**SOFTWARE REQUIREMENTS SPECIFICATION**

**STUDENT OUTCOMES ASSESSMENT AND EVALUATION PLAN**

Version 3.0

January 2019

Student Outcomes Assessment and Evaluation Plan for CPE

Members:

Antolin, Arryll Mori

Antonio, Marlon

De Pano, Claudette Jean

Jawod, Danielle Nicole

Reyes, Jean Marc

Santos, Denzel Antoinette

**Table of Contents**

1. **Introduction …………………………………………………………………………… 3**
   1. *Purpose* 3
   2. *Scope* 3
   3. *Definitions, Acronyms, and Abbreviations* 4
   4. *References* 4
   5. *Overview* 4
2. **The Overall Description …………………………………………………………….. 4**
   1. *Product Perspective* 4
   2. *Product Functions* 5
   3. *User Characteristics* 6
   4. *Constraints*  6
   5. *Assumptions and Dependencies* 6
3. **Specific Requirements ……………………………………………………………… 7**
   1. *Software Product Features Requirements* 7
      1. *System Environment*  7
      2. *Professor Use Case* 8
      3. *Administrator Use Case* 9
   2. *External Interface Requirements* 11
      1. *User Interface Requirements*  11
      2. *Software Interface Requirements* 11
      3. *Communication Interface Requirements* 12
4. **System Features ……………………………………………………………………. 12**
5. **Other Non-Functional Requirements …………………………………………… 12**
   1. ***Performance Requirements …………………………………………………..* 12**
      1. *Capacity*  12
      2. *Dynamic Requirements* 13
      3. *Quality* 13
   2. ***Software System Attributes …………………………………………………..* 13**
      1. *Reliability* 13
      2. *Availability* 13
      3. *Security* 13
      4. *Maintainability* 13
   3. ***Business Rules …………………………………………………………………* 14**
6. **Other Requirements ……………………………………………………………….. 14**

Appendix A: Glossary 14

1. **Introduction**

The software, Student Outcomes Assessment and Evaluation Plan, is to be developed for CPE Department in Mapua University. The software is a computerized assessment plan programmed in C# and a web application framework using ASP.NET that contains student outcomes coming from a course syllabus, performance indicators, summative course, assessment tool to be graded from blackboard account or by the professors, assessment targets and results, quarter and school year, evaluation, and recommendation. It is an online evaluation plan for the professors for easier computing and evaluating of the students’ grades if the desired assessment of Student Outcomes has been met. The program imports data from Microsoft Excel to be processed. The Excel file contains the grades of the students that is manually recorded by the professors or downloaded from the Blackboard account. After that, the program then automatically computes the grades of the students from the spreadsheet and will generate the Assessment Plan through Excel file and is stored to the Database using SQL. Through this program, the user interacts with a user-friendly interface because the interface is simple, able to get things done more quickly, and uses strategic colors for easy understanding the layout of the program.

* 1. ***Purpose***

This SRS defines External Interface, Performance and Software System Attributes requirements of the program. This document is intended for the following group of people:

* Developers for the purpose of maintenance and new releases of the software.
* Management of the student outcomes assessment and evaluation plan.
* Document writers.
* Testers.
  1. ***Scope***

This document applies to student outcomes assessment and evaluation plan software. This software facilitates two users – the professor and the admin. The admin is the one responsible for supervising the staff and monitor their work and has access to all information from database which includes all accounts and contents of data needed to generate the student outcomes assessment and evaluation plan. Whereas the professor is the other user who performs online assessment and evaluation of the students that would be generated to Microsoft Excel and imports the excel file using Visual Studio IDE without computing the grades by a pen and paper way. This software provides easier to compute grades and prevents human errors when evaluating.

* 1. ***Definitions, Acronyms, and Abbreviations***

|  |  |
| --- | --- |
| **ASP.NET** | Web application framework |
| **Assessment** | The evaluation or estimation of the nature, quality, or ability of someone or something |
| **C#** | Object-Oriented programming language |
| **Database** | A structured set of data held in a computer, especially one that is accessible in various ways |
| **Internet** | An interconnected system of networks that connects computers around the world via the TCP/IP protocol. |
| **SQL** | Structured Query Language |
| **SRS** | Software Requirements Specification |
| **Syllabus** | An outline of the subject in a course of study or teaching |

* 1. ***References***

The references for the above software are as follows:

1. Retrieved from www.asp.net
2. IEEE. Software Requirements Specification Std. 830-1993
   1. ***Overview***

Section 1.0 discusses the purpose and scope of the software. Section 2.0 describes the overall functionalities and constraints of the software and user characteristics. Section 3.0 details all the requirements needed to design the software.

