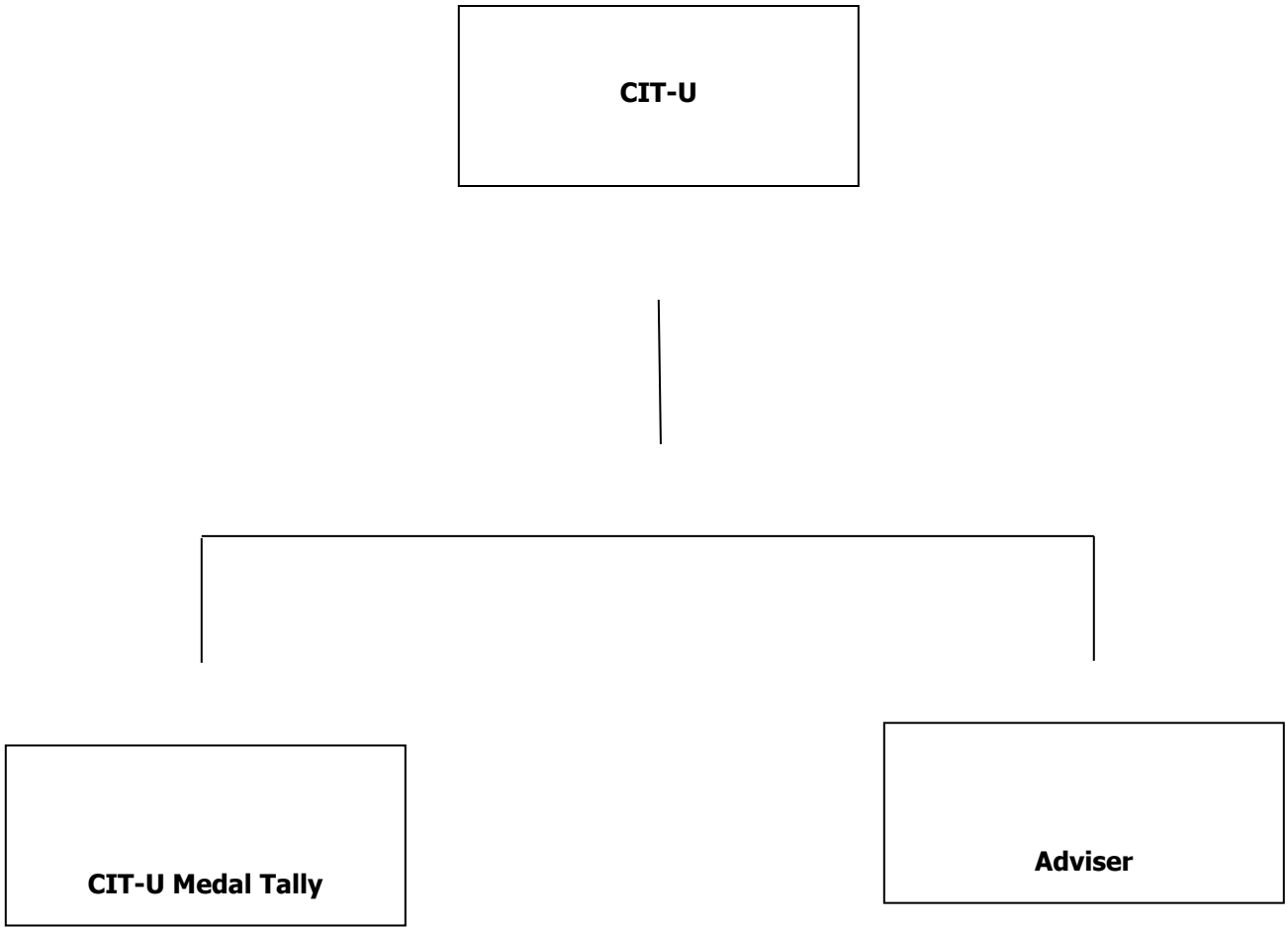
3. Definitions

CIT-U	Cebu Institute of Technology - University
SDD	Software Design Description
SPMP	Software Project Management Plan
SRS	Software Requirements Plan
STD	Software Test Documentation
TBD	To be decided
XML	Extended Markup Language

4. Project organization

a. *External structure*

[Fig 4.1] External Interface



John Vincent Capoy **Mr. Ralph Laviste**

Neil Anthony canton

Noe norman galo

Alwynn batucan

a. *Internal structure*

The medal tally system is to be conducted by our group using the technology learned from the previous subject we've taken up in the CIT-U. The way the group applies what it learns will be checked and advised by the adviser of our software project subject so that the group can produce the qualitative processes and techniques in creating the system software.

4.3. Roles and responsibilities

This section describes the organization of the Inventory management systems as decided by the team during a continuous role definition exercise. It is important to note that this is not a comprehensive list of responsibilities. It is the outcome of an initial team role building session. This section will be updated as the project progresses.

[Table 4.3.1] Roles and Responsibilities

Role	Responsibilities
Documentation Officer	- The one in charge with the document reports. - Software project planning and monitoring - Milestone and schedule planning and monitoring - Set and communicate the team meeting agendas
web developer	- The one in charge of the program for the game. - Plans and designs the templates and User Interface of the software - Write and test codes and then rewrite and refine it if required. - Analyzes the current system status then develops it towards the end of the project.

[Table 4.3.2] Roles and Responsibilities

Role	**Person's Responsible**
Documentation/web developer	john Vincent capoy
web developer	neil Anthony canton
web develop	noe norman galo
tester	alwynn batucan

6. Managerial process plans

a. *Start-up plan*

i. *Estimation plan*

This electronic interactive learning object has been divided into several topics. Though having the same learning approach still each topic has a distinct content tool particularly the dynamic parts of it.

The major topics that was given to our group is the Software Project Management Plan, each topics/sub-topics will be provided a process structure which defines several task and schedules that will be incorporated towards us. Firstly, the certain topic will be basically provided with a storyboard which resembles as a draft or contains the desired output of our project.

Each storyboard topics will be partially facilitated during the timeframe given by our instructor until it will be approved by him. Once the approval is made, we can precede already with the implementation process primarily the static and dynamic contents made from the storyboard in to the final output to our project.

The timeframe that was given to us is only 6 months (July 2014 –October 2014).

There were no necessarily defined costs for this system or project but only time management.

ii. *Staffing plan*

[Table 5.2.2] Staff Plan

Name	Affiliation to project	April-May 2012
John Vincent Capoy	CIT-U Student/Team member	Part Time
Neil Anthony Canton	CIT-U Student/Team member	Part Time
Noe norman galo	CIT-U Student/Team member	Part time
Alwynn batucan	CIT-U Student/Team member	Part time

5.1.3. Resource acquisition plan

Our group will have all access to the software, applications, graphics and all necessary tools available. The course adviser will facilitate the group in determining its support needs and in obtaining the needed tools will giving to us at the ST-buildings facilities.

5.1.4. Project staff training plan

No training for our group will be provided. The group members are already well-knowledgeable enough to their respective disciplines and each has already an experience in working with web development, flash development and its associated phases.

i. Work activities

The group is still working with the assigned work activities given to each one of them and eventually planning to finalize the system output and documentations. Afterwards, once approval will be made, the group will implement primarily functionalities and yet organize everything in our project.

The following confirmed topics/sub-topics that has a storyboard with or without approval already are Software Project Management Plan. Everything was made from first week of July until present.

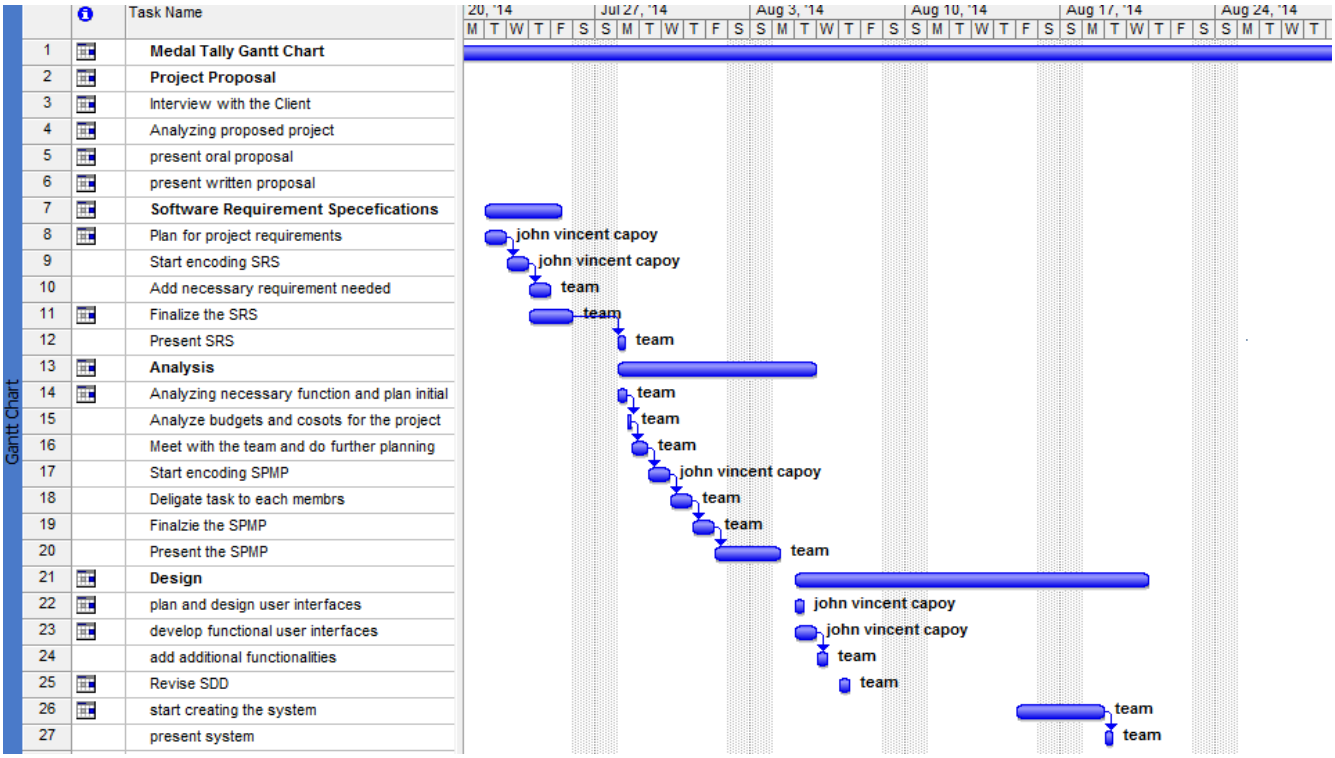
Gantt Chart

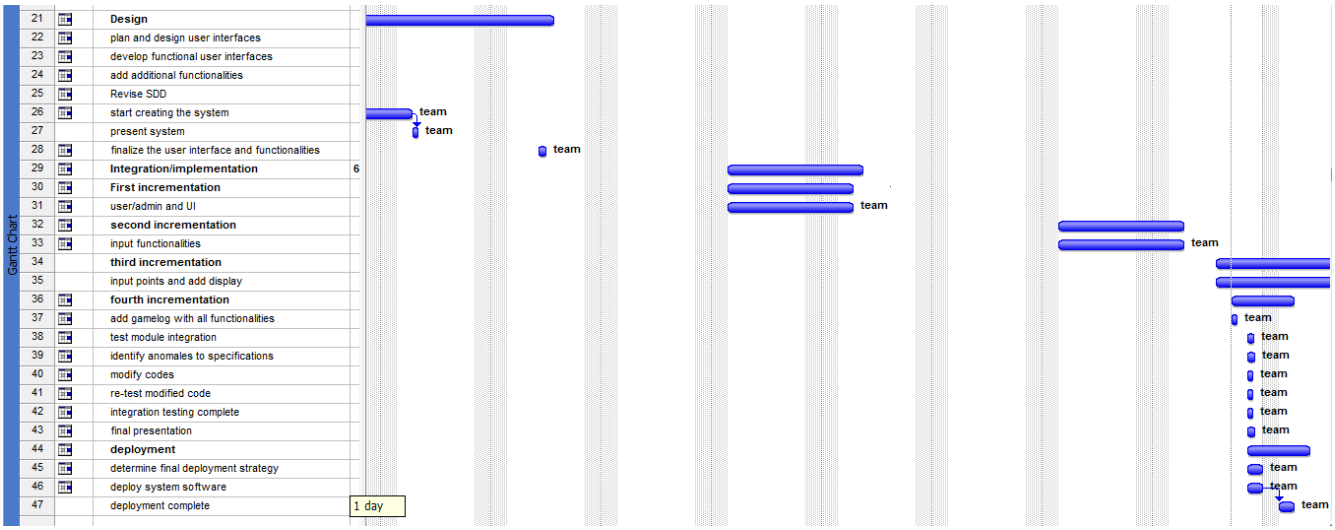
	Task Name	Duration	Start	Finish	Predecessors	Cost	Resource Names
1	**Medal Tally Gantt Chart**	**57 days**	**Fri 7/11/14**	**Mon 9/29/14**		**Php17,910.00**	
2	**Project Proposal**	**1 day**	**Fri 7/11/14**	**Fri 7/11/14**		**Php640.00**	
3	Interview with the Client	8 hrs	Fri 7/11/14	Fri 7/11/14		Php200.00	team
4	Analyzing proposed projec	8 hrs	Fri 7/11/14	Fri 7/11/14		Php200.00	team
5	present oral proposal	4 hrs	Fri 7/11/14	Fri 7/11/14		Php120.00	team
6	present written proposal	4 hrs	Fri 7/11/14	Fri 7/11/14		Php120.00	team
7	**Software Requirement**	**3.5 days**	**Tue 7/22/14**	**Fri 7/25/14**		**Php860.00**	
8	Plan for project requiremer	8 hrs	Tue 7/22/14	Tue 7/22/14		Php200.00	john vincent capoy
9	Start encoding SRS	8 hrs	Wed 7/23/14	Wed 7/23/14	8	Php300.00	john vincent capoy
10	Add necessary requireme	8 hrs	Thu 7/24/14	Thu 7/24/14	9	Php120.00	team
11	Finalize the SRS	16 hrs	Thu 7/24/14	Fri 7/25/14		Php120.00	team
12	Present SRS	2 hrs	Mon 7/28/14	Mon 7/28/14	11	Php120.00	team
13	**Analysis**	**7 days**	**Mon 7/28/14**	**Tue 8/5/14**		**Php1,390.00**	
14	Analyzing necessary func	3 hrs	Mon 7/28/14	Mon 7/28/14		Php200.00	team
15	Analyze budgets and cosc	4 hrs	Mon 7/28/14	Mon 7/28/14	14	Php120.00	team
16	Meet with the team and do	3 hrs	Mon 7/28/14	Tue 7/29/14	15	Php200.00	team
17	Start encoding SPMP	8 hrs	Tue 7/29/14	Wed 7/30/14	16	Php400.00	john vincent capoy
18	Deligate task to each meml	8 hrs	Wed 7/30/14	Thu 7/31/14	17	Php200.00	team
19	Finalzie the SPMP	8 hrs	Thu 7/31/14	Fri 8/1/14	18	Php120.00	team
20	Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14	19	Php120.00	team
21	**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**		**Php1,560.00**	
22	plan and design user inter	3 hrs	Tue 8/5/14	Tue 8/5/14		Php300.00	john vincent capoy
23	develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14		Php500.00	john vincent capoy
24	add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14	23	Php200.00	team
25	Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14		Php120.00	team
26	start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14		Php200.00	team
27	present system	1 hr	Tue 8/19/14	Tue 8/19/14	26	Php120.00	team
28	finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14		Php120.00	team

20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14	19	Php120.00	team
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**		**Php1,560.00**	
22		plan and design user inter	3 hrs	Tue 8/5/14	Tue 8/5/14		Php300.00	john vincent capoy
23		develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14		Php500.00	john vincent capoy
24		add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14	23	Php200.00	team
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14		Php120.00	team
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14		Php200.00	team
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14	26	Php120.00	team
28		finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14		Php120.00	team
29		**Integration/implementa**	**6.75 days**	**Mon 9/8/14**	**Tue 9/16/14**		**Php12,770.00**	
30		**First incrementation**	**1 day?**	**Mon 9/8/14**	**Mon 9/8/14**		**Php2,500.00**	
31		user/admin and UI	1 day?	Mon 9/8/14	Mon 9/8/14		Php2,500.00	team
32		**second incrementation**	**1 day?**	**Mon 9/29/14**	**Mon 9/29/14**		**Php2,000.00**	
33		input functionalities	1 day?	Mon 9/29/14	Mon 9/29/14		Php2,000.00	team
34		**third incrementation**	**1 day?**	**Thu 10/9/14**	**Thu 10/9/14**		**Php3,000.00**	
35		input points and add displa	1 day?	Thu 10/9/14	Thu 10/9/14		Php3,000.00	team
36		**fourth incrementation**	**3 days**	**Fri 10/10/14**	**Mon 10/13/14**	**34**	**Php4,000.00**	
37		add gamelog with all funct	2 hrs	Fri 10/10/14	Fri 10/10/14	35	Php4,000.00	team
38		test module integration	1 day?	Sat 10/11/14	Sat 10/11/14		Php200.00	team
39		identify anomales to speci	1 day?	Sat 10/11/14	Sat 10/11/14		Php300.00	team
40		modify codes	1 day?	Sat 10/11/14	Sat 10/11/14		Php300.00	team
41		re-test modified code	1 day?	Sat 10/11/14	Sat 10/11/14		Php200.00	team
42		integration testing complet	1 day?	Sat 10/11/14	Sat 10/11/14		Php150.00	team
43		final presentation	1 day?	Sat 10/11/14	Sat 10/11/14		Php120.00	team
44		**deployment**	**3 days**	**Sat 10/11/14**	**Tue 10/14/14**		**Php300.00**	
45		determine final deployment	1 day?	Sat 10/11/14	Sat 10/11/14		Php100.00	team
46		deploy system software	1 day?	Sat 10/11/14	Sat 10/11/14		Php100.00	team
47		deployment complete	1 day?	Mon 10/13/14	Mon 10/13/14	46	Php100.00	team

ii. Schedule allocation

The system project duration is constrained to only one semester. The group started from the second week of July 2014with the final deliverables due in the later weeks of October 2014. We will control our schedule with main artifacts.





