Test Design Specification

**By: Tan Le, Hunter Newton, Luis Gonzales, Donald Matthews, Brian Cobo**

This test design specification will demonstrate and test all the available button and function within the xMenAcademy website. We will be extensively testing from both admin interface and the user interface to make sure everything is working properly. Below, we will be conducting various type of test and the result for each test cases.

**Table of Contents**

[1 Introduction 2](#_Toc326083786)

[2 Test Objective 2](#_Toc326083787)

[3 Detailed Testing Strategy 2](#_Toc326083788)

[3.1 Unit Testing 2](#_Toc326083789)

[3.1.1 White Box Testing 2](#_Toc326083790)

[3.1.1.1 Basis Path Testing – Tree Repository Module 2](#_Toc326083791)

[3.1.2 Test Cases 3](#_Toc326083792)

[3.1.3 [TC001] Install mobile application 3](#_Toc326083793)

[3.1.4 [TC002] MA User login 4](#_Toc326083794)

[3.1.5 [TC003] MA User logout 5](#_Toc326083795)

[3.1.6 [TC004] Mobile Application Settings 5](#_Toc326083796)

[3.1.7 [TC005] Start Transmitting 6](#_Toc326083797)

[3.1.8 [TC006] Start Transmitting 7](#_Toc326083798)

[3.1.9 [TC007] Resume Transmitting 7](#_Toc326083799)

[3.1.10 [UC008] Stop Transmitting 8](#_Toc326083800)

[3.1.11 [TC009] WA User Login 8](#_Toc326083801)

[3.1.12 [TC010] WA User Logout 9](#_Toc326083802)

[3.1.13 [TC011] WA User Registration 9](#_Toc326083803)

[3.1.14 [TC012] WA View Route 10](#_Toc326083804)

[3.1.15 [TC013] WA Rename Route 11](#_Toc326083805)

[3.1.16 Black Box Testing 12](#_Toc326083806)

[3.2 Integration Testing 12](#_Toc326083807)

[3.2.1 Incremental Testing 12](#_Toc326083808)

[3.3 System Testing 13](#_Toc326083809)

[3.3.1 Function Validation Testing 14](#_Toc326083810)

[3.3.2 Performance testing 14](#_Toc326083811)

[4 Pass/Fail Criteria 14](#_Toc326083812)

[4.1 Test Log 14](#_Toc326083813)

[4.2 Shipping or Live Release 14](#_Toc326083814)

[4.2.1 Shipping/Live Release Entry Criteria 14](#_Toc326083815)

[4.2.2 Shipping/Live Release Exit Criteria 15](#_Toc326083816)

[5 References 16](#_Toc326083817)

**1 Introduction**

This document provides all the test documentation that will be tested including white box testing, black box testing, integration test any many more. For each test cases, we will guide you through all the specific steps and record all the result to the items that were tested.

**2 Test Objective**

The objective for this test is to provide a test plan and all the necessary information in order to perform the tests. By doing so, we hope to minimize the error and improving on our products if any fault occurs. This will help us to stay on track and see our progress while doing the project. The tester will run the test through all the cases and scenarios; and the test result will be recorded and reported back to the developer to adjust and fix any bug within the code.

**3 Detailed Testing Strategy**

**3.1 Unit Testing**

Unit testing will be done within the code to check any logic errors or any bugs. Making sure every function is working as intended and to see all the database is link correctly to the main page.

**3.1.1 White box Testing**

In white box testing, we will be testing both input and output of the code. This test will only focus on the code and structure instead of the functionality of it. We will be testing each function individually and this will be done by the development/tester team.

**3.1.2 Test Cases**

**3.1.3 [C001] Homepage**

|  |  |
| --- | --- |
| [ID] Name | [C001] Homepage |
| Summary | The homepage of xMenAcademy |
| Items to be tested | The link that will take you to the site |
| Users | All users |
| Pre-conditions | The user must have internet |
| Basic course of events | 1. The user will click on the link or type in the website |
| Input | N/A |
| Expected output | The homepage will appear with a welcome message, a video, and a login button |

**3.1.4 [C002] Sign in Button**

|  |  |
| --- | --- |
| [ID] Name | [C002] Sign in Button |
| Summary | This button will lead to another page where you can login with your information |
| Items to be tested | The sign button that will take you to another page |
| Users | All users |
| Pre-conditions | The user is on the homepage |
| Basic course of events | 1. The user will click on the sign in button on the top right corner or the middle |
| Input | N/A |
| Expected output | The login button will lead you to another page where you input your credentials |

