

# QA Engineer Assessment

## Objective

Demonstrate your ability to perform **manual testing**, **automation**, and **API testing** on sample applications to verify that core features work as expected.

## Instructions

1. Review the user story and acceptance criteria.
2. Create and document test cases.
3. Execute manual testing.
4. Report any bugs or issues found.
5. Automate the main web application test flow using **Playwright**.
6. Perform API testing using the **public JSONPlaceholder API**.
7. Submit your work in a **GitHub repository**

## Part 1: Web Application Testing

### Test Environment

- **URL:** <https://www.saucedemo.com/>
- **Login Credentials:**
  - **Username:** standard\_user
  - **Password:** secret\_sauce

### Feature to Test

1. Add to Cart
2. Checkout

### User Story

As a user, I want to add a product to my cart and complete the checkout process so that I can successfully purchase the item.

### Acceptance Criteria

- The user can add products to the cart.
- The user can proceed to checkout and complete the purchase.

## Part 2: API Testing

### URL To test

<https://jsonplaceholder.typicode.com>

### Tasks

Using Postman (or any API client), create and execute test cases for the endpoints below.

For each test, include in your submission:

- Request details (URL, method, headers, body if any)
- Response details (status code, headers, body)
- Your short analysis of the result
- Include **at least 1** edge case for each endpoint

### Endpoints to Test:

1. **GET /users** – Retrieve all users.
2. **GET /users/1** – Retrieve a specific user.
3. **POST /posts** – Create a new post with title and body.
4. **PUT /posts/1** – Update an existing post.

### Deliverables:

1. Test Documentation
  - Documented **test cases** for both web application and API testing
    - Formats can be in *Excel, Google Sheets, or Docs*.
  - For API tests: request details, response details, short analysis, and at least 1 negative/edge case per endpoint.
  - Screenshots/logs of executed test cases
2. Bug reports, *if any*.
  - Clearly written bug reports with steps to reproduce, expected vs actual results, and evidence (screenshots/logs).
3. Automation Script (Playwright)
  - Covering the main flow based on the requirements provided.

### Evaluation Criteria

Area of Evaluation	Description
Test Case Quality	Clarity, completeness, coverage of positive/negative/edge cases
Automation (Playwright)	Ability to script core flow, code readability, and maintainability
API Testing (Postman)	Proper use of requests, assertions, negative cases, and documentation
Results & Documentation	Clear bug reports, screenshots, and a professional presentation/walkthrough of results.