5.2.3. Resource allocation

As a student in the CIT-U, each team member has a fixed amount of time available for the project. During July to September, each of us is expected to devote a time at least 3-8hours daily to research and help in implement and develop the project, and respective tasks. This work includes time spent working on any laptop or desktop computer, document preparation and inspection, tool development whether individually or by group working with it.

The personnel resources available for the duration of the project along with the schedule constraints are:

2 programmers for any amount of hours daily each (from July 2014 - September 2014).

1		**Medal Tally Gantt Chart**	**57 days**	**Fri 7/11/14**	**Mon 9/29/14**
2		**Project Proposal**	**1 day**	**Fri 7/11/14**	**Fri 7/11/14**
3		Interview with the Client	8 hrs	Fri 7/11/14	Fri 7/11/14
4		Analyzing proposed project	8 hrs	Fri 7/11/14	Fri 7/11/14
5		present oral proposal	4 hrs	Fri 7/11/14	Fri 7/11/14
6		present written proposal	4 hrs	Fri 7/11/14	Fri 7/11/14
7		**Software Requirement Speceficati**	**3.5 days**	**Tue 7/22/14**	**Fri 7/25/14**
8		Plan for project requirements	8 hrs	Tue 7/22/14	Tue 7/22/14
9		Start encoding SRS	8 hrs	Wed 7/23/14	Wed 7/23/14
10		Add necessary requirement needed	8 hrs	Thu 7/24/14	Thu 7/24/14
11		Finalize the SRS	16 hrs	Thu 7/24/14	Fri 7/25/14
12		Present SRS	2 hrs	Mon 7/28/14	Mon 7/28/14
13		**Analysis**	**7 days**	**Mon 7/28/14**	**Tue 8/5/14**
14		Analyzing necessary function and plar	3 hrs	Mon 7/28/14	Mon 7/28/14
15		Analyze budgets and cosots for the pr	4 hrs	Mon 7/28/14	Mon 7/28/14
16		Meet with the team and do further plan	3 hrs	Mon 7/28/14	Tue 7/29/14
17		Start encoding SPMP	8 hrs	Tue 7/29/14	Wed 7/30/14
18		Deligate task to each membrs	8 hrs	Wed 7/30/14	Thu 7/31/14
19		Finalzie the SPMP	8 hrs	Thu 7/31/14	Fri 8/1/14
20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**
22		plan and design user interfaces	3 hrs	Tue 8/5/14	Tue 8/5/14
23		develop functional user interfaces	8 hrs	Tue 8/5/14	Tue 8/5/14
24		add additional functionalities	4 hrs	Wed 8/6/14	Wed 8/6/14
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14
28		finalize the user interface and function	4 hrs	Wed 8/20/14	Wed 8/20/14

20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**
22		plan and design user interf	3 hrs	Tue 8/5/14	Tue 8/5/14
23		develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14
24		add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14
28		finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14
29		**Integration/implementa**	**6.75 days**	**Mon 9/8/14**	**Tue 9/16/14**
30		**First incrementation**	**1 day?**	**Mon 9/8/14**	**Mon 9/8/14**
31		user/admin and UI	1 day?	Mon 9/8/14	Mon 9/8/14
32		**second incrementation**	**1 day?**	**Mon 9/29/14**	**Mon 9/29/14**
33		input functionalities	1 day?	Mon 9/29/14	Mon 9/29/14
34		**third incrementation**	**1 day?**	**Thu 10/9/14**	**Thu 10/9/14**
35		input points and add displa	1 day?	Thu 10/9/14	Thu 10/9/14
36		**fourth incrementation**	**3 days**	**Fri 10/10/14**	**Mon 10/13/14**
37		add gamelog with all funct	2 hrs	Fri 10/10/14	Fri 10/10/14
38		test module integration	1 day?	Sat 10/11/14	Sat 10/11/14
39		identify anomales to speci	1 day?	Sat 10/11/14	Sat 10/11/14
40		modify codes	1 day?	Sat 10/11/14	Sat 10/11/14
41		re-test modified code	1 day?	Sat 10/11/14	Sat 10/11/14
42		integration testing complet	1 day?	Sat 10/11/14	Sat 10/11/14
43		final presentation	1 day?	Sat 10/11/14	Sat 10/11/14
44		**deployment**	**3 days**	**Sat 10/11/14**	**Tue 10/14/14**
45		determine final deployment	1 day?	Sat 10/11/14	Sat 10/11/14
46		deploy system software	1 day?	Sat 10/11/14	Sat 10/11/14
47		deployment complete	1 day?	Mon 10/13/14	Mon 10/13/14

5.2.4. Budget allocation

[Table 5.2.5] Budget allocation

Gantt Chart

	Task Name	Fixed Cost	Fixed Cost Accrual	Cost
1	**Medal Tally Gantt Chart**	Php17,910.00	Prorated	**Php17,910.00**
2	**Project Proposal**	Php640.00	Prorated	**Php640.00**
3	Interview with the Client	Php200.00	Prorated	Php200.00
4	Analyzing proposed project	Php200.00	Prorated	Php200.00
5	present oral proposal	Php120.00	Prorated	Php120.00
6	present written proposal	Php120.00	Prorated	Php120.00
7	**Software Requirement Specefications**	Php860.00	Prorated	**Php860.00**
8	Plan for project requirements	Php200.00	Prorated	Php200.00
9	Start encoding SRS	Php300.00	Prorated	Php300.00
10	Add necessary requirement needed	Php120.00	Prorated	Php120.00
11	Finalize the SRS	Php120.00	Prorated	Php120.00
12	Present SRS	Php120.00	Prorated	Php120.00
13	**Analysis**	Php1,390.00	Prorated	**Php1,390.00**
14	Analyzing necessary function and plan initial	Php200.00	Prorated	Php200.00
15	Analyze budgets and cosots for the project	Php120.00	Prorated	Php120.00
16	Meet with the team and do further planning	Php200.00	Prorated	Php200.00
17	Start encoding SPMP	Php400.00	Prorated	Php400.00
18	Deligate task to each membrs	Php200.00	Prorated	Php200.00
19	Finalzie the SPMP	Php120.00	Prorated	Php120.00
20	Present the SPMP	Php120.00	Prorated	Php120.00
21	**Design**	Php1,560.00	Prorated	**Php1,560.00**
22	plan and design user interfaces	Php300.00	Prorated	Php300.00
23	develop functional user interfaces	Php500.00	Prorated	Php500.00
24	add additional functionalities	Php200.00	Prorated	Php200.00
25	Revise SDD	Php120.00	Prorated	Php120.00
26	start creating the system	Php200.00	Prorated	Php200.00
27	present system	Php120.00	Prorated	Php120.00
28	finalize the user interface and functionalities	Php120.00	Prorated	Php120.00
29	**Integration/implementation**	Php12,770.00	Prorated	**Php12,770.00**

Gantt Chart

20	Present the SPMP	Php120.00	Prorated	Php120.00
21	**Design**	Php1,560.00	Prorated	**Php1,560.00**
22	plan and design user interfaces	Php300.00	Prorated	Php300.00
23	develop functional user interfaces	Php500.00	Prorated	Php500.00
24	add additional functionalities	Php200.00	Prorated	Php200.00
25	Revise SDD	Php120.00	Prorated	Php120.00
26	start creating the system	Php200.00	Prorated	Php200.00
27	present system	Php120.00	Prorated	Php120.00
28	finalize the user interface and functionalities	Php120.00	Prorated	Php120.00
29	**Integration/implementation**	Php12,770.00	Prorated	**Php12,770.00**
30	**First incrementation**	Php2,500.00	Prorated	**Php2,500.00**
31	user/admin and UI	Php2,500.00	Prorated	Php2,500.00
32	**second incrementation**	Php2,000.00	Prorated	**Php2,000.00**
33	input functionalities	Php2,000.00	Prorated	Php2,000.00
34	**third incrementation**	Php3,000.00	Prorated	**Php3,000.00**
35	input points and add display	Php3,000.00	Prorated	Php3,000.00
36	**fourth incrementation**	Php4,000.00	Prorated	**Php4,000.00**
37	add gamelog with all functionalities	Php4,000.00	Prorated	Php4,000.00
38	test module integration	Php200.00	Prorated	Php200.00
39	identify anomales to specifications	Php300.00	Prorated	Php300.00
40	modify codes	Php300.00	Prorated	Php300.00
41	re-test modified code	Php200.00	Prorated	Php200.00
42	integration testing complete	Php150.00	Prorated	Php150.00
43	final presentation	Php120.00	Prorated	Php120.00
44	**deployment**	Php300.00	Prorated	**Php300.00**
45	determine final deployment strategy	Php100.00	Prorated	Php100.00
46	deploy system software	Php100.00	Prorated	Php100.00
47	deployment complete	Php100.00	Prorated	Php100.00

Cost of Documents Presentation or Report is due to printing or hardcopies.

Since software tools has already been provided and ready to access at all time, due costs is N/A.

5.3.1. Requirements control plan

Requirement will be managed or assist in the use case description of SRS as the requirements are changed. Every use case description will be manage by the documentation members.

5.3.2. Schedule control plan

The Group's adviser will have the schedule in every project document. The leader from the group will be responsible for gathering the individual tasks for each team member and making the status report with the guidance of the adviser. If schedule is not on track, we will have to spend extra time on the project to make sure that there will no more delays in dealing with the project.

5.3.3. Budget control plan

This System project has no plan on budget control yet. If the project needs substantial amount, the team will spend extra money to deliver the project.

5.3.4. Quality control plan

The adviser/facilitator already represent as the quality manager. She will generate a separate Quality Control Plan document but only depends with the set standards.

A weekly meetings and reviews at every Monday and Friday for every phase so the group's output will be in quality.

In addition, Group adviser will monitor quality control throughout the project by the mentoring and advising. The adviser is encouraged to review the group's work products and to ask question to determine the health and progress of the project.

5.3.5. Reporting plan

The group will have to submit the require documents/system analysis weekly. in addition to that they will have consultations with their respective adviser on the what they submitted on that week.

For client, we will send status and progress reports in every weekend.

5.3.6. Metrics collection plan

The group member will cooperate and submit their respective parts of the document in every scheduled due date as well as the programmer's individual progress. Each group members will also report and help on task assigned, tasks done and not done, problems, hours planned and future plans at every status meeting whether it is a group meeting or with the adviser.

5.3.7 Risk management plan

The group lead will generate a separate Risk Management Plan document at any necessary means. Risk will be identified at the beginning of each phase. The group lead will assemble them into prioritizing the risks list. The group will monitor priority risks every week. All risks will be documented by the group.

5.3.8 Project closeout plan

Group will ensure the proper closeout of the project in late octoberr 2014. And all the documents will also be submitted in the said date or later.

6. Technical process plans

6.1 Process Model

The Group will use Net Beans IDE for software implementation of the project. Group 4 will provide an iteration plan at the beginning of the iteration. The process will be applied to specify the phase of the implementation for the project.

6.2 Methods, tools, and techniques

The methods and techniques listed in this table will be evaluated and applied in specific areas of the project as appropriate:

[Table 6.2.1] Methods and Techniques

Category	**Methods and Techniques**
Requirements Elicitation	- Elicitation from existing previous modules/ topic from the interactive learning object website. - Meetings - Status report presentation - Brainstorming
Formal Specification and Analysis	- Formal models using UML to model structural aspects of the requirements and design - Use cases to define requirements that should be place in SRS document
Document presentation report	- Weekly partial document report for revisions of documents for completion.
Estimation	- Function Point method for conversion from Function Point count to effort may be used for size estimation and project scope definition.

[Table 6.2.2] Methods and Techniques

Category	**Tools**
Operating System	Windows XP, Vista, 7, 8
Development languages	php, html5, web base
Storyboard Application	Microsoft PowerPoint
Document	All documents will be written in Microsoft Word

6.3 Infrastructure Plan

The 7H Water Supply will primarily be considered for the development of the project. The group has access to the company's computer. The group will be installing application required by our system software with the authority of the personnel in charge.

The group's adviser will be informed immediately in the changes that group's made.

6.3 Product Acceptance Plan

The client with signing appropriate acceptance document accepts every milestone of the project formally. At the end of each phase the client will install the product and perform an acceptance test. This may result in additional requests for changes and improvements.

7. Supporting process plans

7.1. Configuration management plan

Group's Adviser is part and responsible of a separate document and it will be maintained.

7.2. Verification and validation plan

Several tasks collectively make up continuing activities that go across the different life cycle phases. These general activities are traceability analysis, evaluation, interface analysis, and testing. These activities are horizontal threads that tie together the subsequent phase activities and allow verification to be more effectively conducted.

Traceability analysis

The traceability is the ability to identify the relationships between originating requirements and their resulting system features. It permits tracking forward or backward through the network of interrelationships that are created as requirements are decomposed and refined through a system's life cycle. Traceability allows verification of the properties set forth in the concept and that requirement specifications have been carried forward to the design specification, implemented in the code, included in the test plan and cases, and provided to the customer and user in the resulting system.

Evaluation

Evaluation ascertains the value or worth of an item and help to assure that a system meets its specifications. Evaluations are performed by many persons across all life cycle phases, on both interim and final software products, and may be either a comprehensive or selective assessment of a system. Evaluations are used through all phases and for all type of software products, including user documents, manuals, and other project documents. These may be of many forms, such as text or graphic representations, and in various media, such as paper, magnetic tape, diskette, and computer files. This range of product types and forms requires a large variety of techniques for performing and managing software evaluations.

Interface analysis

When information is passed across a boundary, there is always the possibility of losing some information or alerting the information content. The task of interface analysis serves to ensure the completeness, accuracy, and consistency of these interfaces. Interface requirements at the design and implementation phases should be identified analyzed at the functional, physical, and data interface level. The goal of interface analysis is to evaluate the specific software deliverables (e.g., requirements, design, code) for correct, consistent, complete, and accurate interpretation of the interface requirements.

Testing

In the context of software verification and validation, testing can be defined as the testing that is performed in support of the V&V objectives. These objectives may differ from those of the developer. Testing is performed at several points in the life cycle, starting from the requirement phase up to the test phase. The various test activities are listed below:

Component Testing

Testing conducted to verify the implementation of the design for one software elements or a collection of software elements

Integrating Testing

An orderly progression of testing in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated.

System Testing

The process of testing an integrated hardware and software system to verify that the system meets its specified requirements

Acceptance Test

Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system

This section explains out V&V plan for each phase of software development.