**3.1.5 [C003] XmenAcademy User Submit**

|  |  |
| --- | --- |
| [ID] Name | [C003] XmenAcademy User Submit |
| Summary | Users login to access their personal information |
| Items to be tested | The submit button that will take the users to their dashboard |
| Users | All users |
| Pre-conditions | 1. The user is at the sign in page 2. The user must have an account already |
| Basic course of events | 1. The user will input their personal information 2. After inputting the information, the user will hit the submit button |
| Input | 1. Username 2. Password |
| Expected output | After inputting the correct credentials, it will lead you to your personal page/dashboard |

**3.1.6 [C004] Administrator “Create Course” Button**

|  |  |
| --- | --- |
| [ID] Name | [C004] Administrator “Create Course” Button |
| Summary | This button will lead you to another page where you can submit and create your courses |
| Items to be tested | The button that will lead you to a different page (Create Course) |
| Users | Administrator |
| Pre-conditions | The user is on administrator dashboard |
| Basic course of events | 1. The user will click on the button “Create Course” on top left corner |
| Input | N/A |
| Expected output | The button will take you to a “Create Course” page where you can input the courses information |

**3.1.7 [C005] Administrator “Create Course” Tab**

|  |  |
| --- | --- |
| [ID] Name | [C005] Administrator “Create Course” Tab |
| Summary | This page will create and save your courses information into the database for later use |
| Items to be tested | Submit button in “Create Course” tab |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the “Create Course” tab after pressing the “Create Course” Button |
| Basic course of events | 1. The user will input the information 2. After inputting the information, the user will hit the submit button |
| Input | 1. Course Name 2. Course Description |
| Expected output | The course will be saved in the database with an assigned number (ID) attached to it. The information will also display in the administrator dashboard |

**3.1.8 [C006] Administrator “Teacher Course Registration” Button**

|  |  |
| --- | --- |
| [ID] Name | [C006] Administrator “Teacher Course Registration” Button |
| Summary | This button will lead you to another page where you can assign the teacher to the corresponding course |
| Items to be tested | The button that will lead you to a different page (Teacher Course Registration) |
| Users | Administrator |
| Pre-conditions | The user is on administrator dashboard |
| Basic course of events | 1. The users will click on the button “Teacher Course Registration” on top left corner |
| Input | N/A |
| Expected output | The button will take you to a “Teacher Create Course” page where you can input the course and teacher information |

**3.1.9 [C007] Administrator “Teacher Course Registration” Tab**

|  |  |
| --- | --- |
| [ID] Name | [C007] Administrator “Teacher Course Registration” Tab |
| Summary | This page will create and assign the teacher to a specific course into the database for later use |
| Items to be tested | Submit button in “Teacher Course Registration” tab |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the “Teacher Course Registration” tab after pressing the “Teacher Course Registration” Button |
| Basic course of events | 1. The user will input the information 2. After inputting the information, the user will hit the submit button |
| Input | 1. Course Catalog ID 2. Teacher ID 3. Semester |
| Expected output | The course will be assigned to the teacher through ID into the database. The information will also display in the administrator dashboard |

**3.1.10 [C008] Administrator “Student Course Registration” Button**

|  |  |
| --- | --- |
| [ID] Name | [C008] Administrator “Student Course Registration” Button |
| Summary | This button will lead you to another page where you can assign the student to the corresponding course |
| Items to be tested | The button that will lead you to a different page (Student Course Registration) |
| Users | Administrator |
| Pre-conditions | The user is on administrator dashboard |
| Basic course of events | 1. The users will click on the button “Student Course Registration” on top left corner |
| Input | N/A |
| Expected output | The button will take you to a “Student Course Registration” page where you can input the student and course information |

**3.1.11 [C009] Administrator “Student Course Registration” Tab**

|  |  |
| --- | --- |
| [ID] Name | [C009] Administrator “Student Course Registration” Tab |
| Summary | This page will create and assign the student to a specific course into the database for later use |
| Items to be tested | Submit button in “Student Course Registration” tab |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the “Student Course Registration” tab after pressing the “Student Course Registration” Button |
| Basic course of events | 1. The user will input the information 2. After inputting the information, the user will hit the submit button |
| Input | 1. Course Catalog ID 2. Student ID |
| Expected output | The course will be assigned to the student through ID into the database. The information will also display in the administrator dashboard |

