**Light & Wonder - Amazon shopping**

**Scope:**

The scope of “L&W Amazon shopping” project is to provide an automation testing framework which includes Page Object Model test scripts with BDD(Cucumber) approach using Java and Selenium. The outcome of the project is provided using Cucumber reports for validation of the test scenarios.

**Requirements:**

The “L&W Amazon shopping” project requirement is to test the below scenarios on Amazon website:

* Adding the first item of the search list for “Monitor” keyword to the cart and verify whether it is added to the cart and the product price is the same as the cart page price and the subtotal price.
* Adding the second item of the search list for “Laptop” keyword to the cart and verify whether it is added to the cart and the product price is the same as the cart page price and the subtotal price.
* Adding multiple items to the cart which is adding first item of search list for “Headphones“ keyword to the cart followed by adding first item of search list for “Keyboard“ keyword to the cart and whether it is added to the cart and the product price of each item is the same as the cart page price and the subtotal price is the sum of item1 of item2.

**Test Design:**

The design approach for this project involves the below components for a reusable, robust, efficient testing framework:

**Configuration:** Initiates the test suite with required beans/objects/property variables and also makes sure the environment is ready for execution.

**Feature:** Test scenarios listed as part of requirement will be added in the feature as a BDD language with each feature representing a specific functionality.

**Step Definitions:** For each step added in the feature file, the action method will be called in step definitions to adhere to the business logic. It is the intermediate between the feature and the action.

**Actions:** The business action defined in the test step is performed in the action package. It also helps in reusability and for better maintenance.

**Page Object Model:** All locators and methods related to respective pages are organized within dedicated page classes.

**Validations**: Validation of each page is done through assertions using TestNG

**Utils**: Includes common and reusable methods essential for the project's functionality.

**Reports**: Cucumber test reports represents the test results status, test execution time, test steps status and the error of each scenario in a html page.