[Table 7.2.2] Verification and validation plan

Phase	V&V Input	V&V Tasks	V&V Output
Requirements	SRS Interface requirements documentation User documentation	Requirements traceability analysis Requirements evaluation Requirements interface analysis Test plan generation	Requirements phase tasks reporting Test plan : System - Acceptance
Design	SRS Interface requirements documentation Interface design documentation User documentation	Design traceability analysis Design evaluation Interface analysis Test plan generation Test design generation	Design phase task reporting Test plan Component Integration Test design Component Integration System acceptance
Implementation	Source Code listing Executable code Interface design documentation User documentation	Code traceability analysis Code evaluation Interface analysis Documentation	Implementation phase task reporting Test cases Component Integration

		evaluation Test case generation Test procedure generation Component test execution	System Acceptance Test procedure Component Integration - System
Test	Source code listing Executable code User documentation	Test procedure generation Integration test execution System test execution Acceptance test execution	Test phase task reporting Test procedure Acceptance Anomaly report V&V phase summary
Installation and Checkout	Installation package	Installation configuration audit V&V final report generation	Installation and checkout phase task reporting

7.3. Documentation plan

There are a number of documents that will be produced during the lifetime of the project. All documents are responsibility of the project team members. The lists of documents that will be created and maintained under version control include:

Project Proposal

Confirmation Letter

Software Requirements Specification (SRS) – defines the functionality that is required by the client.

Software Project Management Plan (SPMP) – defines the project management plan.

Software Test Documentation (STD) – defines the testing of the project on documentation.

Software Design Description (SDD) – defines the project design description.

Status Report

Use Case Diagram

Use Case Description

7.4. Quality assurance plan

The group's project will be assured to fulfill the commitment to the software process and the software product as specified in the requirement specification by the documents made for the project. The scope of this project is only allowable and limited for this Inventory and Management software, so quality and specification is need for this project to be running.

7.5. Reviews and audits

The SPMP specifies the plan, schedule and methods to be used in conducting the project reviews

and audits. So far, the only products that were created are documents and the initiation of the project. It is expected that in the future the details about the review and audits will be maintained within the team's QA Plan. And most of the reviews will be provided at the final stage of this project.

7.6. Problem resolution plan

For the problems encountered, it will be noted in order to improve the project. A series of test will be taken to thoroughly check the software. Evaluate and retest all the information and data gathered for the quality and stability of the software.

7.7. Subcontractor management plan

Not applicable to group's project.

7.8. Process improvement plan

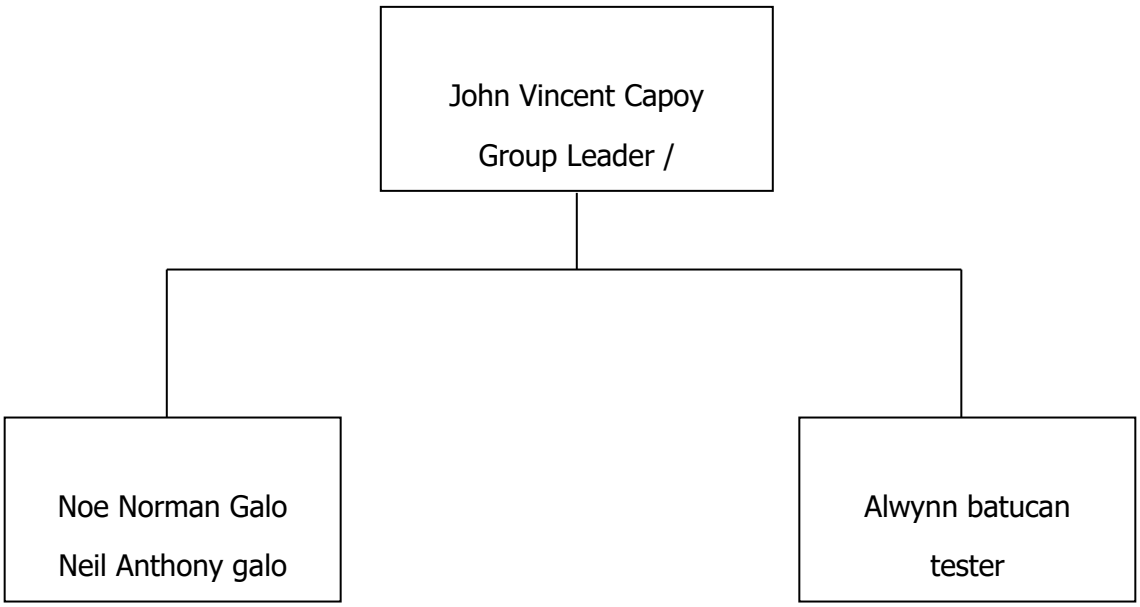
Process improvement will be done as a part of the final project evaluation and "lessons learned" phase. At that time the process improvement plan will be created and will be implemented.

8.0 Additional Plans

When Additional Plans are needed, the group lead will assemble the group for a meeting

9. Plan Annexes

APPENDIX A – ORGANIZATION CHART INTERNAL



APPENDIX B – WORK ACTIVITIES

Gantt Chart

	Task Name	Duration	Start	Finish	Predecessors	Cost	Resource Names
1	**Medal Tally Gantt Chart**	**57 days**	**Fri 7/11/14**	**Mon 9/29/14**		**Php17,910.00**	
2	**Project Proposal**	**1 day**	**Fri 7/11/14**	**Fri 7/11/14**		**Php640.00**	
3	Interview with the Client	8 hrs	Fri 7/11/14	Fri 7/11/14		Php200.00	team
4	Analyzing proposed projec	8 hrs	Fri 7/11/14	Fri 7/11/14		Php200.00	team
5	present oral proposal	4 hrs	Fri 7/11/14	Fri 7/11/14		Php120.00	team
6	present written proposal	4 hrs	Fri 7/11/14	Fri 7/11/14		Php120.00	team
7	**Software Requirement**	**3.5 days**	**Tue 7/22/14**	**Fri 7/25/14**		**Php860.00**	
8	Plan for project requiremer	8 hrs	Tue 7/22/14	Tue 7/22/14		Php200.00	john vincent capoy
9	Start encoding SRS	8 hrs	Wed 7/23/14	Wed 7/23/14	8	Php300.00	john vincent capoy
10	Add necessary requireme	8 hrs	Thu 7/24/14	Thu 7/24/14	9	Php120.00	team
11	Finalize the SRS	16 hrs	Thu 7/24/14	Fri 7/25/14		Php120.00	team
12	Present SRS	2 hrs	Mon 7/28/14	Mon 7/28/14	11	Php120.00	team
13	**Analysis**	**7 days**	**Mon 7/28/14**	**Tue 8/5/14**		**Php1,390.00**	
14	Analyzing necessary func	3 hrs	Mon 7/28/14	Mon 7/28/14		Php200.00	team
15	Analyze budgets and cosc	4 hrs	Mon 7/28/14	Mon 7/28/14	14	Php120.00	team
16	Meet with the team and do	3 hrs	Mon 7/28/14	Tue 7/29/14	15	Php200.00	team
17	Start encoding SPMP	8 hrs	Tue 7/29/14	Wed 7/30/14	16	Php400.00	john vincent capoy
18	Deligate task to each meml	8 hrs	Wed 7/30/14	Thu 7/31/14	17	Php200.00	team
19	Finalzie the SPMP	8 hrs	Thu 7/31/14	Fri 8/1/14	18	Php120.00	team
20	Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14	19	Php120.00	team
21	**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**		**Php1,560.00**	
22	plan and design user inter	3 hrs	Tue 8/5/14	Tue 8/5/14		Php300.00	john vincent capoy
23	develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14		Php500.00	john vincent capoy
24	add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14	23	Php200.00	team
25	Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14		Php120.00	team
26	start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14		Php200.00	team
27	present system	1 hr	Tue 8/19/14	Tue 8/19/14	26	Php120.00	team
28	finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14		Php120.00	team

20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14	19	Php120.00	team
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**		**Php1,560.00**	
22		plan and design user inter	3 hrs	Tue 8/5/14	Tue 8/5/14		Php300.00	john vincent capoy
23		develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14		Php500.00	john vincent capoy
24		add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14	23	Php200.00	team
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14		Php120.00	team
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14		Php200.00	team
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14	26	Php120.00	team
28		finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14		Php120.00	team
29		**Integration/implementa**	**6.75 days**	**Mon 9/8/14**	**Tue 9/16/14**		**Php12,770.00**	
30		**First incrementation**	**1 day?**	**Mon 9/8/14**	**Mon 9/8/14**		**Php2,500.00**	
31		user/admin and UI	1 day?	Mon 9/8/14	Mon 9/8/14		Php2,500.00	team
32		**second incrementation**	**1 day?**	**Mon 9/29/14**	**Mon 9/29/14**		**Php2,000.00**	
33		input functionalities	1 day?	Mon 9/29/14	Mon 9/29/14		Php2,000.00	team
34		**third incrementation**	**1 day?**	**Thu 10/9/14**	**Thu 10/9/14**		**Php3,000.00**	
35		input points and add displa	1 day?	Thu 10/9/14	Thu 10/9/14		Php3,000.00	team
36		**fourth incrementation**	**3 days**	**Fri 10/10/14**	**Mon 10/13/14**	**34**	**Php4,000.00**	
37		add gamelog with all funct	2 hrs	Fri 10/10/14	Fri 10/10/14	35	Php4,000.00	team
38		test module integration	1 day?	Sat 10/11/14	Sat 10/11/14		Php200.00	team
39		identify anomales to speci	1 day?	Sat 10/11/14	Sat 10/11/14		Php300.00	team
40		modify codes	1 day?	Sat 10/11/14	Sat 10/11/14		Php300.00	team
41		re-test modified code	1 day?	Sat 10/11/14	Sat 10/11/14		Php200.00	team
42		integration testing complet	1 day?	Sat 10/11/14	Sat 10/11/14		Php150.00	team
43		final presentation	1 day?	Sat 10/11/14	Sat 10/11/14		Php120.00	team
44		**deployment**	**3 days**	**Sat 10/11/14**	**Tue 10/14/14**		**Php300.00**	
45		determine final deployment	1 day?	Sat 10/11/14	Sat 10/11/14		Php100.00	team
46		deploy system software	1 day?	Sat 10/11/14	Sat 10/11/14		Php100.00	team
47		deployment complete	1 day?	Mon 10/13/14	Mon 10/13/14	46	Php100.00	team

APPENDIX C – RESOURCE ALLOCATION

1		**Medal Tally Gantt Chart**	**57 days**	**Fri 7/11/14**	**Mon 9/29/14**
2		**Project Proposal**	**1 day**	**Fri 7/11/14**	**Fri 7/11/14**
3		Interview with the Client	8 hrs	Fri 7/11/14	Fri 7/11/14
4		Analyzing proposed project	8 hrs	Fri 7/11/14	Fri 7/11/14
5		present oral proposal	4 hrs	Fri 7/11/14	Fri 7/11/14
6		present written proposal	4 hrs	Fri 7/11/14	Fri 7/11/14
7		**Software Requirement Speceficati**	**3.5 days**	**Tue 7/22/14**	**Fri 7/25/14**
8		Plan for project requirements	8 hrs	Tue 7/22/14	Tue 7/22/14
9		Start encoding SRS	8 hrs	Wed 7/23/14	Wed 7/23/14
10		Add necessary requirement needed	8 hrs	Thu 7/24/14	Thu 7/24/14
11		Finalize the SRS	16 hrs	Thu 7/24/14	Fri 7/25/14
12		Present SRS	2 hrs	Mon 7/28/14	Mon 7/28/14
13		**Analysis**	**7 days**	**Mon 7/28/14**	**Tue 8/5/14**
14		Analyzing necessary function and plar	3 hrs	Mon 7/28/14	Mon 7/28/14
15		Analyze budgets and cosots for the pr	4 hrs	Mon 7/28/14	Mon 7/28/14
16		Meet with the team and do further plan	3 hrs	Mon 7/28/14	Tue 7/29/14
17		Start encoding SPMP	8 hrs	Tue 7/29/14	Wed 7/30/14
18		Deligate task to each membrs	8 hrs	Wed 7/30/14	Thu 7/31/14
19		Finalzie the SPMP	8 hrs	Thu 7/31/14	Fri 8/1/14
20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**
22		plan and design user interfaces	3 hrs	Tue 8/5/14	Tue 8/5/14
23		develop functional user interfaces	8 hrs	Tue 8/5/14	Tue 8/5/14
24		add additional functionalities	4 hrs	Wed 8/6/14	Wed 8/6/14
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14
28		finalize the user interface and function	4 hrs	Wed 8/20/14	Wed 8/20/14

20		Present the SPMP	8 hrs	Fri 8/1/14	Mon 8/4/14
21		**Design**	**12 days**	**Tue 8/5/14**	**Wed 8/20/14**
22		plan and design user inter	3 hrs	Tue 8/5/14	Tue 8/5/14
23		develop functional user int	8 hrs	Tue 8/5/14	Tue 8/5/14
24		add additional functionalitie	4 hrs	Wed 8/6/14	Wed 8/6/14
25		Revise SDD	4 hrs	Thu 8/7/14	Thu 8/7/14
26		start creating the system	16 hrs	Fri 8/15/14	Mon 8/18/14
27		present system	1 hr	Tue 8/19/14	Tue 8/19/14
28		finalize the user interface	4 hrs	Wed 8/20/14	Wed 8/20/14
29		**Integration/implementa**	**6.75 days**	**Mon 9/8/14**	**Tue 9/16/14**
30		**First incrementation**	**1 day?**	**Mon 9/8/14**	**Mon 9/8/14**
31		user/admin and UI	1 day?	Mon 9/8/14	Mon 9/8/14
32		**second incrementation**	**1 day?**	**Mon 9/29/14**	**Mon 9/29/14**
33		input functionalities	1 day?	Mon 9/29/14	Mon 9/29/14
34		**third incrementation**	**1 day?**	**Thu 10/9/14**	**Thu 10/9/14**
35		input points and add displa	1 day?	Thu 10/9/14	Thu 10/9/14
36		**fourth incrementation**	**3 days**	**Fri 10/10/14**	**Mon 10/13/14**
37		add gamelog with all funct	2 hrs	Fri 10/10/14	Fri 10/10/14
38		test module integration	1 day?	Sat 10/11/14	Sat 10/11/14
39		identify anomales to speci	1 day?	Sat 10/11/14	Sat 10/11/14
40		modify codes	1 day?	Sat 10/11/14	Sat 10/11/14
41		re-test modified code	1 day?	Sat 10/11/14	Sat 10/11/14
42		integration testing complet	1 day?	Sat 10/11/14	Sat 10/11/14
43		final presentation	1 day?	Sat 10/11/14	Sat 10/11/14
44		**deployment**	**3 days**	**Sat 10/11/14**	**Tue 10/14/14**
45		determine final deployment	1 day?	Sat 10/11/14	Sat 10/11/14
46		deploy system software	1 day?	Sat 10/11/14	Sat 10/11/14
47		deployment complete	1 day?	Mon 10/13/14	Mon 10/13/14

10. Index

Software Design Description

for

CIT-U Medal Tally

Signature

Members	Roles	Signature
John Vincent Capoy	Documentation/Web Developer	
Noe norman galo	Web developer	
Niel Anthony Canton	Web Developer	
Alwynn Batucan	System tester	

Change History

VERSION	DATE	AUTHOR	CHANGES
1.0	August 20, 2014	John Vincent Capoy Neil Anthony Canton	Initial Version
2.0	August 27, 2014	John Vincent Capoy Neil Anthony Canton	Final Version

Preface

Software Design Description is documented to show all the possible design to use in the system. This is to show the different design requirements in compliance to the specific requirements in the system. All modules in designing the system must be detailed in this document.

This will specify the different descriptions on how will the components will be constructed, on how all the applications must be implemented and how will it be organized according to its relationship.

Mainly, this is serves as a reference on how will the project teams organize the data flows for implementations. It will also be the basis for the program coding construction.

Table of Contents

List of Figures

List of Tables

8. Introduction

8.1. Purpose

Software Design Description is provided to detail how the software requirements should be implemented. This acts as a blueprint to the programmers to follow. Once this was implemented, this will be the baseline for limiting changes of the scope of the project. This document helps determine if all the requirements are meet.

8.2. Scope

Since this Software Design Description acts as developer blueprint, all the architectural aspects are discuss. Interfaces design must be fully presented, must be detailed for easy reviews and development progress is easy to be picture out. This is intended to the project team for a better understanding to keep them in focus and aligned in working the project.

