Problem Statement: Automate an E-Commerce Web Application - Insist

Pseudocode:

1. Setup and Configuration:

Install Eclipse IDE and set up a new Maven project.

Add the required dependencies for Selenium and TestNG in the pom.xml file.

2. Create TestNG Test Classes:

Create a new TestNG test class for each module of the e-commerce application (e.g., Login, Product Search, Cart, Checkout, etc.).

3. Test Setup:

Use TestNG annotations like `@BeforeSuite`, `@BeforeClass`, or `@BeforeMethod` to set up the WebDriver, open the browser, and navigate to the e-commerce application URL.

4. Test Cases:

Define individual test cases within each TestNG test class.

For each test case:

Use Selenium WebDriver methods to interact with the web elements (click buttons, fill forms, etc.).

Implement assertions using TestNG's `Assert` class to verify expected results.

Handle any exceptions or errors that may occur during test execution.

5. Test Teardown:

Use TestNG annotations like `@AfterSuite`, `@AfterClass`, or `@AfterMethod` to clean up resources after test execution.

Close the browser and WebDriver instance.

6. TestNG Suite Configuration:

Create a TestNG XML file to configure test suites.

Include all the test classes in the XML file to be executed as part of the test suite.

7. Run the Tests:

Right-click on the TestNG XML file and select "Run As" > "TestNG Suite" to execute all the test cases.

8. Reporting and Logging:

Use TestNG's built-in reporting features to generate test execution reports.

Implement logging using log4j or similar libraries to capture relevant test execution information.

9. Continuous Integration:

Integrate the project with a CI/CD tool (e.g., Jenkins) to automate test execution and reporting on each code commit.

10. Maintain and Update Test Scripts:

Regularly update test scripts to accommodate changes in the e-commerce application's UI or functionality.

Execute regression tests after each update to ensure the application's stability.

11. Finalize and Deploy:

After thorough testing and successful automation, deploy the e-commerce web application.

12. Monitor and Improve:

Continuously monitor test results and performance to identify areas of improvement.

Actively involve in bug tracking and resolution processes to ensure the application's reliability.