BNFIN Automated Testing Documentation

Robot Framework, Selenium

This documentation will outline the testing procedure for the automated testing suites used to test the BNFIN platform and its functionality.

Technologies used: Robot Framework, Selenium, Python.

**Robot Framework** is an open-source automation testing framework that uses a keyword-driven approach for acceptance testing and acceptance test-driven development (ATDD). It's designed to be easy to use by allowing users to write tests in a natural language-like syntax using keywords. Making for a flatter learning curve and easier for all stakeholders (whether competent programmers or not) to be able to read and understand the testing process.

**Selenium** is a widely used automation and test writing tool widely used for web browser automation. While Selenium itself can be integrated and used with a wide variety of programming languages (Python, Java, and C# being the most commonly used), it can be imported and integrated with Robot Framework via the **SeleniumLibrary** plugin, which allows the user to access the Selenium functionalities (such as clicking buttons, entering text, navigating through pages etc.) but using the simple Keyword syntax provided by Robot. Essentially eliminating the need to use any of the abovementioned programming languages.

Example of Selenium Syntax:

A black background with white text

Description automatically generated

Example of Robot Syntax for comparison:

A screen shot of a computer

Description automatically generated

**Python** will be integrated with Selenium for certain tests which may be difficult to achieve using Robot Framework alone. (Such as spell checker, file saving mechanisms, or complex conditional statements).

Features that will be tested:

* Login/Logout/Password reset
* Registration
* Navigation between links

Testing Limitations:

Due to the nature of Robot script and Selenium, and the use of “locators” (elements that tell the framework where to find specific elements on a web page, those could be a CSS element, a text element, xpath, link, ID etc...), the tests in this system heavily rely on the structure of the HTML in the source code of the original application. Should that source code change, some of the tests in this system may be rendered unusable or require an update.

Running the tests:

While the tests can be ran using a variety of means, from individual tests to group runs using a selected Tag, to scheduling specific tests using the task scheduler, the method chosen for this system is through the running of batch files that contain groups of related tests.

Clicking on each batch file will run the tests defined in it and open the “log.html” report that would display the test results.

The contents and tests that can be found in each file are as follows:

|  |  |  |
| --- | --- | --- |
| File Name | Overview of included tests | Detailed tests |
| {Sign in, sign out, registration, forgot password.bat} | This testing suite will check for the user’s ability to Login, Logout, Reset Password and Register | * Login in as: Reviewer BBI-1, BBI-2. Reviewer Tropicana-1, Tropicana-2. Applicants 1,2 and 3.  Lead reviewers- BBI, Tropicana.  Admin. * Logout. (As admin) * Successful password reset with valid fields. * Unsuccessful password reset with empty fields. * Unsuccessful password reset with invalid fields. * Unsuccessful password reset with invalid format fields. * Navigation from “forgot password” back to “login” page. |