8.3. Definitions and Acronyms

Table 1.3.1 ***Definitions and Acronyms***

Term	Definition
CIT-U	Cebu Institute of Technology – University
	An object that embodies method that perfectly complements the lecture-type approach. It completes the whole picture the students need.
CCS	College of Computer Studies
	An organization assisting students needs.
CAMS	Customer Account Management System
	A system controls customer information.
SRS	Software Requirements Specifications
	A module wherein all requirements are being specified for project planning.

9. References

1016-1987 - IEEE Recommended Practice for Software Design Descriptions. 1987. doi:10.1109/IEEESTD.1987.122643. ISBN 0-7381-0402-7.

1016-1998 - IEEE Recommended Practice for Software Design Descriptions. 1998. doi:10.1109/IEEESTD.1998.88828. ISBN 0-7381-1455-3.

1016-2009 - IEEE Standard for Information Technology-Systems Design- Software Design Descriptions. 2009. doi:10.1109/IEEESTD.2009.5167255. ISBN 978-0-7381-5926-3.

10. Decomposition Description

10.1. Module Decomposition

10.1.1. Module 1 Description

This module determines the actions of the companies administrator on their database on how will they manage and organize the Operations perform between the user and the database system.

Table 3.1.1.1

Admin Module Description

Identification	Admin
Type	Module
Purpose	To allow the Admin by performing functions such as adding order input from customer and updating customer Operations.

10.1.2. Module 2 Description

This module describes all the possible actions that a customer will perform. Where he/she can only allow to use the given process on how can he/she order and view his/her account and Operations.

Table 3.1.1.2

User Module Description

Identification	Customer
Type	Module
Purpose	This module provides a way for the customer to order, to pay, and to view recent Operations that he or she performs.

10.2. Concurrent Process Decomposition

10.2.1. Process 1 Description

User Operation Process

Table 3.2.1.1

Operation View

Identification	Operation View
Type	Web page
Purpose	It summarizes all the Operation that has been made by the customer.

10.2.2. Process 2 Description

Admin Database Access

Table 3.2.2.1

Operations Table

Identification	Operations
Type	Table
Purpose	Wherein all user medals and Operations are recorded including dates and total of medals.

Table 3.2.2.4

Operations Table

Identification	Medal history
Type	Table
Purpose	Wherein all updated user medals are in details and shown.

10.3. Data Decomposition

10.3.1. Data Entity 2 Description

Table 3.3.2.1 *Medals*

Medals		
Field Name	**Description**	**Data Type**
Medal_id	field name that represents Medal id	int
Medal_name	field name that represents Medal name	Varchar
Medal_price	field name that represents Medal price	Varchar
Medal_brand	field name that represents Medal brand	Varchar
Medal_type	field name that represents Medal type	varchar

Table 3.3.2.2 *Operations*

Operations		
Field Name	**Description**	**Data Type**

Operation_id	field name that represents Operation	int
User_id	field name that represents users id	int
Customer_name	field name that represents name of the customer	Varchar
Operation_date	field name that represents date of Operation	date
Due_date	Field name represents the due	date

11. Dependency Description

11.1. Inter-module Dependencies

4.1.1. Independent Modules

Admin module:

Dependencies	Reason(s)
None	None

This module will not rely on any other forum module. They may be however expecting certain tools to be present. The presence of these tools will have to be checked when installing.

11.2. Inter-process Dependencies

Dependencies	Reason(s)
None	None

Data Storing Process: All the details from the Medal won't display or can't be processed if all data are not inputted in the database.

11.3. Data Dependencies

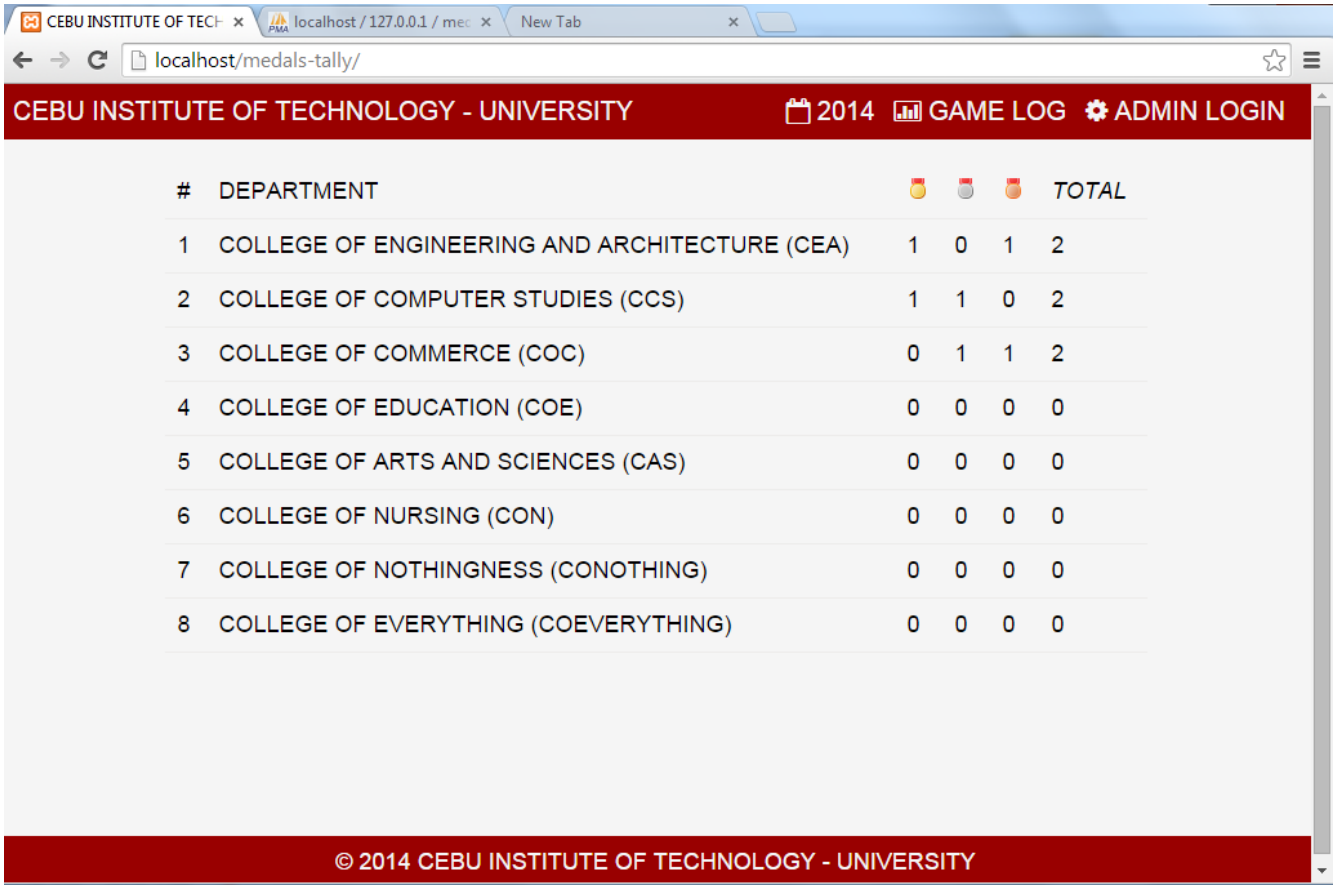
Dependencies	Reason(s)
None	None

Operation Process: This process is dependent to login or sigh up account process. This won't perform if the end user has not been logged in. Interface Description

11.4. Module Interface

11.4.1. Module 1 Description

Figure 5.1.1 Dashboard View



Description: Counts medal of the user

Figure 5.1.6 Operation View

Description: This is shown in regards to all the Operation of a customer

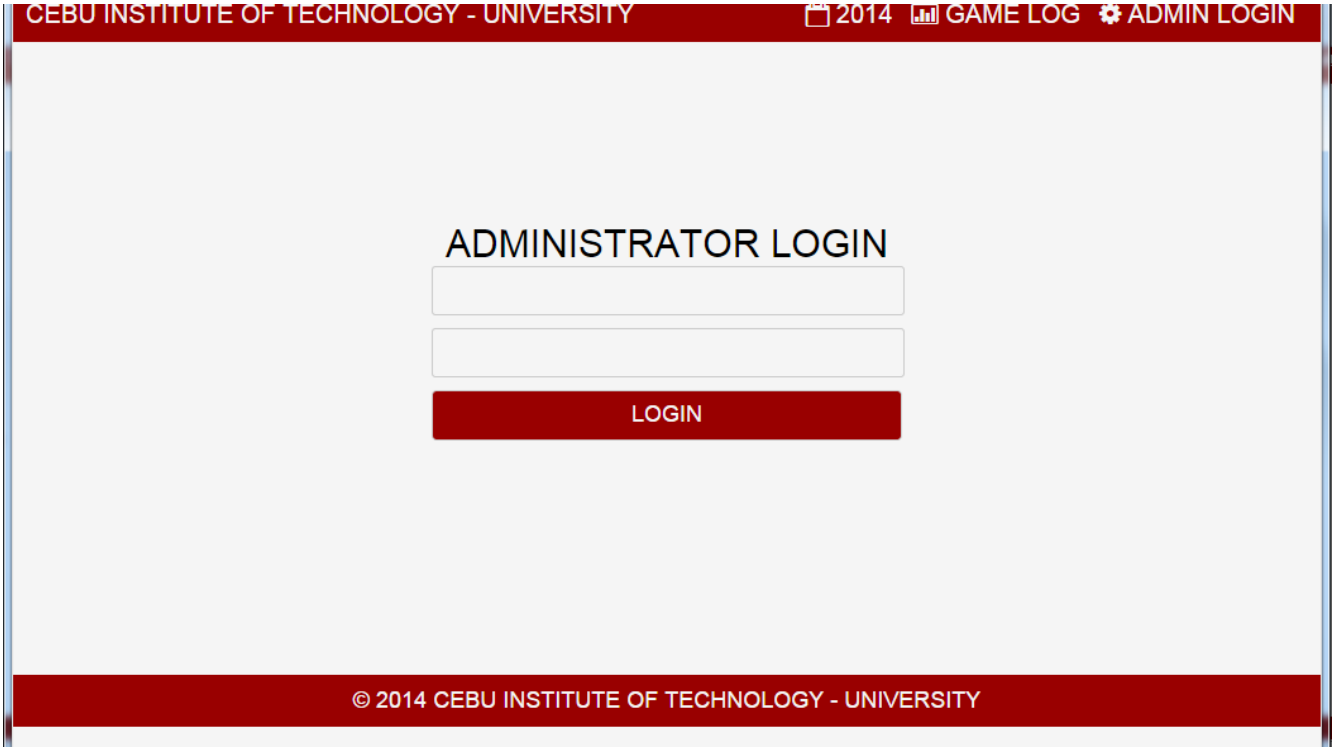
Figure 5.1.7 Receipt View

11.4.2. Module 2 Description

11.5. Process Interface

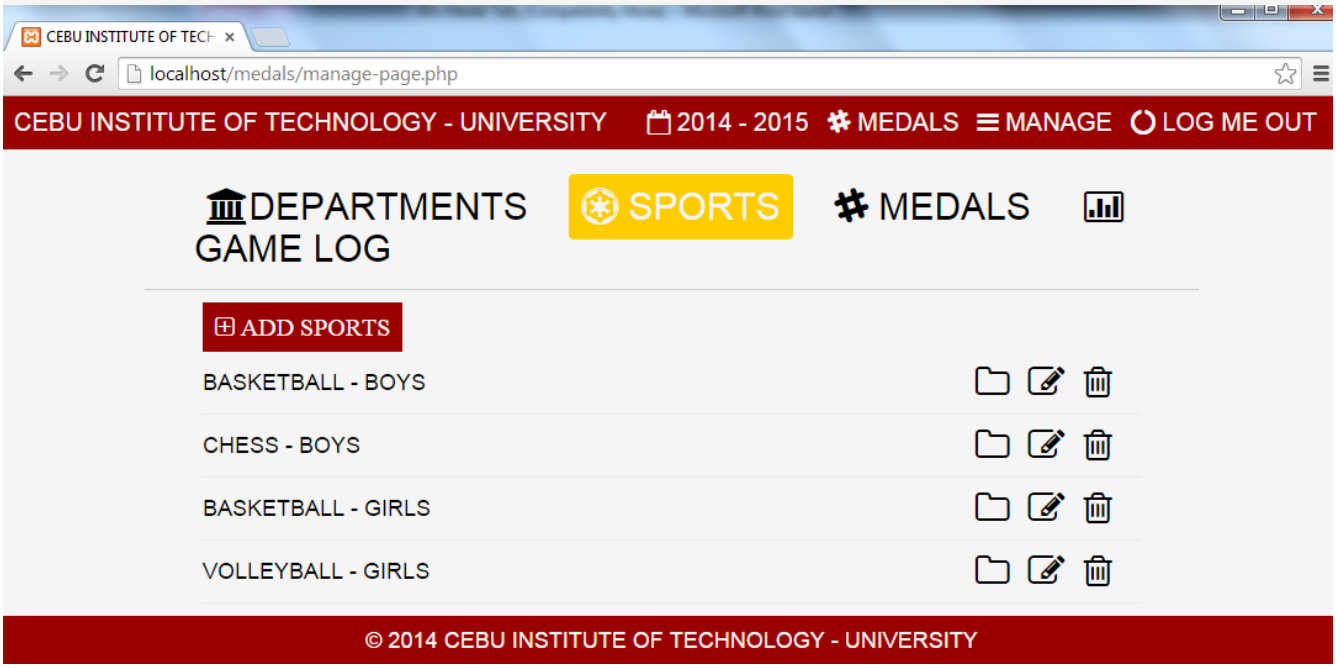
11.5.1. Process 1 Description

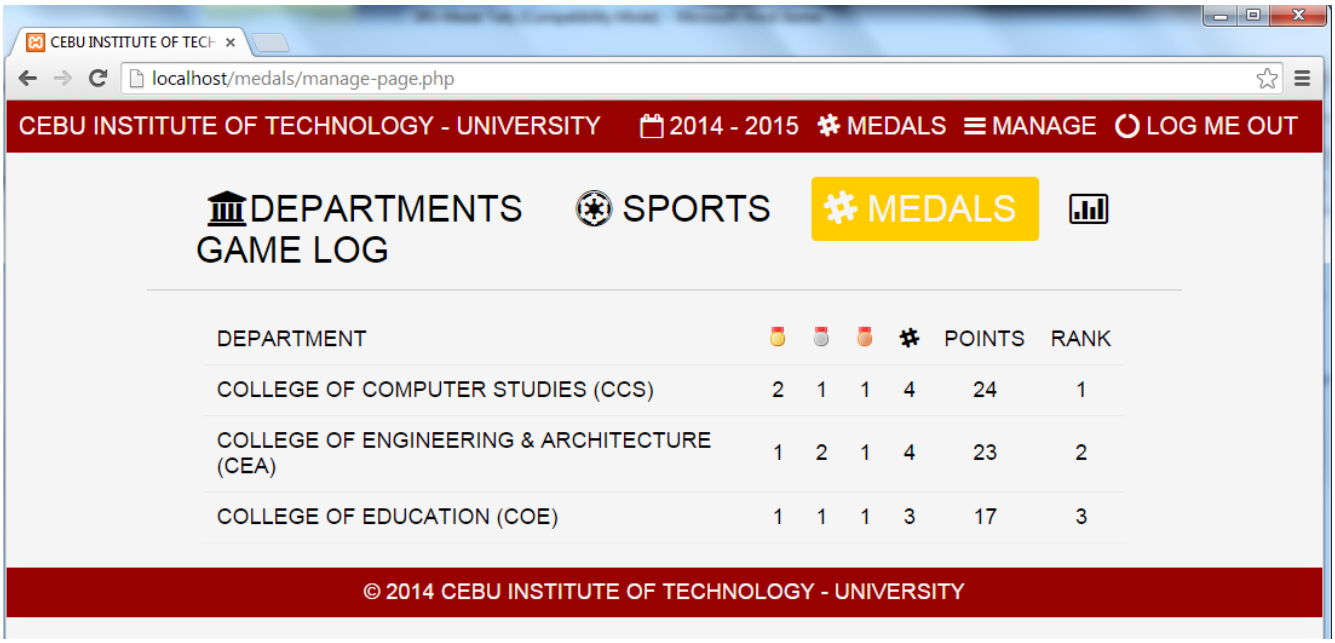
Identification	Admin
Type	Form
Description	The Admin must login in order to use the system.
Function/s	loginRequest ()