1. **The Overall Description**
   1. ***Product Perspective***

The product perspective of the software:

* The software allows two users – the admin and the professor.
* The software allows the admin to have access in all the contents of the program.
* The software allows the professor (another user) to provide the excel files containing the grades from Microsoft Excel as the system import these. Such files from the website [www.mapua.blackboard.com](http://www.mapua.blackboard.com) which, in computing, the format is Microsoft Excel Open XML Format (XLSX) or to be provided by the professor.
* Must be connected to the Internet when using the system.
* This software allows menu page for the admin and the user. It also generates assessment plan page for computing and evaluation.
  1. ***Product Functions***

The major functions that the program performs are described as follows:

* **Account Login: -** The user must login first before using the program.
* **User Section:** - For the user’s (professor) menu page, he can either generate the new assessment plan, download the current assessment plan from the database or change the current account password.
  + **Generate Assessment Plan:** - It will ask the user to import the necessary excel files containing the students’ grades per course after which inputs the starting quarter and SY. The assessment plan is made in excel file that contains necessary student outcomes, program indicators, courses, assessment tools, and the target set number of students with grades of greater than or equal to 3.5 or 70%. The program will then import the necessary excel files for each course to compute for the students’ average coming from the assessment tool. It also processes per quarter and SY. The excel file that contains the assessment plan is generated and updates the Database (SQL).
  + **Download Assessment Plan:** - The system shall retrieve the current assessment plan stored from the database and allows the user to download it.
  + **Change Password:** - The system will ask the user to input the current password and new password for the account. The security requirements shall be met upon changing the current password.
* **Admin Section:** - the admin can create or delete account, change current password, view logs, and edit data such as student outcomes, performance indicators, summative courses, assessment tools, and target. The admin can also manage information inside the database which includes the necessary information such as creation or deletion of user accounts, altering the relationship between student outcomes, performance indicators, courses, assessment tools, and target. It has the right to access the overall function of the program.
  1. ***User Characteristics***

There are different kind of users that will be interacting with the system. The intended user of the software are as follows:

* **Administrator:** The admin has the right to access all the contents in the system.
  + **Professor:** The user (professor) is the one who inputs the starting quarter and school year which lets the system identify which courses, containing specified assessment tools, are to be imported by the program to evaluate the grades. The user is the one responsible in creating the assessment plan.
  1. ***Constraints***

The major constraints that the project has are as follows:

* The software requires reliable Internet connection to use the program.
  + The database used should be SQL and the web application framework used is ASP.NET.
  + The program is dependent on the naming convention of such excel files to be imported by the system containing the grades of students per course and such assessment tools used.
  + Before the process in evaluating the assessment is carried out, the system first computes the grades from the specified assessment tool for each course.
  + The user (professor) does not have any right to edit or delete an account and alter contents of data needed to generate the student outcomes assessment and evaluation plan. The user can input only the necessary needed.
  1. ***Assumptions and Dependencies***

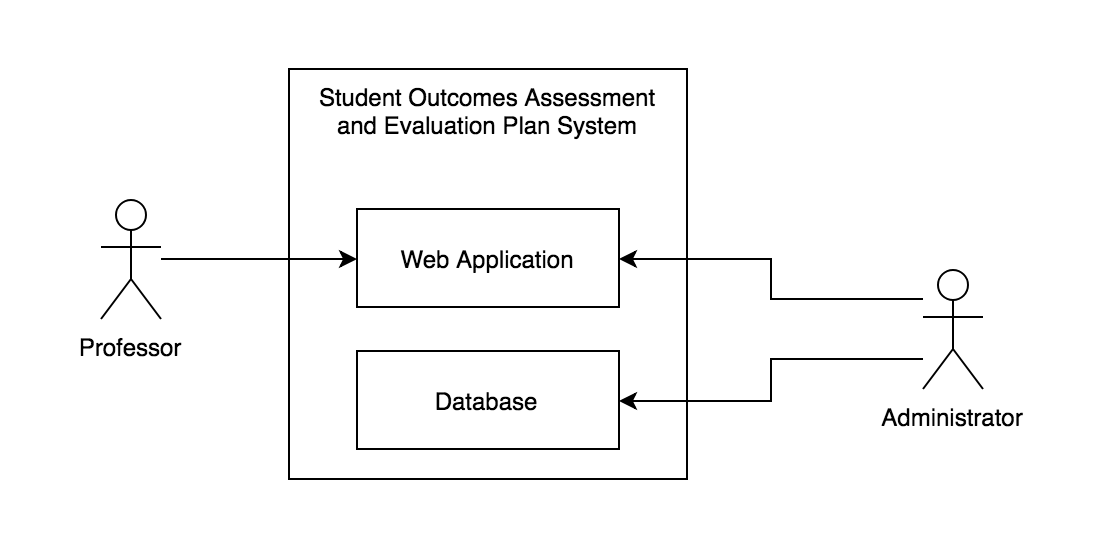
The requirements stated in the SRS could be affected by the following factors:

* + The major dependency that the program might face is the changes need to be incorporated with the changes in the student’s blackboard account regarding the grades in their assessment tasks. As the policies changes the system or the data in the database needs to be updated with the same immediately. This change is required for the evaluation of the students’ performance.
* The database used in this system is SQL.
* The system is dependent on the Internet when using the program.
  + The operating system of the personal computer must work under either Windows 7, 8, or 10.
* Microsoft Excel must be 2007 or to present.

1. **Specific Requirements**

In this section, the specific requirements of the system will be discussed and described in two major categories: software product features and external interfaces. Each category contains subsections that are all equally important for the system to perform at its maximum functionality.

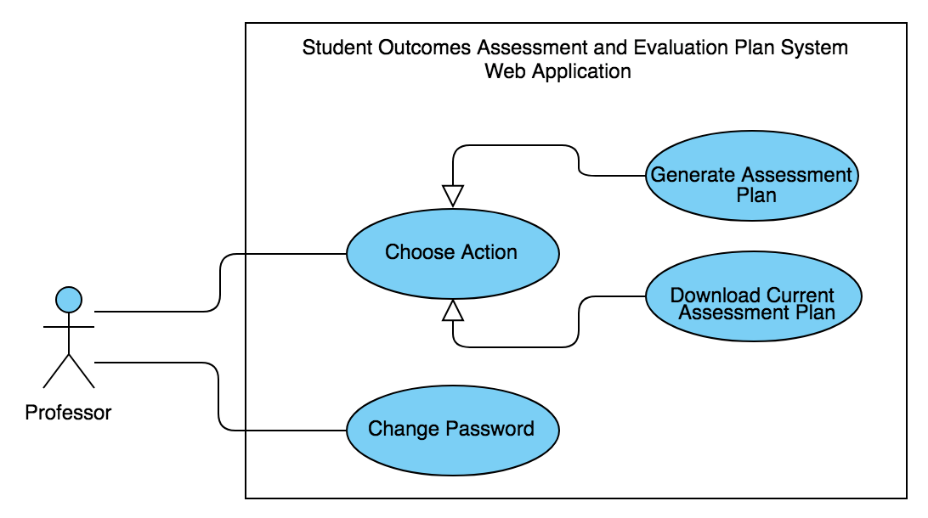
* 1. ***Software Product Features Requirements***
     1. ***System Environment***



**Figure 1.** System Environment

The Student Outcomes Assessment and Evaluation Plan System has two active actors: Professor/s and the Administrator. The former accesses the Web Application through the Internet while the latter accesses the entire system directly.

* + 1. ***Professor Use Case***

The Professor has the following set of use cases:

**Figure 2.** Professor Use Cases

*Note: Before this use case can be initiated, the user must first login for the system to determine whether he/she is a professor or the admin. In the case that the user is a Professor, the system will proceed with the steps below.*

**Use Case:** Choose Action

**Brief Description:** The Professor chooses an action, whether to generate an assessment plan or download the current data available.

