I. Assumptions

1. Tests are executed in an environment with a properly configured Node.js and Playwright setup. It is assumed that the development or CI environment has access to node_modules, required browsers (chromium, firefox, webkit), and tools (npm, npx).

2. The test environment consistently includes data-test attributes, e.g. data-test="username". Most selectors use getByTestId, which assumes that DOM elements are marked with unique and stable data-test attributes.

3. Tests are designed to be independent and isolated. Each test (or test group in *.spec.ts files) runs independently without shared runtime data or requiring a specific execution order.

4. The Page Object Model (POM) structure is consistent and complete. Each UI section has an assigned class in the pages/ directory containing methods to interact with that module (e.g., LoginPage, ProductPage).

5. Test data is generated dynamically using the faker library. Variable test content (e.g., users, addresses, emails) is dynamically created with @faker-js/faker, reducing the risk of collisions and eliminating the need to maintain manual test data.

6. The user should have a local .env file with required environment variables. The framework assumes that values such as BASE_URL, USERNAME, or other test configurations are stored in .env and loaded automatically

II. Limitations

1. There is no user data cleanup mechanism after tests. After a test is executed, data created by the user (e.g., items in the cart) is not removed. This lack of cleanup can lead to test environment pollution and may affect the outcome of subsequent tests.

2. Authentication is handled exclusively via the UI. Currently, user login is performed through the user interface, which increases test duration. Logging in via API would significantly speed up and stabilize the test execution.

3. Tests do not cover users with different roles.The framework does not validate system behavior for various user roles (e.g., admin, manager), which limits test coverage in role-specific scenarios.