**Write-Up**

**GitHub:** https://github.com/olomansaas/Phase3\_RestAssured.git

**Handling the Test Data and Executing It in Multiple Environments for User Module of an Application.**

**Description**

To create a framework for validating the test data using the web services for the authentication module

**Overview:**

**Class Name**: Assisted\_Project.

**Purpose**: Handling the Test Data and Executing It in Multiple Environments for User Module of an Application

**API Used:** reqres.in - A sample API for testing.

**Test Methods:**

Each method in this class is annotated with @Test, indicating that it's a test case. They are executed in the order of their priority.

1. **baseurl()**

Priority: 1

**Function:** Tests the base URL of the API.

**Process**:

Sends a GET request to the base URL of the API.

Expects a 404 status code, indicating the base URL alone is not a valid endpoint.

Logs the response for debugging.

2. **RegsiterUrl()**

Priority: 2

**Function:** Tests the user registration endpoint.

**Process:**

Prepares a map with user credentials (email and password).

Sends a POST request to the /register endpoint with the user data.

Expects a 200 status code, indicating successful registration.

Logs the response for validation.

3. **GetUser()**

Priority: 3

**Function:** Tests retrieving user information.

**Process:**

Sends a GET request to the /api/users endpoint.

Validates the response to ensure it contains specific user details (like email, first name, last name, and avatar of the first user).

Uses assertions to check the data in the response.

4. **Loggingtest()**

Priority: 4

**Function:** Tests logging capabilities on a known endpoint.

**Process**:

Sends a GET request to the /api/unknown endpoint.

Expects a 200 status code, indicating the endpoint is valid.

Logs the entire response for review.

**Conclusion**

1. The class systematically tests various aspects of the API, from basic URL validation to user registration and data retrieval.
2. It demonstrates the use of GET and POST requests and how to validate responses.
3. Logging is an essential part of each test for debugging and validation purposes.

4.The Assisted\_Project class effectively demonstrates the use of REST Assured for API testing in a structured and organized manner. Each test method serves a specific purpose, from checking the base URL's response to validating user registration and fetching user details. The class showcases essential practices in API testing, such as response validation, status code checking, and the use of HTTP methods (GET, POST).

5.The tests are prioritized to ensure a logical sequence of execution, which is important in cases where one test's outcome might depend on another. The use of logging throughout the test methods aids in debugging and provides clear insights into the API responses and the behavior of the endpoints under test. This class is a robust example for anyone looking to understand or implement API testing using REST Assured in Java