**Initial Step-by-Step Description:**

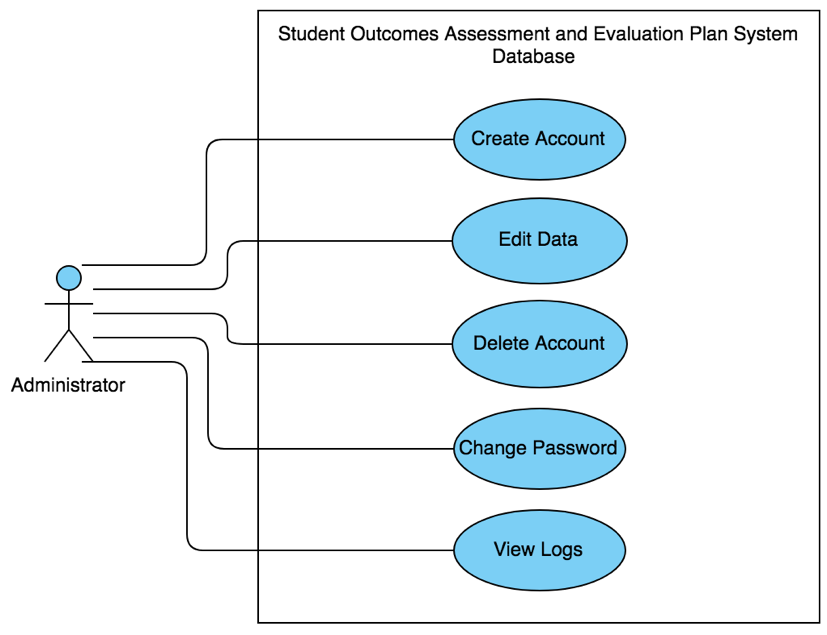
1. The Professor selects *“Choose Action”.*
2. The system presents a choice whether to generate assessment plan or download current assessment plan.
3. The Professor chooses to generate or download.
4. If the Professor is to generate an assessment plan, he/she will be asked to input the necessary information to do so; else, the system allows the Professor to download the current assessment plan available from the database.
5. The Professor imports the necessary excel files containing the students’ grades per course after which edits or fills in the information required then submits the form.
6. The system imports the necessary excel files from each course, containing specified assessment tools, and computes for the students’ average ratings per quarter and school year.
7. The system will then process and generate the excel file that contains the assessment plan then updates the database.

**Use Case:** Change Password

**Brief Description:** The Professor changes account password.

**Initial Step-by-Step Description:**

1. The Professor selects *“Change Password”.*
2. The system displays the *“Change Password*” window.
3. If the Professor is to change the current account password, he/she will be asked to input the necessary information to do so.
4. The Professor edits or fills in the information required then submits the form.
5. The system then saves changes and goes back to the initially opened window.
   * 1. ***Administrator Use Case***

The Administrator has the following set of use cases:

**Figure 3.** Administrator Use Cases

*Note: Before this use case can be initiated, the user must first login for the system to determine whether he/she is a professor or the admin. In the case that the user is the Administrator, the system will proceed with the steps below.*

**Use Case:** Create Account

**Brief Description:** The Admin creates a new faculty member account.

**Initial Step-by-Step Description:**

1. The Administrator selects *“Create Account”*
2. The system displays a window with the necessary information to be filled out.
3. The Administrator fills out all the required data and submits the form.
4. The system verifies if all the required fields are filled out.
5. The system saves and updates the data in the database.

**Use Case:** Delete Account

**Brief Description:** The Professor deletes an existing faculty member’s account.

**Initial Step-by-Step Description:**

1. The Administrator selects “*Delete Account*”.
2. The system presents names of all the faculty members with an account.
3. The Administrator chooses which account is to be deleted.
4. The system verifies if the Administrator wants to delete the account.
5. If the Administrator confirms to delete the chosen account, the system proceeds to step 6; else, return to step 2.
6. The system then saves and updates the database.

**Use Case:** Edit Data

**Brief Description:** The Administrator edits information in the assessment plan.

**Initial Step-by-Step Description:**

1. The Administrator selects “*Edit Data*”.
2. The system displays the program’s assessment plan window.
3. The Administrator edits information such as performance indicators, student outcomes, and the like.
4. The system verifies if the Administrator wants to edit the information.
5. If the Administrator confirms to edit the information, the system proceeds to step 6; else, return to step 2.
6. The system then saves and updates the database.

**Use Case:** Change Password

**Brief Description:** The Administrator changes account password.

**Initial Step-by-Step Description:**

1. The Administrator selects *“Change Password”.*
2. The system displays the *“Change Password*” window.
3. If the Administrator is to change the current account password, he/she will be asked to input the necessary information to do so.
4. The Administrator edits or fills in the information required then submits the form.
5. The system then saves changes and goes back to the initially opened window.

**Use Case:** View Logs

**Brief Description:** The Administrator displays all the processes done by the users when generating an assessment plan.

