**Software Testing Plan**

**AAU Gold Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 04/24/2017 | 1.0 | Document creation | Joshua Potrawski, Jonathon Nowak |
| 04/25/2017 | 1.0 | Final Document Approval | Philip Parker, PM |

**Table of Contents**

1. INTRODUCTION 3

1.1. Purpose 3

1.2. Project Overview 3

1.3. Audience 3

2. TEST STRATEGY 3

2.1. Test Objectives 3

2.2. Test Assumptions 3

2.3. Test Principles 4

2.4. Data Approach 4

2.5. Scope and Levels of Testing 4

2.5.1. Exploratory 4

2.5.2. Functional Test 4

2.5.3. User Acceptance Test (UAT) 4

2.6. Test Effort Estimate 5

3. EXECUTION STRATEGY 5

3.1. Entry and Exit Criteria 5

3.2. Test Cycles 6

3.3. Test Metrics 6

3.4. Defect tracking & Reporting 6

4. TEST MANAGEMENT PROCESS 6

5. TEST ENVIRONMENT 6

1. INTRODUCTION

1.1 Purpose

This Software Test Plan (STP) will describe the testing strategies, techniques, and methodologies employed by the Gold Team during the Testing Phase of the Software Engineering Lifecycle.

1.2 Project Overview

The Above Average University Web Portal Tool provides a convenient environment for students, professors, unregistered users, and other faculty to communicate with one another. The portal presents information like upcoming events and recent news while also providing several interactive features such as students enrollment and file sharing between students and professors.

1.3 Audience

* Project team members perform tasks specified in this document and provide feedback related to testing.
* The Testing Engineers are responsible for writing and maintaining this document.
* The Project Manager will review and approve this document. Furthermore, he will assign testing and bug related tasks to team members.

2. TEST STRATEGY

2.1 Test Objectives

The objective of our testing is to minimize the total number of defects and ensure that the functionality of the application meets its requirements as described in the Software Requirement Specification (SRS) document.

2.2 Test Assumptions

* The database is correctly populated before testing begins.
* Performance and speed testing will not be considered for this application.
* Testing engineers have adequate knowledge and experience for the task.
* There is no downtime due to testing or defect fixes.

2.3 Test Principles

* Testing will be focused on meeting the design plans and detailed requirements laid out by the Software Architects and Requirements Engineer, respectively.
* Testing procedures will be common and consistent for all members engaged in testing activities.
* Testing procedures will be well-defined to more easily verify and compare tested element, i.e. expected vs. actual outputs or behavior.
* Testing involves emulating a user’s experience, therefore testing environment and data directly presents the AAU.
* Testing will be a repeatable and measurable activity.
* Testing will be composed of comparing expected results against the actual results.

2.4 Data Approach

To assist in ensuring that the application is fully functional, all testing activities will be completed while using actual, non-fictional data from the Above Average University.

2.5 Scope and Levels of Testing

2.5.1 Exploratory Test

The exploratory testing phase has the purpose of exploiting any obvious or critical defects in the application. These types of testing activities require no external software tools or scripts.

2.5.2 Functional Test

Functional testing will be performed to check the reliability of each function by feeding input values to each function and comparing the expected output with the actual output. The input will cover all cases, especially focusing on the boundary input values which are most commonly defective.

2.5.3 User Acceptance Test

This test focuses on validating the business logic, meaning allowing the end user to complete a final review of the product. We will accomplish this by presenting a thorough demo of the product to Professor Ruijan Zhang who will provide feedback on the final product.

2.6 Test Effort Estimate

The testing effort will be approximated on a iterative basis. This means that when one iteration of testing has been completed, verification and validation processes will take place to conclude whether additional testing iterations are needed.

3. EXECUTION STRATEGY

3.1 Entry and Exit Criteria

In our plan, we test functionality through a series of tests to mimic a user’s experience in order to verify that particular functions exhibit the expected output/behavior. Examples are included below, split into an action (entry) and result (exit).

* **Action:** Click Login button located in the navigation bar.

**Result:** A drop-down menu appears, with fields for “username” and “password.”

* **Action:** A student user inputs their username and password into the Login menu.

**Result:** If their login info is correct, the student is redirected to their respective portal page.

* **Action:** Navigating to Course Search page.

**Result:** Correct page displays with search categories ready for input.

* **Action:** Choosing a course category in the Course Search page and submitting.

**Result:** Page displays all relevant courses related to the search input.

* **Action:** Student user registers for a course.

**Result:** Database is updated with new student enrollment in that particular course.

* **Action:** User logs out from any page.

**Result:** User is redirected to home page where an alert is displayed, indicating that the user has logged out.

* **Action:** Logged in user clicks an announcement from table.

**Result:** The full announcement appears in a modal window.

* **Action:** The logged in faculty member is a department head.

**Result:** A button is displayed which allows the department head to create an announcement. Once created, the page is refreshed with the updated announcement, viewable by all logged in users.

3.2 Test Cycles

3.3 Test Metrics

3.4. Defect tracking & Reporting

Covered in the Software Quality Assurance Plan (SQAP) and Software Verification and Validation Plan (SVVP) documents.

4. TEST MANAGEMENT PROCESS

Each team member is responsible for testing their added features and functionality. Once committed, other team members will review the addition and provide feedback.

The Project Manager is responsible for assigning tasks related to testing and defect resolution. Testing Engineers are responsible for ensuring that testing is carried out on all vital features and functionality.

5. TEST ENVIRONMENT

Tests are carried out personally, with team members logging the comparisons between expected versus actual output and behavior.