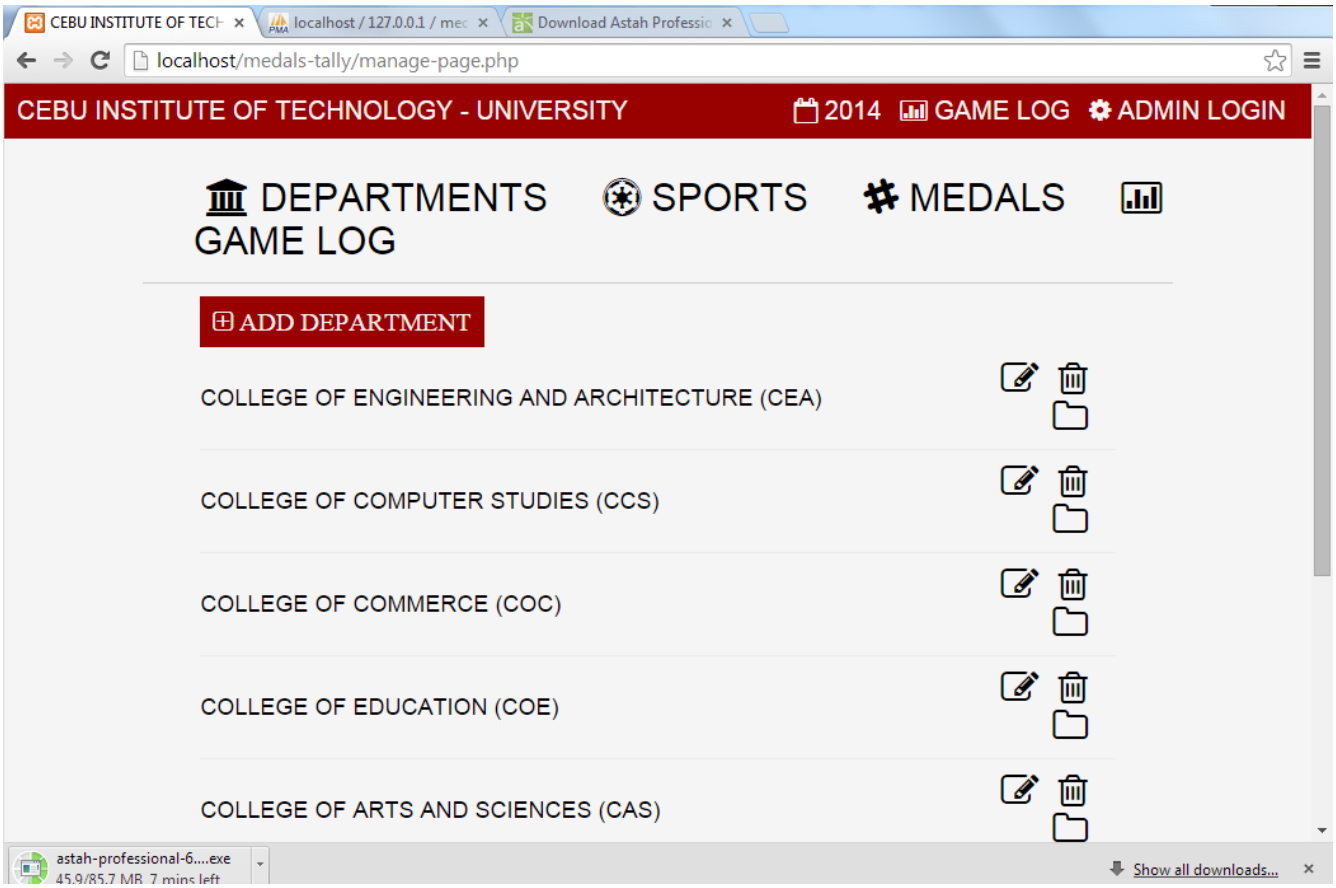


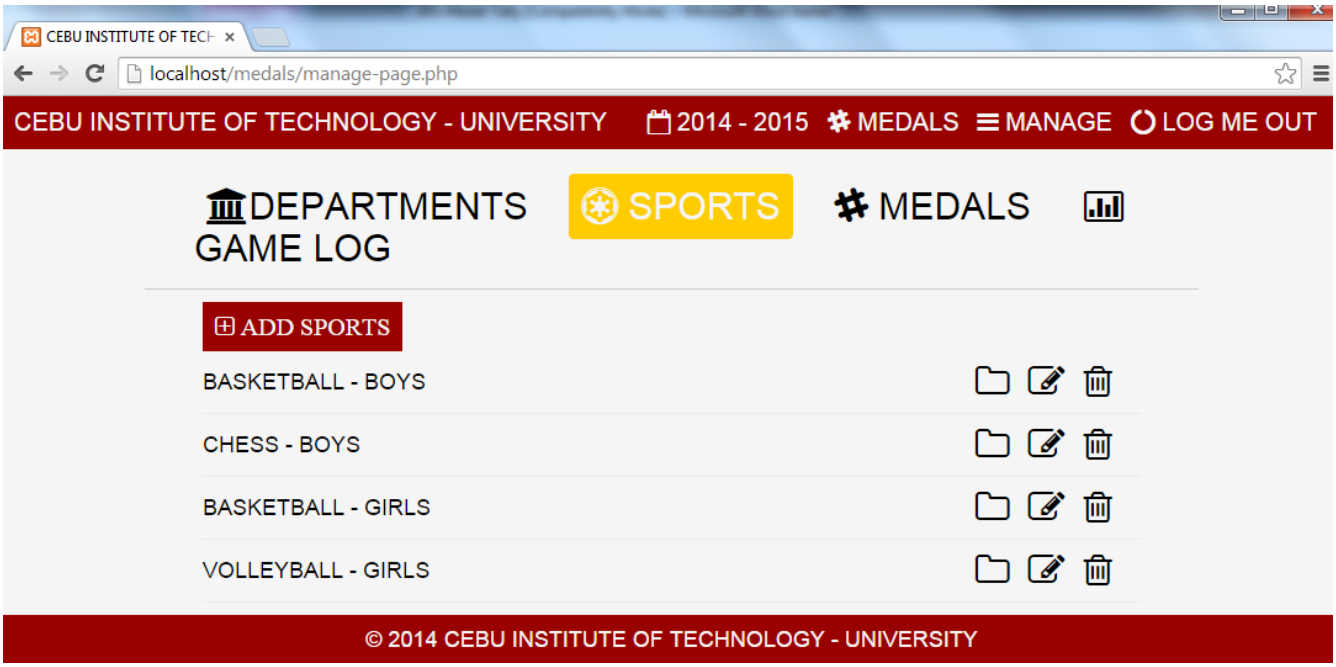
11.5.2. Process 2 Description

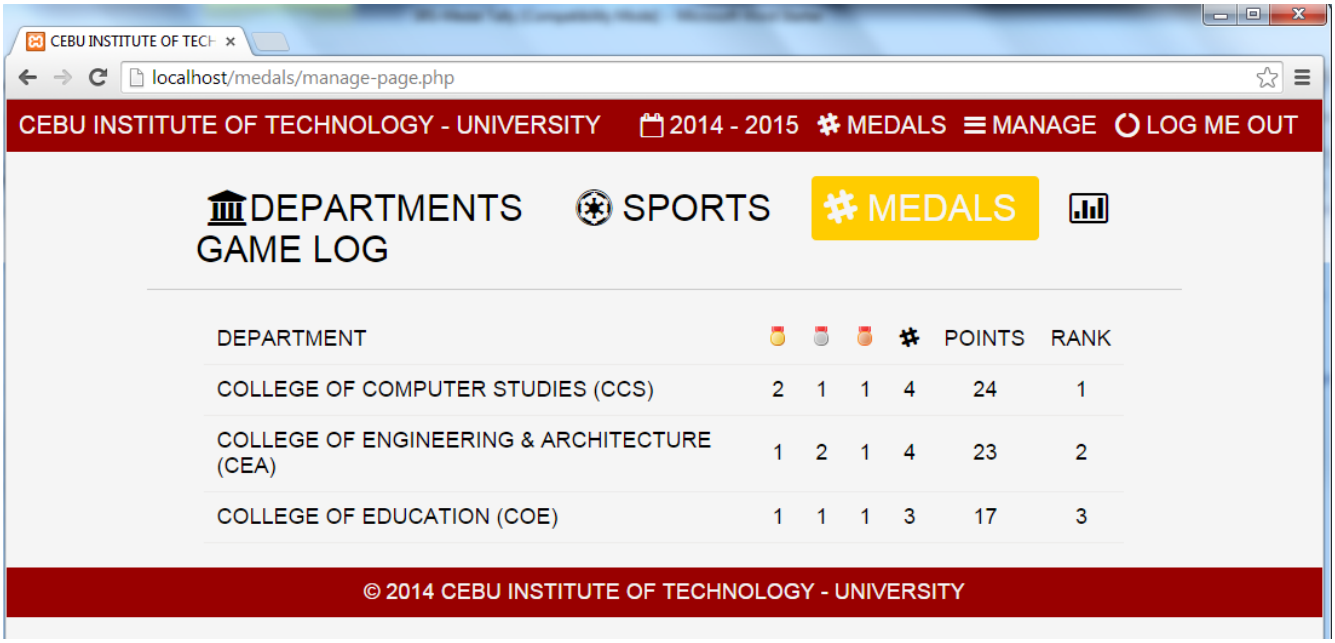
Identification	Admin
Type	Form
Description	This will display all the functions that the Admin can use.
Function/s	sendRequest ()

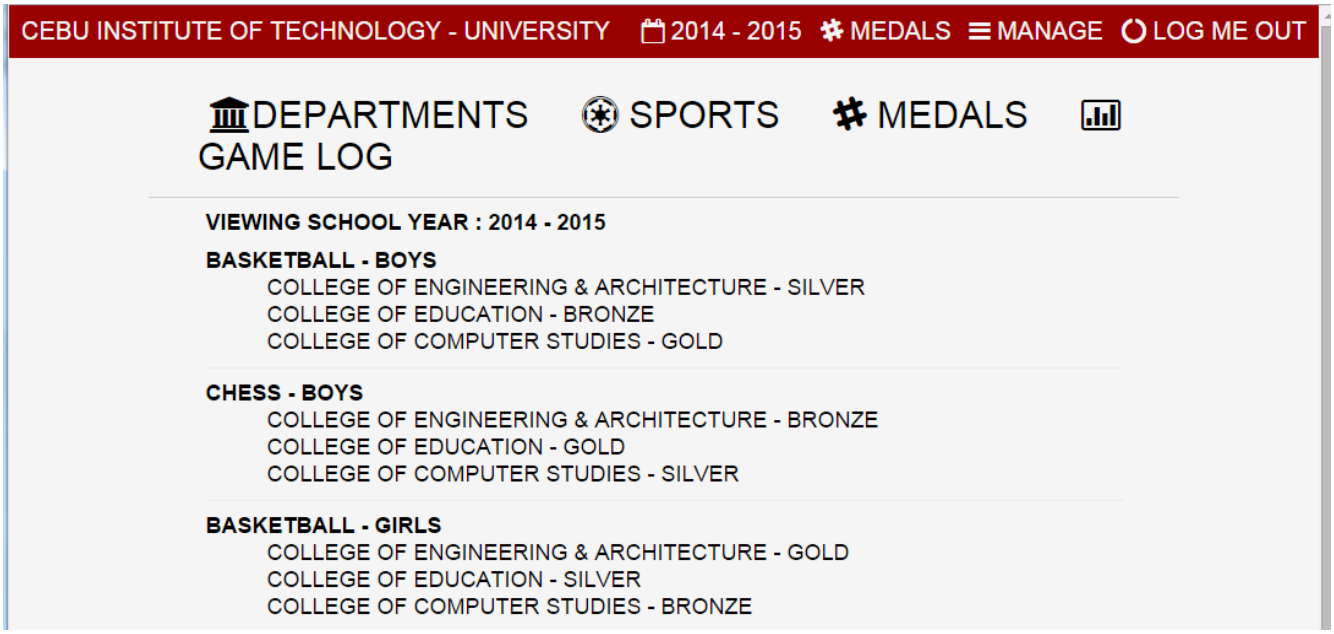


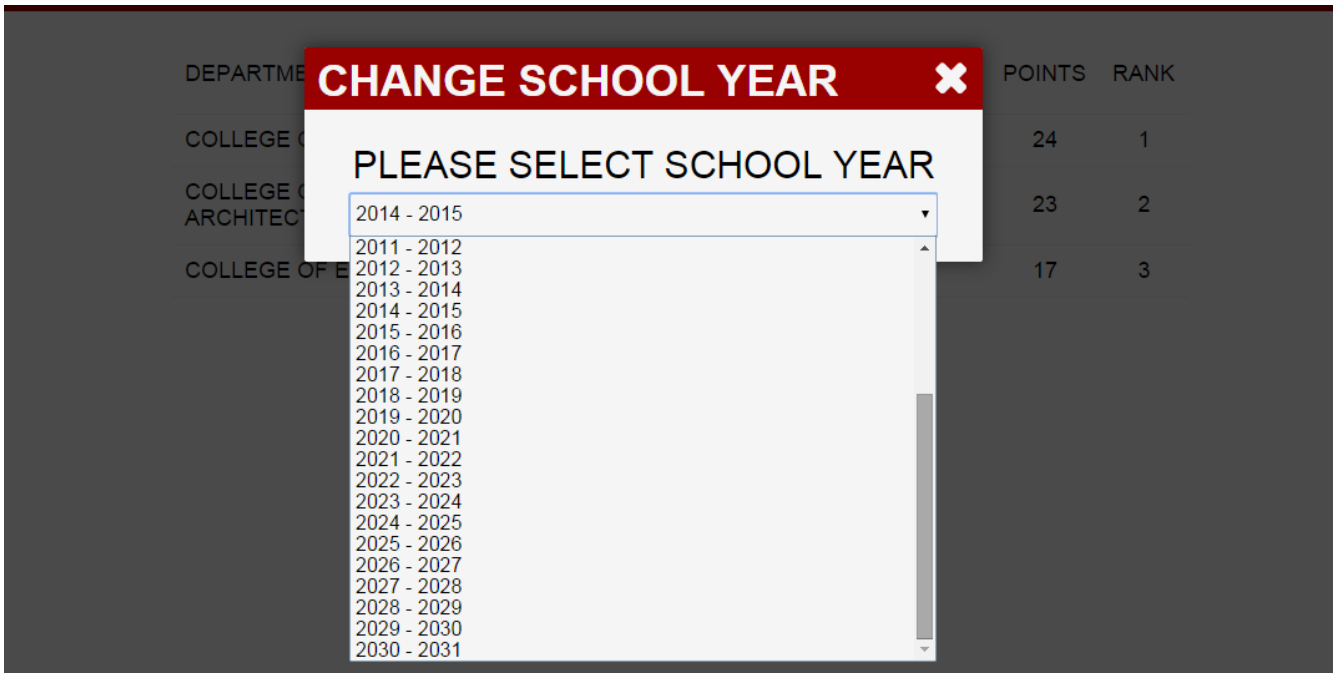

CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
DEPARTMENT
POINTS
RANK
COLLEGE OF COMPUTER STUDIES (CCS) 2 1 1 4 24 1
COLLEGE OF ENGINEERING & ARCHITECTURE (CEA) 1 2 1 4 23 2
COLLEGE OF EDUCATION (COE) 1 1 1 3 17 3
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY


CEBU INSTITUTE OF TECH
localhost / 127.0.0.1 / mec
Download Astah Professio
localhost/medals-tally/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014
GAME LOG
ADMIN LOGIN
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
ADD DEPARTMENT
COLLEGE OF ENGINEERING AND ARCHITECTURE (CEA)
COLLEGE OF COMPUTER STUDIES (CCS)
COLLEGE OF COMMERCE (COC)
COLLEGE OF EDUCATION (COE)
COLLEGE OF ARTS AND SCIENCES (CAS)
astah-professional-6....exe
45.9/85.7 MB, 7 mins left
Show all downloads...


CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
ADD SPORTS
BASKETBALL - BOYS
CHESS - BOYS
BASKETBALL - GIRLS
VOLLEYBALL - GIRLS
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY


CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
DEPARTMENT
POINTS
RANK
COLLEGE OF COMPUTER STUDIES (CCS) 2 1 1 4 24 1
COLLEGE OF ENGINEERING & ARCHITECTURE (CEA) 1 2 1 4 23 2
COLLEGE OF EDUCATION (COE) 1 1 1 3 17 3
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY





11.5.3. Process 2 Description

12. Detailed Design

12.1. Module Detailed Design

12.1.1. Module 1 Detail

Flow of events:

1. User selects.

2. Clicking the add medal button, successfully the medal has been added to the user's account

2. Clicking view medals allows the user view the total medals he/she have.

Operations:

Name: medal

Pre-condition: connected to users page

Post-condition: saving the data

Name: **Operation**

Type: Operation Page

Description: Customer can view the recent and old Operation that has been made.