**3.1.12 [C010] Administrator “User Registration” Button**

|  |  |
| --- | --- |
| [ID] Name | [C010] Administrator “User Registration” Button |
| Summary | This button will lead you to another page where you can register a student in the system |
| Items to be tested | The button that will lead you to a different page (User Registration) |
| Users | Administrator |
| Pre-conditions | The user is on administrator dashboard |
| Basic course of events | 1. The users will click on the button “User Registration” on top left corner |
| Input | N/A |
| Expected output | The button will take you to a “User Registration” page where you can input the student information |

**3.1.13 [C011] Administrator “User Registration” Tab**

|  |  |
| --- | --- |
| [ID] Name | [C011] Administrator “User Registration” Tab |
| Summary | This page will create and save new user information into the database for later use |
| Items to be tested | Submit button in “User Registration” tab |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the “User Registration” tab after pressing the “User Registration” Button |
| Basic course of events | 1. The user will input the information 2. After inputting the information, the user will hit the submit button |
| Input | 1. First Name 2. Last Name 3. Email 4. Username 5. Role |
| Expected output | The new user (admin/student/teacher) will be saved in the database with an assigned number (ID) attached to it. The information will also display in the administrator dashboard |

**3.1.14 [C012] Administrator “User Registration” Roles**

|  |  |
| --- | --- |
| [ID] Name | [C012] Administrator “User Registration” Roles |
| Summary | By selecting the roles (admin/teacher/student) the program will know which table it should assign the user to while inside the database |
| Items to be tested | The “role” tab drop-down menu |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the “User Registration” tab after pressing the “User Registration” Button 3. The user must have all the information in “User Registration” tab fill in |
| Basic course of events | 1. The user opens the drop-down menu in “roles” 2. The user will select either teacher, student, or admin |
| Input | N/A |
| Expected output | By assigning the role, the computer will generate an ID and put the corresponding user into the correct table in the database |

**3.1.15 [C013] Administrator “Edit” Button**

|  |  |
| --- | --- |
| [ID] Name | [C013] Administrator “Edit” Button |
| Summary | By pressing edit, you can change the student, admin, or teacher information |
| Items to be tested | The “Edit” button |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. A student/teacher/admin already register |
| Basic course of events | 1. The user will press edit button on the register person |
| Input | N/A |
| Expected output | Take you to a different page where you can adjust any personal information |

**3.1.16 [C014] Administrator “Delete” Button**

|  |  |
| --- | --- |
| [ID] Name | [C014] Administrator “Delete” Button |
| Summary | By pressing “delete” button, you can delete the student, admin, or teacher information from the database |
| Items to be tested | The “Delete” button |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. A student/teacher/admin already register |
| Basic course of events | 1. The user will press delete button on the register person |
| Input | N/A |
| Expected output | Delete the person information from the database and refresh the page |

**3.1.17 [C015] Administrator “Main” Tab**

|  |  |
| --- | --- |
| [ID] Name | [C015] Administrator “Main” Tab |
| Summary | By pressing the tab on top of the screen (student/teacher/administrator/course catalog/available course) you can view the person or course information in the corresponding tab |
| Items to be tested | All the tab button (student/teacher/administrator/course catalog/available course) |
| Users | Administrator |
| Pre-conditions | 1. The user is an administrator 2. The user is on the dashboard |
| Basic course of events | 1. The user will press on each tab (student/teacher/administrator/course catalog/available course) |
| Input | N/A |
| Expected output | The dashboard will change to the information that you want depends on which tab you press |

**3.1.18 [C016] Teacher “Grade Book” Button**

|  |  |
| --- | --- |
| [ID] Name | [C016] Teacher “Grade Book” button |
| Summary | By pressing “Grade Book” button, you can see and adjust all the student grades within a specific course |
| Items to be tested | The “Grade Book” button |
| Users | Teacher |
| Pre-conditions | 1. The user is a teacher 2. The user is on the dashboard |
| Basic course of events | 1. The user will press “Grade Book” button on a specific course |
| Input | N/A |
| Expected output | Take you to a different page where you can view all the student in a specific course |

