

Test Plan for UWA

Test Plan for Product/Feature

Overview

The test project is intended to verify and validate the quality of the User Web Application (UWA). There are three main pages associated with this application. Each of these pages are to be considered application functions. The functions are User Login, Registration, and Profile.

The application is being used to provide a target for students to assess their ability to exercise test analyst activities. It is known that the application in its first release has some defects. It is expected that students will attempt to find as many defects and issues as possible.

When the defects have been reported, changes will be made to the application and a new release will be published for the students to regression test the new release and confirm what is working and what is not in a closure report.

Test Objectives

The objectives of this test project will include the following:

Number	Description	<input checked="" type="checkbox"/>
1	User Login Page features & functions are working	<input type="checkbox"/>
2	User Registration Page features & functions are working	<input type="checkbox"/>
3	User Profile/Edit Page features & functions are working	<input type="checkbox"/>

Once the test events are completed, review this table to check the boxes of the pages that are working without defects open.

Related Materials

There are PDF documents that support the test project entitled:

- UWAAssessment 1
- UWAAssessment 2
- Test Environment Setup

Test Requirements

Test Requirements are listed here for the three application pages being developed. Coverage analysis in the next section will verify that all test requirements are being tested by at least one test case.

Test Requirement ID	Description	
TR_LoginUser_001	Verify that the system requests the user's username and password when the user requests to login.	
TR_LoginUser_002	Verify that the system checks the user's username and password against all registered users.	
TR_LoginUser_003	Verify that the system starts a login session when the user enters valid credentials and displays a welcome message based on the user's preferences.	
TR_LoginUser_004	Verify that the system requests the user to re-enter the password when an invalid password is entered.	
TR_LoginUser_005	Verify that the system goes back to step 2 when an invalid username is entered.	
TR_RegisterUser_001	Verify that clicking the register button on the home page takes the user to the registration page.	
TR_RegisterUser_002	Verify that the registration page requests a username and password.	
TR_RegisterUser_003	Verify that the registration page requests the user's name, password, and email address.	
TR_RegisterUser_004	Verify that the system checks that the username does not duplicate any existing registered usernames.	
TR_RegisterUser_005	Verify that the system determines the user's id and access level based on internal data.	
TR_RegisterUser_006	Verify that the system executes use case Register Edit Profile.	
TR_RegisterUser_007	Verify that the system starts a login session and displays a welcome message based on the user's	

	preferences.	
TR_General_001	Check page flow for gaps or access without requiring login.	
TR_General_002	Make sure all buttons are working as expected.	
TR_General_003	Check images and titles for appropriate ALT displays.	
TR_General_004	Verify no unexpected or unnecessary text exists on any page.	
TR_General_005	Are any entry text boxes too small or text too big?	
TR_General_006	Any text or page colors that are visual issues?	
TR_General_007	Any visual issues of any kind?	

Coverage Analysis

Coverage analysis data here

Test Requirement ID	Test Case ID	Test Events
TR_LoginUser_001		System/Integration, Regression
TR_LoginUser_002		System/Integration, Regression
TR_LoginUser_003		System/Integration, Regression
TR_LoginUser_004		System/Integration, Regression
TR_LoginUser_005		System/Integration, Regression

Testing Strategy

Explain the reasoning for selecting the relevant testing strategy, for example:

- ☒ Relevant platforms
- ☐ **Testing limitations**
 - Computer configuration for optimal application execution
 - Optimal** Computer Specifications
 - CPU → Processor Intel(R) Core(TM) i3-8130U CPU @ 2.20GHz, 2208 Mhz,
 - 2 Core(s), 4 Logical Processor(s)
 - Installed Physical Memory (RAM) 16.0 GB
 - Total Virtual Memory 20.8 GB
 - Acceptable** Computer Specification
 - AMD Ryzen 3 3250U with Radeon Graphics, 2500 Mhz,
 - 2 Core(s), 4 Logical Processor(s)
 - Installed Physical Memory (RAM) 16.0 GB
 - Total Virtual Memory 14.8 GB
 - Additional Needs**
 - Browsers: Chrome, Edge, or Firefox
 - Knowledge to GitHub and Git commands
- ☐ **Relevant testing environments**
 - Using an IDE on a local machine

- Installed MySQL Server and Workbench
- Installed Python and libraries
 - MySQL libraries
 - Flask libraries

☐ Team needs

Optional strategies to consider:

1. Smoke Testing
2. System/Integration Testing
3. Regression Testing
4. User Acceptance Testing
5. Batch Testing
6. Beta Testing

Required Preparations

The following board details the required preparations needed to get started:

Table

Testing Environment Requirements

Detail any relevant requirement, such as:

- ☒ Hardware
- ☐ Software
- ☐ **Users**
 - Admin and Test accounts are loaded into the database
 - Additional test accounts should be created via the user registration
 - The Admin and Test accounts probably have invalid passwords, so they are good for testing invalid password tests
 - Invalid users should be values that are not added to the application

- ☐ **Mock data**
- When creating new user accounts, a good practice is to assign the user ID and password the same value
 - The Registration page supports entry for two additional fields: email and account type
 - Specify the email format as XXX@example.com where XXX is the user ID value
 - The account type can be either "Admin" or "Member"

Scenarios to Be Tested

List relevant scenarios that will be tested.

TIP: Use a board with a Status Column to easily to see at a glance what's on track and what's marked done, directly from your monday workdoc.

Table

Scoped Out

List all scenarios/environments that will not be tested.

- *User Acceptance Testing*
- *Batch Testing*
- *Beta Testing*
- ...

Findings