Attributes: texts, buttons

Operations:

Name: Operation

Pre-condition: connected to database Operation table

Post-condition: successfully update

Flow of events:

1. Clicking update button update the user account, this update the medals he/she had.

12.1.2. Module 2 Detail

Name: **Log In**

Type: Log in Form

Description: Customer log in to do Operation.

Attributes: radio button, button

Operations:

Name: login

Pre-condition: connected to page

Post-condition: Operation process

Flow of events:

1. User input for username
2. User input for password
3. Clicking login button connect to page.

12.2. Data Detailed Design

12.2.1. Data Entity 1 Detail

Table 6.2.1.1

Medals

Field	Type
User_id	int[10]
User_name	Varchar[20]
Medal_type	Varchar[20]

Table 6.2.1.2

Medal counts

Field	Type
MedalOperation_id	int[10]
Medal_id	int[10]

subtotal	Decimal[13,2]

Table 6.2.1.3

Operations

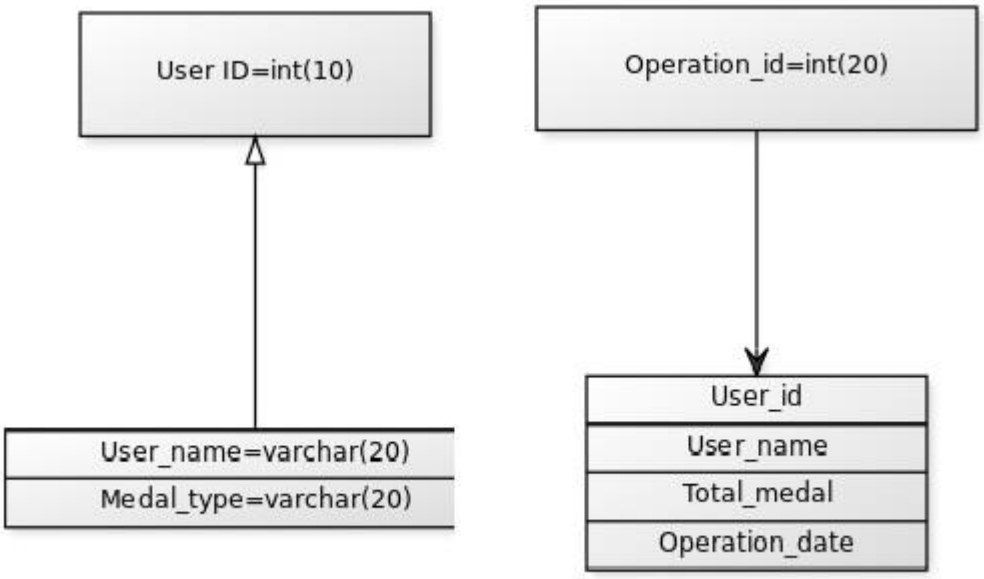
Field	Type
Operation_id	int(20)
User_id	int[20]
User_name	varchar[20]
Total_medal	decimal[20]
Operation_date	date

13. Appendixes

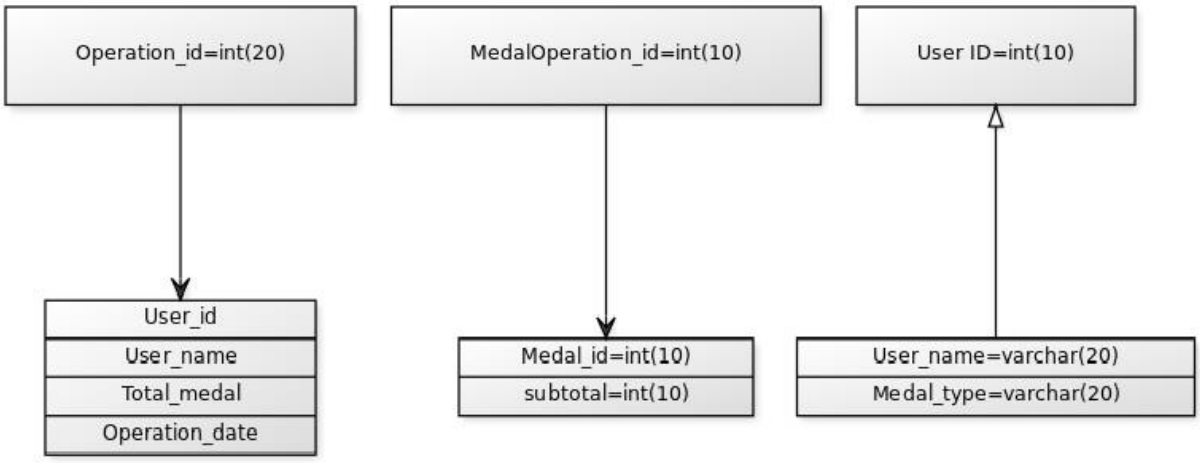
14. Index

15. Annexes

9.1. Data flow diagram (optional)



9.2. Class diagram



9.3. Use case realization (Sequence diagram / Communication diagram)

Figure 9.3.2 *Operation*

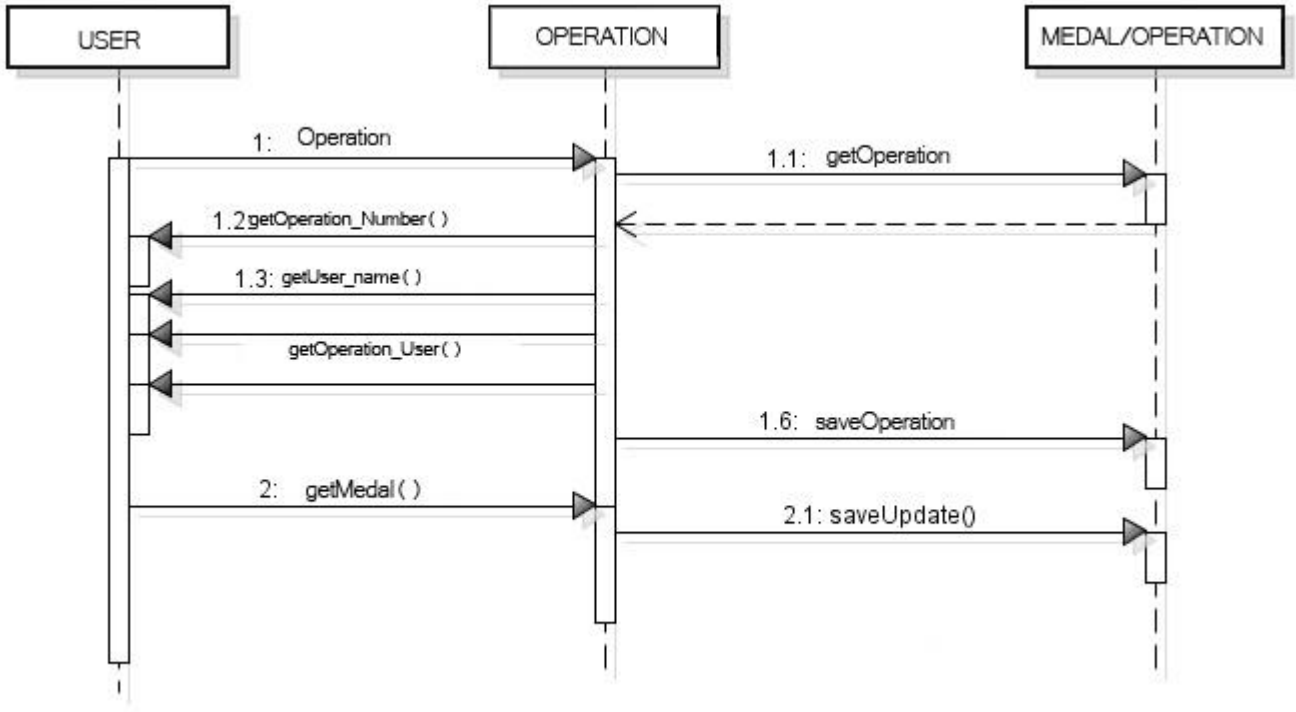
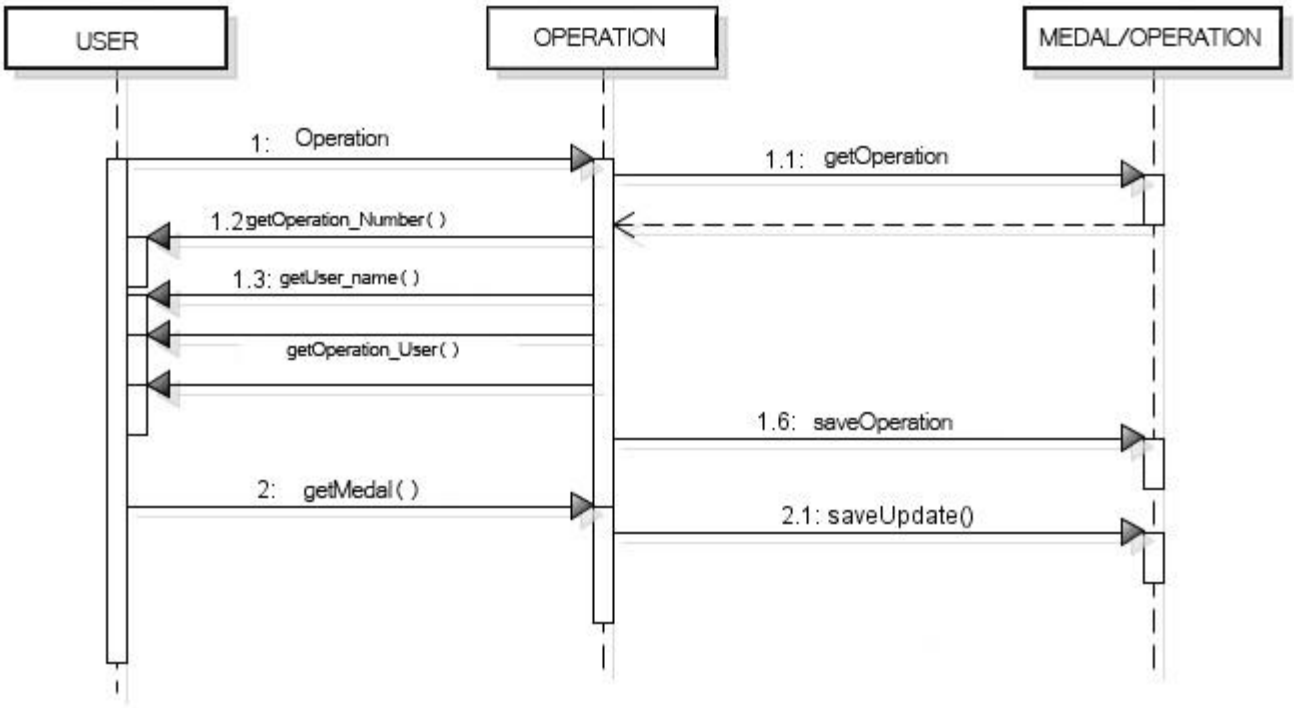
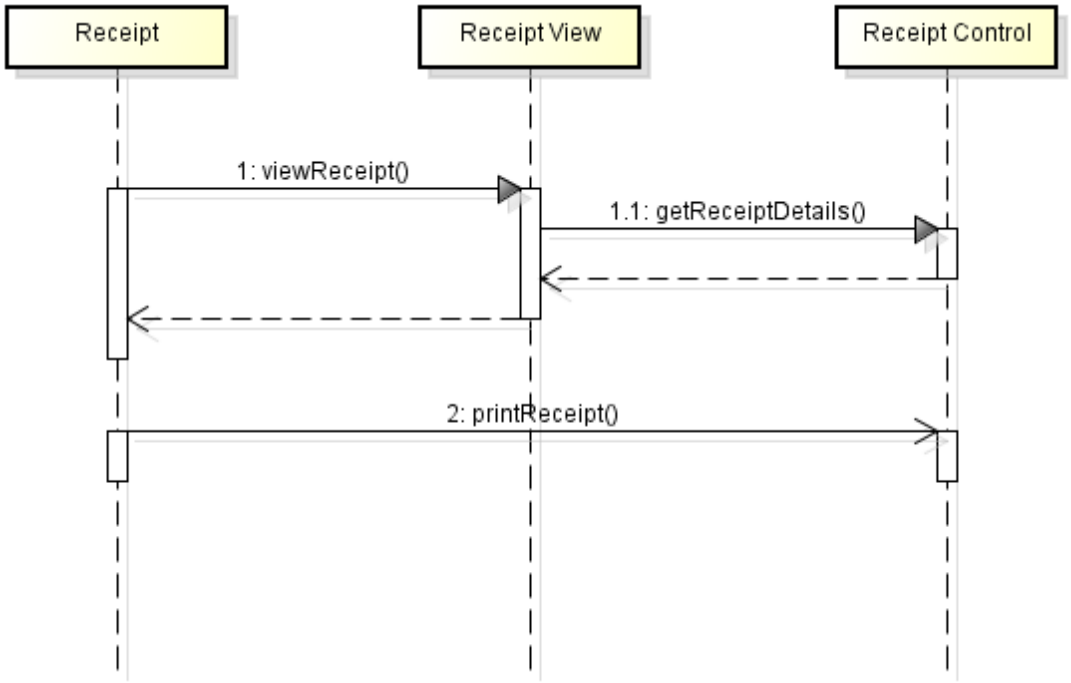