**3.1.19 [C017] Teacher “Grade Book” Button**

|  |  |
| --- | --- |
| [ID] Name | [C017] Teacher “Update Grades” button |
| Summary | By pressing “Update Grades” button, it will lead you the update grades page where you can adjust the exam score |
| Items to be tested | The “Update Grades” button |
| Users | Teacher |
| Pre-conditions | 1. The user is a teacher 2. The user is in the grade book tab |
| Basic course of events | 1. The user will press “Grade Book” button on a specific course |
| Input | N/A |
| Expected output | Take you to a different page where you can adjust the student score |

**3.1.20 [C018] Teacher “Submit” Grades**

|  |  |
| --- | --- |
| [ID] Name | [C018] Teacher “Submit” Grades |
| Summary | By pressing “Submit” button in the grades page, it will record the grades in the database and also converting to the corresponding GPA |
| Items to be tested | The “Submit” Grades Button |
| Users | Teacher |
| Pre-conditions | 1. The user is a teacher 2. The user is in the update grades page |
| Basic course of events | 1. The user will press “Submit” on the page |
| Input | 1. Score for Exam 1 2. Score for Exam 2 3. Score for Final |
| Expected output | Record the grades into the database and take you back to the teacher dashboard. Also convert the grade into the corresponding GPA |

**3.1.21 [C019] Sign in for all user**

|  |  |
| --- | --- |
| [ID] Name | [C019] Sign in for all user |
| Summary | The sign in will take you to different dashboard depends on if you are a student, teacher, or admin |
| Items to be tested | The “sign in” button function |
| Users | All |
| Pre-conditions | 1. The user is the login page |
| Basic course of events | 1. The user will enter their personal information 2. The user will press sign in |
| Input | 1. Username 2. Password |
| Expected output | Take you to the corresponding dashboard depends on the person and role that were assigned |

**3.1.22 [C020] Change Password**

|  |  |
| --- | --- |
| [ID] Name | [C010] Change Password |
| Summary | This will allow you to change the original password to your own |
| Items to be tested | Change password page and submit button |
| Users | All |
| Pre-conditions | 1. The user is in change password page |
| Basic course of events | 1. The user will enter the new password 2. The user will press submit after |
| Input | 1. Password 2. Confirm password |
| Expected output | Change the password for the corresponding person |

**3.1.22 Black Box Testing**

Black box testing is used in order to verify and making sure everything is working properly. By using this testing method, we can see all the expected outcome and if the database is working properly. This will ensure all user have a smooth experience and not running into any logic error or bugs.

* Legal input values: These input and test are within the boundaries and the program should produce an expected result.
  + Username/password: by entering correct information, it should take you to the proper dashboard
  + Grades: By entering from 0-100 in the grade book, it will covert the grade into the corresponding GPA
* Illegal input values: These input and test are outside the boundaries and the program should either compensate or give no result
  + Username/password: by entering incorrect information, the user will not be able to log in
  + Grades: By entering a character in the grade book, the page will force you to enter a valid decimal. If the number is outside of 0-100 then the GPA will be 4.00 if its above 100 and GPA will be 0.00 if its below 0

This black box testing ensures everything is working properly even if its an invalid input.

**3.2 Integration Testing**

**3.2.1 Incremental Testing**

There will be two module that we will mainly integrate for this testing: the GUI and the backend module. This test helps the team to stay on track and to see what step we need to do next. By using the GUI, we can see if all the button is working properly and taking you to the right place. As for the backend module, we can see how the database work and if its writing data properly.

**Module 1 – Graphic User Interface (GUI) Module**

This module provides a layout where the user can test out different tab and button. This module shows if all the tabs/button are working properly (student/teacher/admin/courses) and to see if it is taking us to the proper pages.

**Module 2 – Backend Module**

This module provides us with all the behind the scene data from the database. This module will be tested following the incremental method and to show if the data is working properly. We will keep adding new data for each tab to see if there is any problem while adding new information.

|  |  |
| --- | --- |
| **GUI Function** | **Database Function** |
| Edit | Read  Write  Store |
| Delete | Search  Delete |
| Submit | Insert  Store |
| Grade | Read  Insert  Store |
| Assign course/teacher/student | Search  Store |
| Role | Search  Assign  Store |