**Initial Step-by-Step Description:**

1. The Administrator selects *“View Logs”.*
2. The system displays the *“View Logs*” window.
3. From the “*View Logs*” window, the system displays all the processes executed by users when generating an assessment plan. The filenames are formatted as: “*AssessmentPlan\_accountID\_processID*” to distinguish which and whose log is it.
4. The Admin then exits the window.
   1. ***External Interface Requirements***

***3.2.1 User Interface Requirements***

The system’s user interface must be uncomplicated to work around with and must function efficiently and effectively. Here, the following displays will be provided:

1. First, a login page will be displayed where the user must enter the necessary login information for the system to determine whether he/she is a Professor or the Administrator.
2. An unsuccessful login will lead to a reattempt screen where the user can retype his or her information. A successful login, on the other hand leads to another window displaying information in coordination with the need of the user.
3. If the logged in user is a Professor, the system will display the following options: *Generate Assessment Plan, Download Current Assessment Plan, and Change Password.* From here, the Professor can choose any of the three and proceed to filling out the required pieces of information.
4. If the logged in user is the Administrator, the system will display a window where he/she can: *Create an Account, Delete an Account, Edit Data, Change Password, or View Logs.*
5. Once changes have been made, the system will then save the data and update the database.

***3.2.2 Software Interface Requirements***

For the system to work at its maximum functionality, it must be in coordination with various other software. That being said, the following software interface requirements must be fulfilled:

* The database that keeps record of the generated student outcomes assessment and evaluation plan shall communicate or should be done in MS SQL Server.
* The source codes of the system should be done on Microsoft Visual Studio 2013 or later.
* The assessment plan data of the system should be done on Microsoft Excel 2007 or later.
  + 1. ***Communication Interface Requirements***

The program needs to communicate with Mapua Blackboard for various functions such as access to student accounts for outcomes assessment and evaluation plan.

1. **System Features**

* ***Digital Student Outcomes Assessment and Evaluation Plan System***

Unlike the conventional way of generating student outcomes assessment and evaluation which is by manually inputting information, this system is designed to provide users with a user-friendly, digitized, and efficient approach to computing and evaluating students’ grades.

* ***Validity Check***

To access the system, the user must be a bona fide faculty member of Mapúa’s CPE department and must be able to correctly input his/her login information.

* ***Error Handling***

In case of inaccuracies with the information entered, fitting error messages will be prompted.

1. **Other Nonfunctional Requirements**
   1. **Performance Requirements**

**Capacity**

• The program should be able to accommodate all uploaded files and generate an output.

**Dynamic Requirements**

• The speed of account verification will be dependent on the content of the database.

• The uploading speed of the grades into the web application will be dependent on the user’s internet connection.

• The speed of generating file is dependent on the number of files uploaded.

• The operating system of the personal computer (PC) must work under either Windows 7,8 or 10.

**Quality**

The main objective is to develop a quality software. To measure the quality of the software, the following criteria are presented:

**•** Consistency –the code of the program should be consistent as to the style of the programmer.

**•** Test Cases –All functions must be thoroughly tested.

**5.2 Software System Attributes**

**Reliability**

**•** The user can redownload the generated file just in case the system provided a corrupted file.

**Availability**

• If there is a problem encountered on the system, the program will be redirected to the Login page.

• The user can only access the program/app when he or she has only an internet connection.

**•** The process of uploading the file/s should be consistent for the file to be generated has an accurate result.

**Security**

**•** The password will have a minimum of 8 characters length.

• The password should also contain 1 upper, lower, number, and a special character in the password.

**Maintainability**

• There will be a process log so that the admin can monitor the processes made by the users when generating an assessment plan.

**5.3 Business Rules**

The business rules are as follows:

• The administrator is the only authorized personnel to access all the necessary details and regulations of the software. They can also manage (add, delete, and edit) information inside the database.

• The admin of the system performs the following:

* Can manage accounts
* Can change password
* Can edit courses, assessment tools, target, SO, and performance indicators
* Maintain backup of all information entered in the system.
* Corrects any mistakes discovered in the database.
* Update the database if changes are made throughout the usage of the software.

• The users of the system perform the following:

* Can change password
* The user can provide the files containing the information about the students
* Generate a file containing the result
* Download generated file from database

1. **Other Requirements**

None

Appendix A: Glossary

ASP.NET - Active Server Pages, an open-source server-side web application framework designed for web development to produce dynamic web pages

Assessment - The evaluation or estimation of the nature, quality, or ability of someone or something

C# - Object-Oriented programming language

Database - A structured set of data held in a computer, especially one that is accessible in various ways

Internet - An interconnected system of networks that connects computers around the world via the TCP/IP protocol

SQL - Structured Query Language

SRS - Software Requirements Specification

Syllabus - An outline of the object in a course of study or teaching