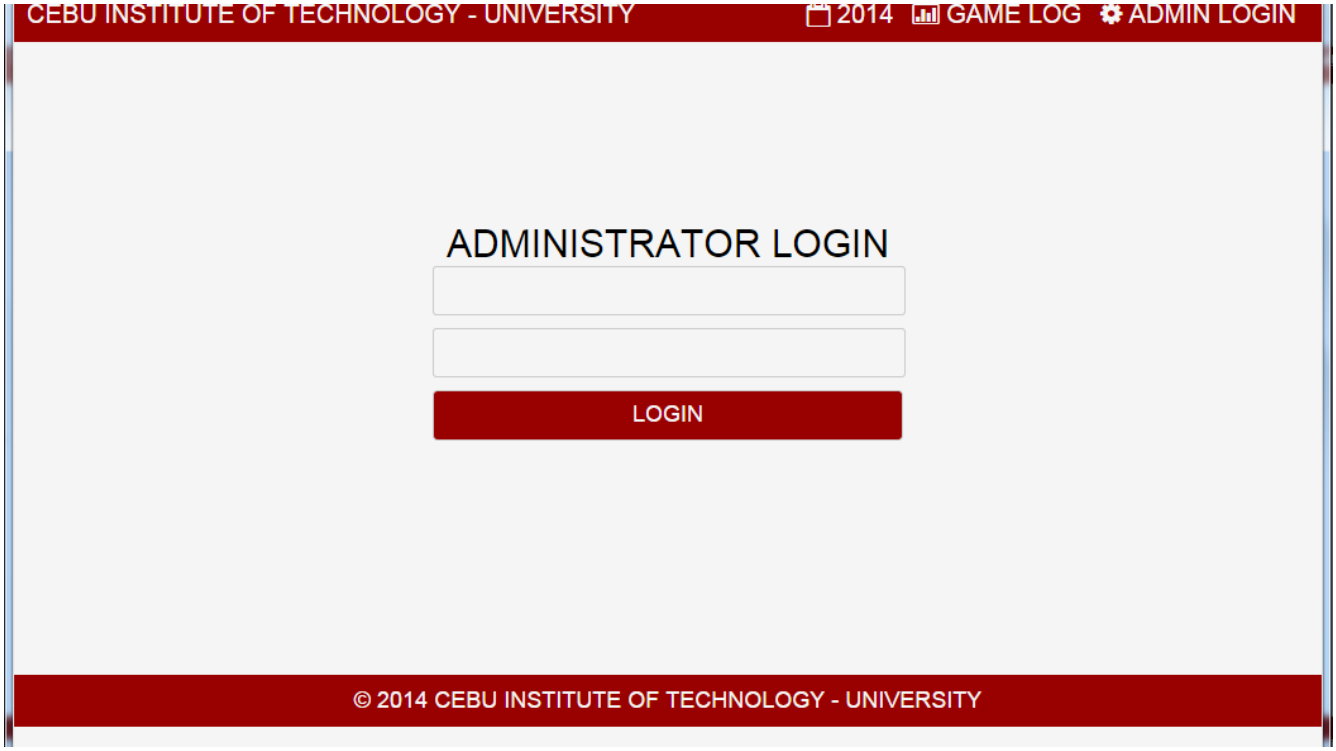
Figure 9.3.3 Medal

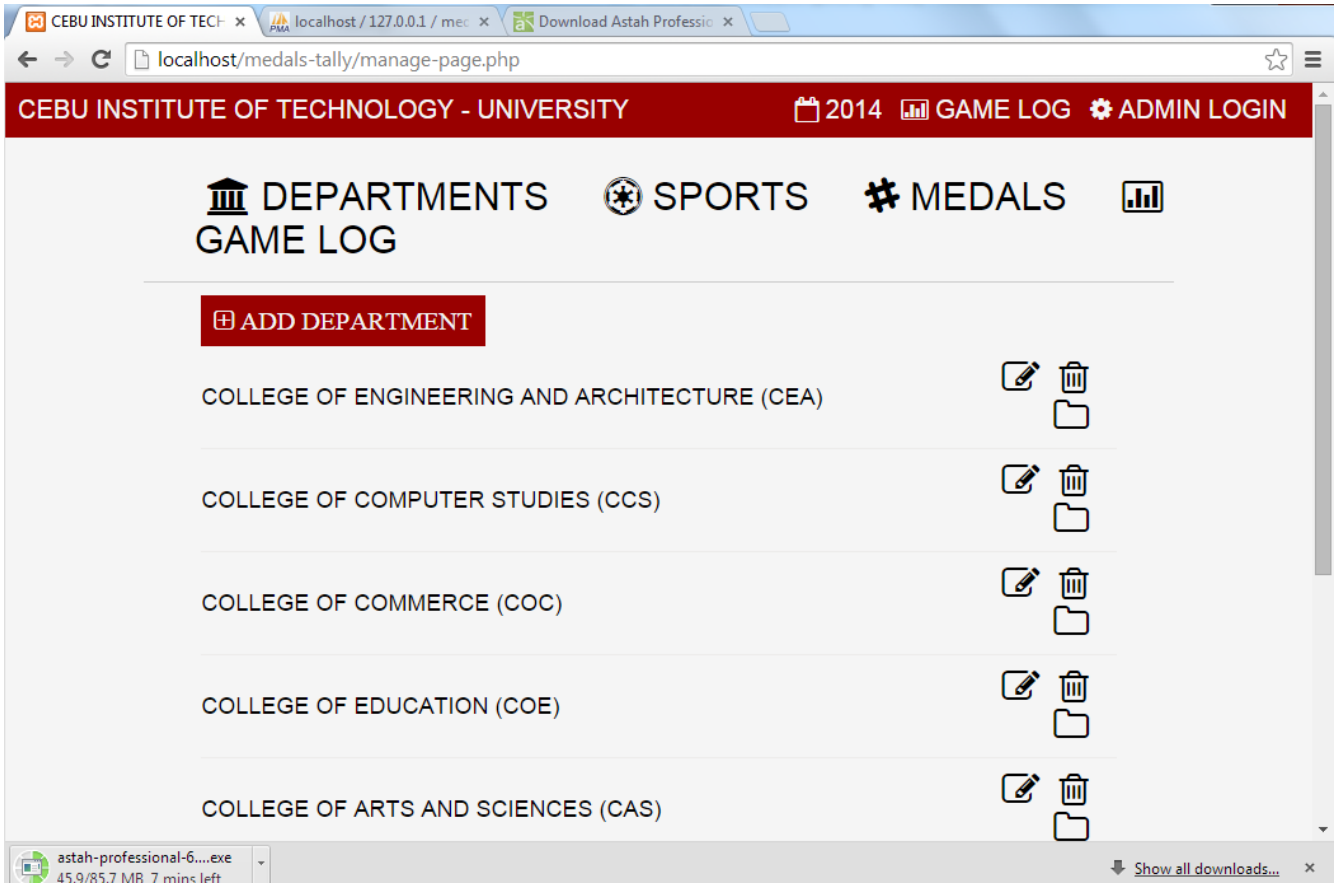

USER
OPERATION
MEDAL/OPERATION
1: Operation
1.1: getOperation
1.2 getOperation_Number( )
1.3: getUser_name( )
getOperation_User( )
1.6: saveOperation
2: getMedal( )
2.1: saveUpdate()

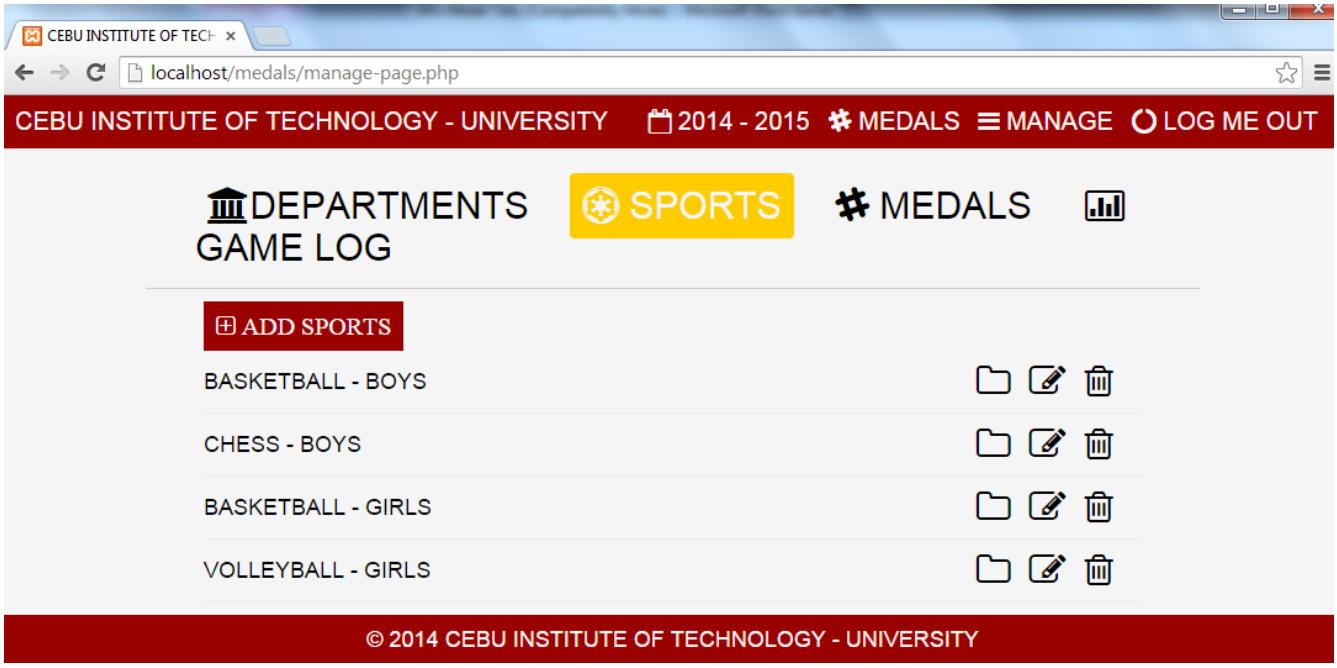

Receipt
Receipt View
Receipt Control
1: viewReceipt()
1.1: getReceiptDetails()
2: printReceipt()

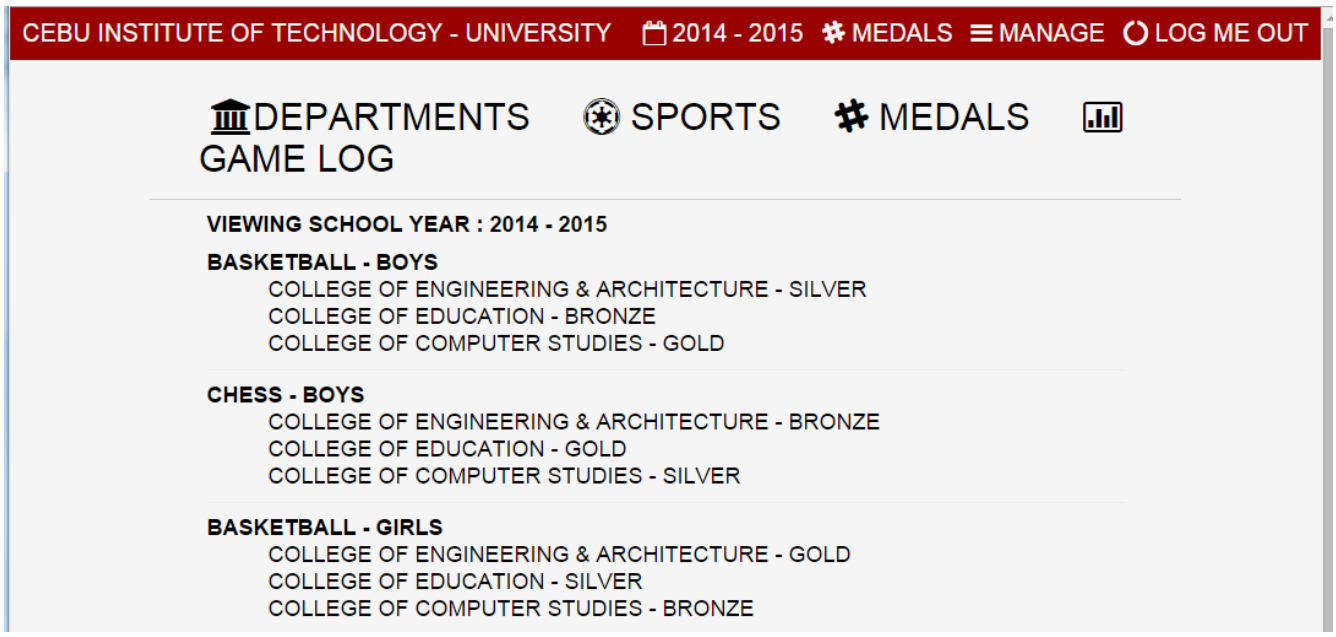
9.4. User interface design

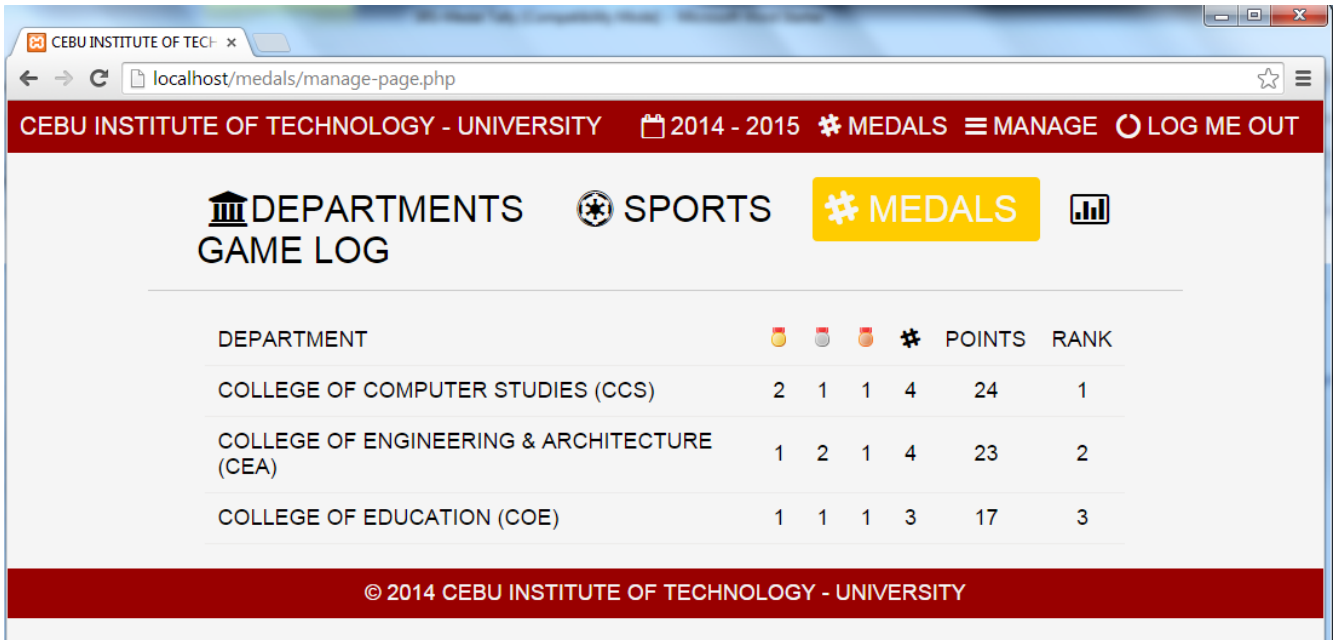
Figure 9.4.1 ***Dashboard***

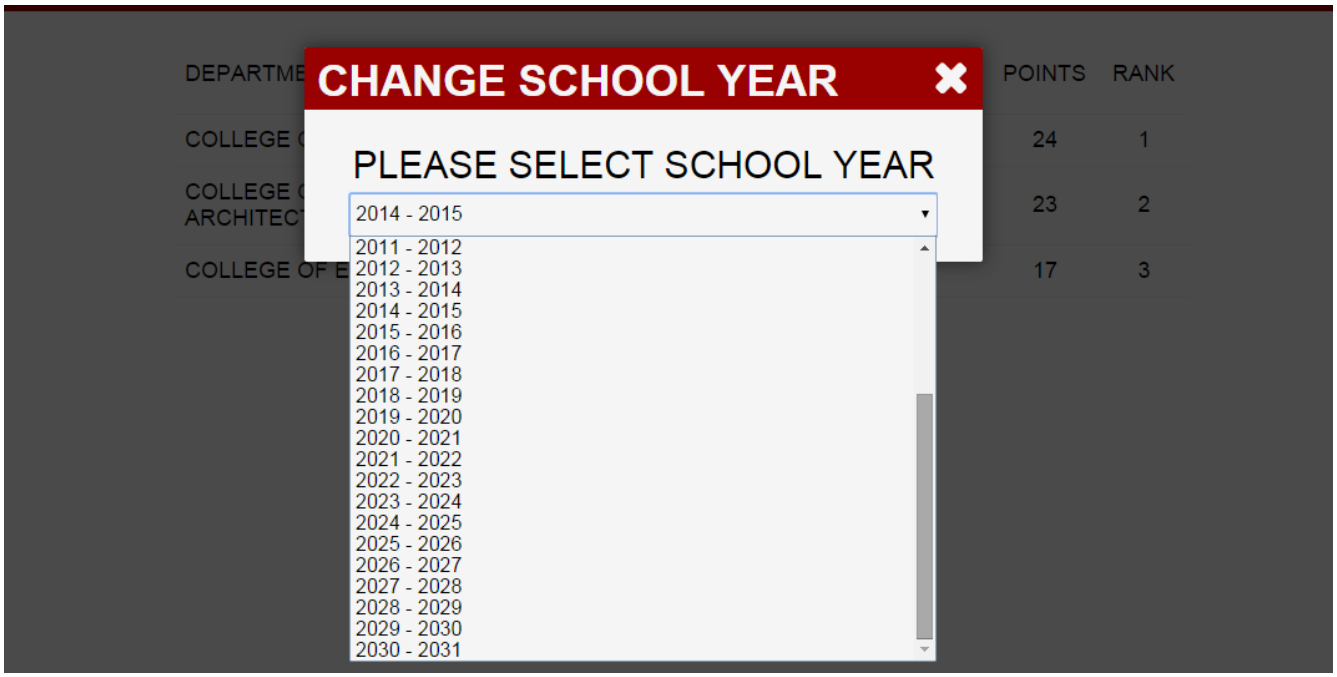



CEBU INSTITUTE OF TECH
localhost / 127.0.0.1 / mec
Download Astah Professio
localhost/medals-tally/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014
GAME LOG
ADMIN LOGIN
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
ADD DEPARTMENT
COLLEGE OF ENGINEERING AND ARCHITECTURE (CEA)
COLLEGE OF COMPUTER STUDIES (CCS)
COLLEGE OF COMMERCE (COC)
COLLEGE OF EDUCATION (COE)
COLLEGE OF ARTS AND SCIENCES (CAS)
astah-professional-6....exe
45.9/85.7 MB, 7 mins left
Show all downloads...


CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
ADD SPORTS
BASKETBALL - BOYS
CHESS - BOYS
BASKETBALL - GIRLS
VOLLEYBALL - GIRLS
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY


CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
VIEWING SCHOOL YEAR : 2014 - 2015
BASKETBALL - BOYS
COLLEGE OF ENGINEERING & ARCHITECTURE - SILVER
COLLEGE OF EDUCATION - BRONZE
COLLEGE OF COMPUTER STUDIES - GOLD
CHESS - BOYS
COLLEGE OF ENGINEERING & ARCHITECTURE - BRONZE
COLLEGE OF EDUCATION - GOLD
COLLEGE OF COMPUTER STUDIES - SILVER
BASKETBALL - GIRLS
COLLEGE OF ENGINEERING & ARCHITECTURE - GOLD
COLLEGE OF EDUCATION - SILVER
COLLEGE OF COMPUTER STUDIES - BRONZE


CEBU INSTITUTE OF TECH
localhost/medals/manage-page.php
CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY
2014 - 2015
MEDALS
MANAGE
LOG ME OUT
DEPARTMENTS
SPORTS
MEDALS
GAME LOG
DEPARTMENT
POINTS
RANK
COLLEGE OF COMPUTER STUDIES (CCS) 2 1 1 4 24 1
COLLEGE OF ENGINEERING & ARCHITECTURE (CEA) 1 2 1 4 23 2
COLLEGE OF EDUCATION (COE) 1 1 1 3 17 3
© 2014 CEBU INSTITUTE OF TECHNOLOGY - UNIVERSITY


DEPARTME
POINTS
RANK
COLLEGE
24
1
COLLEGE
ARCHITEC
23
2
COLLEGE OF E
17
3
CHANGE SCHOOL YEAR
PLEASE SELECT SCHOOL YEAR
2014 - 2015
2011 - 2012
2012 - 2013
2013 - 2014
2014 - 2015
2015 - 2016
2016 - 2017
2017 - 2018
2018 - 2019
2019 - 2020
2020 - 2021
2021 - 2022
2022 - 2023
2023 - 2024
2024 - 2025
2025 - 2026
2026 - 2027
2027 - 2028
2028 - 2029
2029 - 2030
2030 - 2031

Software Test Document

For

CIT-U Medal Tally

Signature

Members	Roles	Signature
John Vincent Capoy	Documentation	
Niel Anthony Canton	Web Developer	
Noe Norman Sapio Galo	Web Developer	
Alwynne Batucan	Testing	

Change History

VERSION	DATE	AUTHOR	CHANGES
1.0	August 20, 2014	John Vincent Capoy Alwynn Batucan Neil Anthony Canton Noe Norman Galo	Initial Version
2.0	August 27, 2014	John Vincent Capoy Alwynn Batucan Neil Anthony Canton Noe Norman Galo	Final Version

Preface

Where in this documentation comprises the different testing method in implementing the functionalities and operations of the systems program.

This is based upon the concepts espoused in the document IEEE Std. 829-1998 IEEE Standard for Software Test Documentation, IEEE Std. 1008-1997 IEEE Standard for Software Unit Testing and IEEE Std. 1012-1998 IEEE Standard for Software Verification and Validation.

The context of the Software Test Document is the process of every testing method need to be done of what was stated in the Software Specification Requirements. It concerns the flow of the system which is generated the codes. It lists approaches and standards to ensure that a quality product that meets the needs of the user is produced. For the maintenance phase, this document provides the context for regression testing when any changes are made.

Table of Contents

Contents

List of Tables

1.0. Scope

This summarizes all the different testing processes and action procedures in delivering the different executions of the system.

This portion will tell us about the execution of the program with dynamic aspects of software testing. This show us the purpose, outline, the content of what's inside of the project and some documentation. It focuses on the dynamic part of the system by doing some testing in every module.

2.0. References

IEEE Std 830-1998 (Revision of IEEE Std 830-1993) **IEEE Recommended Practice for Software Requirements Specifications**

IEEE Std 1016-1998 (Revision of IEEE Std 1016-1987) **IEEE Recommended Practice for Software Design Descriptions**

IEEE Std 1058-1998 (Revision and redesignation of IEEE Std 1058.1-1987, incorporating IEEE Std 1058-1998 and IEEE Std 1058a-1998) **IEEE Standard for Software Project Management Plans**

3.0. Definitions

[Table 3.1] ***Definitions and Acronyms***

Term	Definition
CIT-U	Cebu Institute of Technology – University
	An object that embodies method that perfectly complements the lecture-type approach. It completes the whole picture the students need.
CCS	College of Computer Studies
	An organization assisting students needs.
IEEE	Institute of Electrical and Electronics Engineers
	a professional association headquartered in New York City that is dedicated to advancing technological innovation and excellence.
CAMS	Customer Account Management System
	A system controls customer information.
SRS	Software Requirements Specifications
	A module wherein all requirements are being specified for project planning.
SPMP	Software Project Management Plan
	A module wherein all charts, project scheduling are being plan.
SDD	Software Design Description
	A module wherein all design interfaces are being showed.
STD	Software Test Document
	A module wherein all the functionalities of the designs are implemented.

TDS	Test Design Specification
	Detailing test conditions and the expected results as well as test pass criteria
TCS	Test Case Specification
	Specifying the test data for use in running the test conditions identified in the Test Design Specification
TPS	Test Procedural Specification
	Detailing how to run each test, including any set-up preconditions and the steps that need to be followed.
TITR	Test Item Transmittal Report
	Reporting on when tested software components have progressed from one stage of testing to the next
TL	Test Log
	Recording which tests cases were run, who ran them, in what order, and whether each test passed or failed

4.0. Test Plans

4.1. Purpose

This Test Plan will prescribe the scope, approach, resources, and schedule of the testing activities. In addition, to identify the items being tested, the features to be tested, the testing tasks to be performed, the personnel responsible for each task, and the risks associated with this test plan. The primary focus of this plan is to ensure that the functionalities provide the expected output.

The purpose of this document is to describe the standards and procedures to follow during the software testing phases of the CIT-U Medal Tally. This document supports the section on Testing and Validation in the Integration and Methods Quality Manual.

The test plan is to provide the necessary information's needed for the verification and validation of the features and functionalities of the system.

4.2. Outline

4.2.1. Introduction

This test plan covers a full systems test of the Customer Account Management System of the *CIT-U Medal Tally*. This includes operator and user procedures. In addition to comprehensively testing multi program functionality, external interfaces, security, recovery, and performance will also be evaluated.

This test plan is a planning document that shows the following:

- How the testing will be done
- Who will do it/persons responsible for the tasks
- What functions will be tested
- How long it will take

- ***4.2.2. Test Functions***

All functions that make up the *CIT-U Medal Tally* features of will be tested during the testing. The administrator will also control changes to the versions under test and notify the test group when new versions are available. The following documents will provide the basis for defining correct operation:

- Software Requirements Specifications (SRS v1)
- Software Project Management Plan (SPMP v1)
- Software Design Description (SDD v1)

[Table 4.2.2.1] Functions to be tested

Log In	This functionality is where user input username and password
Register	This functionality is where the customer register to create an account for his/her transaction
Transactions	This functionality retrieves all the report of product and customer information entries.
Update Account	This functionality update customer transaction

- ***4.2.3. Features to be tested***

The features to be tested are the functionalities of the system.

- ***4.2.4. Features not to be tested***

The following features will not be included in the system tests like the initiated buttons and fields that only used for when acted upon a user.

- ***4.2.5. Approach***

The testing for this project is the all functionalities of the system that will undergo testing. Each function shall be tested directly from the execution. Whoever may the tester is, he/she will analyze the result and to check whether it functions are correctly implemented or not. Thereon every tested function is to be tested individually and will be verified for any errors.

This testing approach must be attended by the project teams especially if possible the team leader would be keen to the undergoing testing procedures. In order to verified the occurrences of defects and errors of the testing.

In order to ensure privacy, all test data extracted from the company will have privacy-sensitive fields changed

Interface testing – The system will have a user – friendly GUI for the user to navigate the system easily. In order to test the interface, user will choose or either input data if there are required fields to have an input. For further testing, it must be complied all the fields so that it can proceed to the next requirement to be test.

Security testing – Attempted access without a proper password to the system will be tested. Passwords must be encrypted automatically during the registration of user. The tests will verify that unauthorized user access to confidential data is prevented.

The project will be running on a server which supports MySQL application associated with the PHP framework.

- ***4.2.6. Items Pass/Fail Criteria***

The system must satisfy the standard requirements for system pass/fail. This system will be implemented through applications like CodeIgniter PHP Framework. MySQL will be used as a database back -end.

- ***4.2.7. Suspension Criteria and resumption requirements***

If any defects are found which seriously impact the test progress, the Quality Assurance officer may choose to suspend testing. Criteria that will justify test suspension are:

- Source code contains one or more critical defects, which seriously prevents or limits testing progress.
- Assigned test resources are not available when needed by the test team.

If testing is suspended, resumption will only occur when the suspension of the problem has been resolved.

- ***4.2.8. Test Deliverables***

This document will be generated by the test group and delivered to the client after completion.

- ***4.2.9. Testing Tasks***

- Analysis on encountered errors
- Debugging
- Re – Test

- Operation Testing
- Error Recording

- ***4.2.10. Environmental Needs***

This system will be implemented to a PHP Framework CodeIgniter and MySQL as its database.

For hardware, any computer will do if it is supported by the required specification. Operating System: WINDOWS XP/ WINDOWS 7.

- ***4.2.11 .Responsibilities***

System test group:

The management that manage the testing procedures of the system.

Development project group:

The group that initiate the different type of testing in concurrent of determining what are to be tested.

[Table 4.2.11.1] ***Roles and Responsibilities***

Role	**Responsibilities**
John Vincent Capoy - Team Leader	-Software project planning and monitoring -Milestone and schedule planning and monitoring -Set and communicate the team meeting agendas. -Risk Management.

Neil Anthony Canton **Noe Norman Galo** - Process Manager	-Process definition -Set and implement development standards. -Conduct process reviews to ensure compliance. -Leads the team in defining the processes, practices, and procedures -Ensures that the team's recording all required data Manages the team process
Alwynn Batucan - Risk Manager	-Champion mitigation implementation. -Review status, severity, ownership, and completeness of risks. -Determine risks to be returned to the appropriate project teams. -Establish severity of risks and define target dates. -Establish ownership of risk and confirm target dates. -Identify risks for escalation to the Steering Committee. -Work with project teams, subject matter experts, and the Risk Manager to facilitate solutions to risks.

- ***4.2.12. Staffing and Training Needs***

All of the members already have the knowledge and skill to conduct testing. Training is not necessary no more.

- ***4.2.13. Schedule***

Software testing will be conducted in the final week and the revision changes must be

updated.

- ***4.2.14. Risk and Contingencies***

It is possible that a system might encounter failure during testing period. Possibly the project team might require to extend the software testing to their time frame. Changes of schedules and compression of time must be applied.

- ***4.2.15. Approvals***

Team	Signature
15.1.1.1.1.1.1.1. John Vincent Capoy	
Alwynn Batucan	
Noe Norman Galo	
Neil Anthony Canton	

5.0. Test Design Specification

• 5.1. Purpose

The purpose of Test Design Specification is to determine the detailed condition and the expected result if it passes the criteria of the specific requirements.

• 5.2. Outline

• *5.2.1. Features to be Tested*

The features which are testable and the design of tests for those features are completely dependent on the access that the testers have to enter and observe information in the systems under test.

The following features of Customer Account Management System will be tested.

- Log In
- Feature
- Category option
- Update account

• *5.2.2. Approach Refinements*

The test personnel will use the IEEE Standard as a guideline in making the test design and test case specifications. In order to organize the testing phase of the Customer Account Management System of (Enter System Name).

Interface Testing

During interface testing, under the order products page the only things to do is to select and views. No necessary inputs are demonstrated.

Security Testing

The only functions needs to be secured are the customer account transactions. This is to identify the current and previous transactions made.

Constraints

The overall testing of the system is scheduled on the final week. Any constraints may occur but however, project teams will give all the effort to fully meet the target output of this system.

- ***5.2.3. Test Identification***

The system must satisfy the set requirements of the clients. The following will be the basis for the item testing:

- User input
- Resulting action based on its purpose
- Generated output

- ***5.2.4. Features Pass/Fail Criteria***

The system must satisfy all the necessary requirements set by the client. Test items will be passed if the following requirements are met:

- Interface – if the interface are met in regards to the specification requirements (passed)

  If the client didn't like the design (failed)

- Functionality - If the data returns the right information (passed).

  If the wrong data is returned (failed).

- Data - If the data being stored in the access is correct (passed).

  If the data being stored in the access is wrong (failed).

6.0. Test Case Specification

• 6.1. Purpose

The purpose of this document is to specify the test data use in for running the test conditions that is identified in the Test Design Specification. It includes the reference that corresponds to the design that describes the dependency of its executions.

• 6.2. Outline

• *6.2.1. Test*

The items to be tested in Account Management System:

- Log In
- Register
- Transactions

• *6.2.2 Input Specifications*

Customer Account Management System select a specific user and do operations once or repeatedly. These are the variables with its type.

A. Log In
   - Id int(20)
   - username varchar(20)
   - password varchar(20)

B. Transactions
   - transaction_id int(20)

- total_payment int(20)
- transaction_date datetime(20)

- ***6.2.3. Output Specifications***

Specify the output required and provides the exact results after executing the inputs.

A. Login
    1. User inputs username and a password.
    2. Click Log In button
    3. Database will check if username and password exist.
    4. If username and password exist, user will be navigated to the order product page.
    5. If username and password did not exist the user will register again.
B. Transactions
    1. The transaction form will automatically display after the View transaction was click, this shows the transaction details of the customer.
    2. Clicking update account button, will display payment form and update the balance of the customer.

- ***6.2.4. Environmental Needs***

This system will be implemented through application like PHP Framework CodeIgniter, MySQL will be used as database back-end.

For hardware, any computer will do as long as supported by the required requirements.

- ***6.2.5. Specific Procedure Requirements***

User must be a computer literate and have knowledge in exploring the different aspects of the computer and internet.

7.0. Test Procedure Specification

• 7.1. Purpose

The purpose of this document is to describe how the user will run the test including all the pre-conditions and steps that need to be followed. This is developed according to the stated process and definitions in Test Design and Test Case Specification.

• 7.2. Outline

• *7.2.1. Purpose*

All procedures describe the actions to be perform in testing the design and cases executions. Which is specifies in the Test Design and Test Case Specification.

• *7.2.2. Special Requirements*

Project Teams and the tester must have knowledge ideas on how to deal the testing process.

• *7.2.3. Procedural Steps*

- The members must be prepared for the test and ready to anticipate what could be the possible errors of the testing.
- Gather all requirements and data that are need for the test.
- Document the test results in any way suitable for recording.

A. Log

Test results will be immediately recorded.

B. Set Up

Hardware and software performance will be added to ensure that such external factors will not influence the test results.

C. Start

Start the testing with the external functionalities and then its internal functionalities.

D. Proceed

During the execution of test procedure, a priority must be met which is to finish the interrupted testing.

E. Measure

Precautionary measures are applied during the execution of the procedure such as time table management to have enough time and to be well-organized.

F. Shutdown

Insufficiency of information or suspension of testing due to unnecessary events may take to consideration but still take some extra effort just to cover time being.

G. Restart

Any revisions made to the interface or some of the codes, considering the time can be covered by debugging codes and making retouches from the user interface, must construct ideas out of the members to help generate a new output.

H. Stop

To put up an end, conducting short review to every little details of the whole testing procedure executed and has a checking of the entire test made.

8.0. Test transmittal report

• 8.1. Purpose

The purpose of this report is reporting the progress of each item that are already tested and transmitted the status of the test to proceed on the next stage of testing. This also includes the responsible tester of each item.

• 8.2. Outline

• *8.2.1. Status*

In testing phase, possible changes may occur during the test of the system if the testing encounters flaws and negative situations the team must be attentive and alert in making a turn back to the previous phase to trace and check the cause of the problem.

9.0. Test Log

• 9.1. Purpose

The purpose of the test log report is to record the running and error functionalities of the system. To record the details of the functions to be run, when it is tested, where it is tested and who made the test execution.

• 9.2. Outline

• *9.2.1. Description*

This document contains all the records of the functions tested and what are the results of the said testing executions.

• *9.2.2. Activity and event entries*

Every week of the summer, different documentations has been reported and necessary changes are being updated in the same week.