



QA Automation Engineer – Assignment

Overview

This assignment evaluates your manual testing, UI automation, API automation, unit testing, and an end-to-end (E2E) flow. Please follow the steps carefully and keep your work clear and easy to review.

Part A — Manual Testing (on Altera Institute website)

Target: <https://alterainstitute.com>

Goal: Perform exploratory manual testing and document at least 5 test cases.

Focus areas: Navigation, content correctness, forms, responsiveness, broken links/visual issues.

Deliverable: Create a file named 'ManualTesting.md' in the root of your repo with a table like:

#	Scenario	Steps	Expected Result	Actual Result	Severity	Priority	Evidence
---	----------	-------	-----------------	---------------	----------	----------	----------

Part B — Automation Testing

B1. UI Automation — SauceDemo

Target: <https://www.saucedemo.com/>

Creds (public demo): Username: standard_user | Password: secret_sauce

Use any framework: Playwright / Cypress / Selenium / WebdriverIO / TestCafe.

Recommended: Page Object Model (POM), reusable helpers, assertions, and test data.

- Test Case 1 — Successful Login

Steps:

1. Open site
2. Enter standard_user/secret_sauce
3. Click Login

Assertions:

4. URL contains /inventory
5. Products page is visible
6. Product grid displayed

- Test Case 2 — Invalid Login

Steps:

7. Enter locked_out_user/secret_sauce
8. Click Login

Assertions:

9. Error banner displayed with locked out message

- Test Case 3 — Add Multiple Items to Cart

Steps:

10. Add 2 products from Products page
11. Click cart icon

Assertions:

12. Cart badge = 2
13. Both items listed correctly with prices

- Test Case 4 — Checkout: Your Information

Steps:

14. Click Checkout in cart
15. Fill name & postal code
16. Click Continue

Assertions:

17. Overview page displayed with items and totals

- Test Case 5 — Finish Checkout

Steps:

18. On Overview, check totals

19. Click Finish

Assertions:

20. Checkout: Complete! confirmation message shown

- (Optional) Test Case 6 — Sorting

Steps:

21. Use dropdown to sort by price low→high

Assertions:

22. Products reorder accordingly

B2. API Automation — JSONPlaceholder

Target: <https://jsonplaceholder.typicode.com/>

Use any framework: Postman/Newman, REST Assured, SuperTest, pytest+requests, etc.

Note: This is a fake API; some operations succeed without persistence.

- API Test 1 — GET /posts

Steps:

23. Send GET /posts

24. Verify status = 200

25. Verify 100 items

26. Validate schema

Assertions:

27. All checks pass

- API Test 2 — GET /users/{id}

Steps:

28. Send GET /users/1
29. Verify status = 200
30. Check email format, address fields

Assertions:

31. Valid response, no nulls

- API Test 3 — POST /posts

Steps:

32. Send POST with {title, body, userId}
33. Verify status = 201

Assertions:

34. Response includes sent fields and id

- API Test 4 — PUT & PATCH /posts/{id}

Steps:

35. Send PUT /posts/1
36. Send PATCH /posts/1

Assertions:

37. Response reflects changes, status = 200

- API Test 5 — DELETE /posts/{id}

Steps:

38. Send DELETE /posts/1

Assertions:

39. Status = 200 or 204, response valid

Also include some negative tests (invalid IDs, missing fields, wrong headers).

Part C — Unit Testing

Write unit tests for at least one utility function in your chosen language (JS/TS, Python, or Java). Cover valid inputs, invalid inputs, and edge cases.

Example functions: isValidEmail, calculateDiscount, formatCurrency

Part D — End-to-End (E2E) Flow (SauceDemo)

Scenario: Complete Purchase Flow

40. Login as standard_user/secret_sauce
41. Add 2 items from Products
42. Go to cart → Checkout
43. Fill checkout info → Continue
44. Verify overview totals → Finish
45. Confirm Checkout: Complete! → Back Home

Include Page Objects, assertions, test data, and a README with setup/run instructions.

Submission Guidelines

- Everything must be in a single GitHub repository.
- ManualTesting.md at repo root.
- /ui-tests, /api-tests, /unit-tests, /e2e directories.
- README.md with setup/run instructions.
- Additional docs/screenshots if needed.

Evaluation Criteria

Manual Testing (Part A)	25%
UI Automation (Part B1)	25%
API Automation (Part B2)	15%
Unit Testing (Part C)	15%
E2E Test (Part D)	15%
Documentation & Clarity	5%