

### Test Automation Framework Approach

Component	Details
Objective	To design a scalable test automation framework that ensures key features (user login, registration, product search, add to cart, checkout, and order history) are thoroughly tested.
Framework Type	Hybrid Framework combining Data-Driven and Keyword-Driven approaches. This allows reusability and easy maintenance of test scripts.
Tools	<b>Selenium WebDriver:</b> For browser-based UI automation. <b>Maven/Gradle:</b> For dependency management and running tests. <b>TestNG/JUnit:</b> For organizing and managing test cases. <b>ExtentReports/Allure:</b> For generating detailed test execution reports. <b>Git:</b> For version control of test scripts. <b>Jenkins:</b> For continuous integration and automated test execution.
Test Structure	<b>Page Object Model (POM):</b> Separates page elements and actions from test logic, enhancing maintainability and readability. <b>Test Data:</b> External sources like Excel or JSON files to provide input data for testing various scenarios. <b>Modular Design:</b> Breaks down test cases into reusable modules (e.g., login, search, checkout) to simplify updates and extensions.
Test Execution	<b>Parallel Execution:</b> Runs multiple tests simultaneously to speed up the test cycle. <b>Cross-Browser Testing:</b> Tests are executed across different browsers (e.g., Chrome, Firefox, Edge) to ensure compatibility. <b>Headless Mode:</b> Runs tests in the background without opening browser windows for faster performance during CI processes.
Key Features to Automate	<b>User Login:</b> Test various login scenarios with valid and invalid credentials. <b>Registration:</b> Validate the registration form functionality with different inputs. <b>Product Search:</b> Automate searches using keywords, filters, and categories. <b>Add to Cart:</b> Verify adding, updating, and removing items from the cart. <b>Checkout:</b> Automate the complete checkout process including payment methods and discount application. <b>Order History:</b> Ensure users can view their past orders, including details like order status and tracking information.
Test Reporting	Provides real-time reports showing test results (pass/fail) with screenshots for failures. Tracks key metrics such as test coverage, execution time, and defect density.
Maintenance	<b>Script Updates:</b> Modify test scripts in response to UI or business logic changes. <b>Version Control:</b> Manage test scripts with Git to track changes and collaborate. <b>Coverage Enhancement:</b> Regularly review and enhance test coverage based on execution results and new feature additions.