**pom.xml:** Defines the project configuration, dependencies, build settings and other project-related information.

**Test Execution:**

**Prerequisite**:

* Download and Setup the below:
  + Java
  + Maven

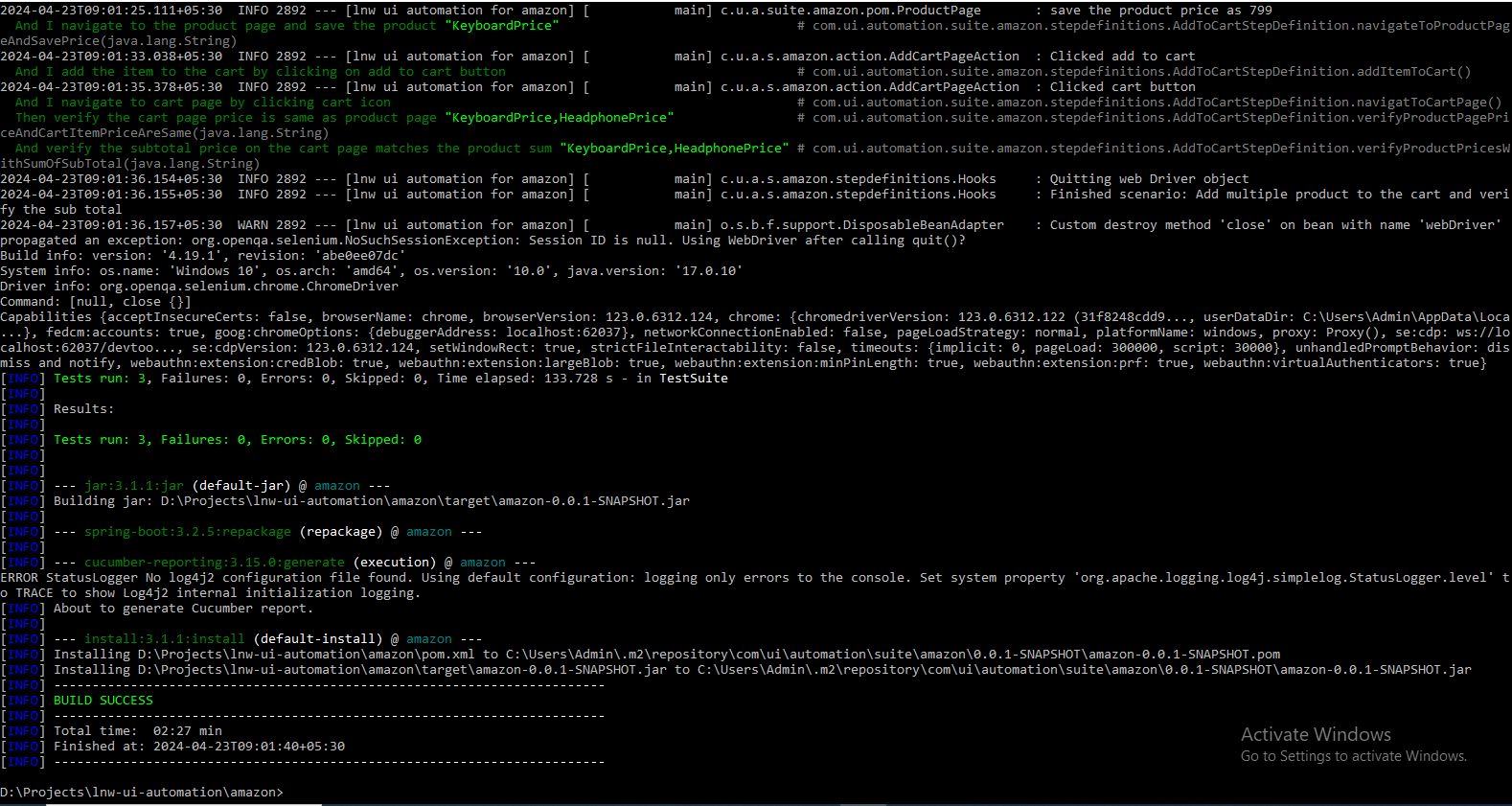
**Steps to execute:**

Project is committed in the GitHub repo - <https://github.com/GayathriUser/lnw-ui-automation>

1. Get a copy of the code from the above link by cloning the project into your local drive.
2. Open the project directory in your local drive and open a terminal.
3. Give the command: mvn clean install and press Enter.

## Test execution:

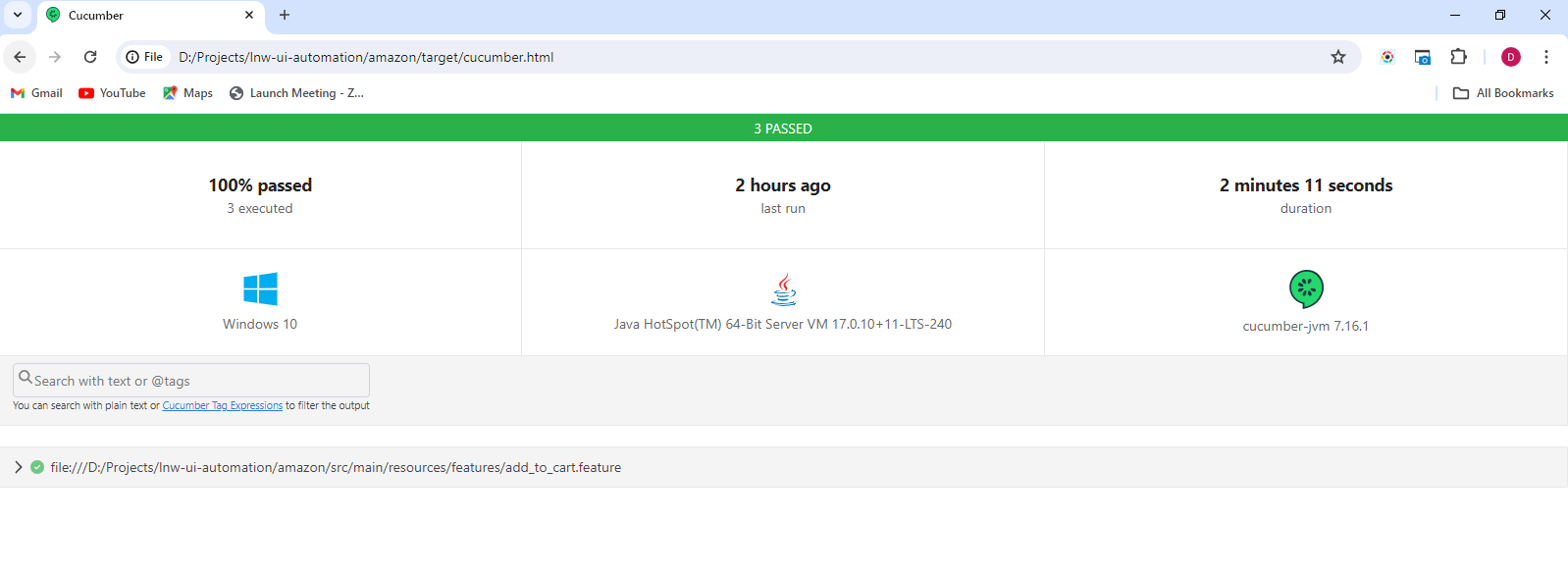
## 



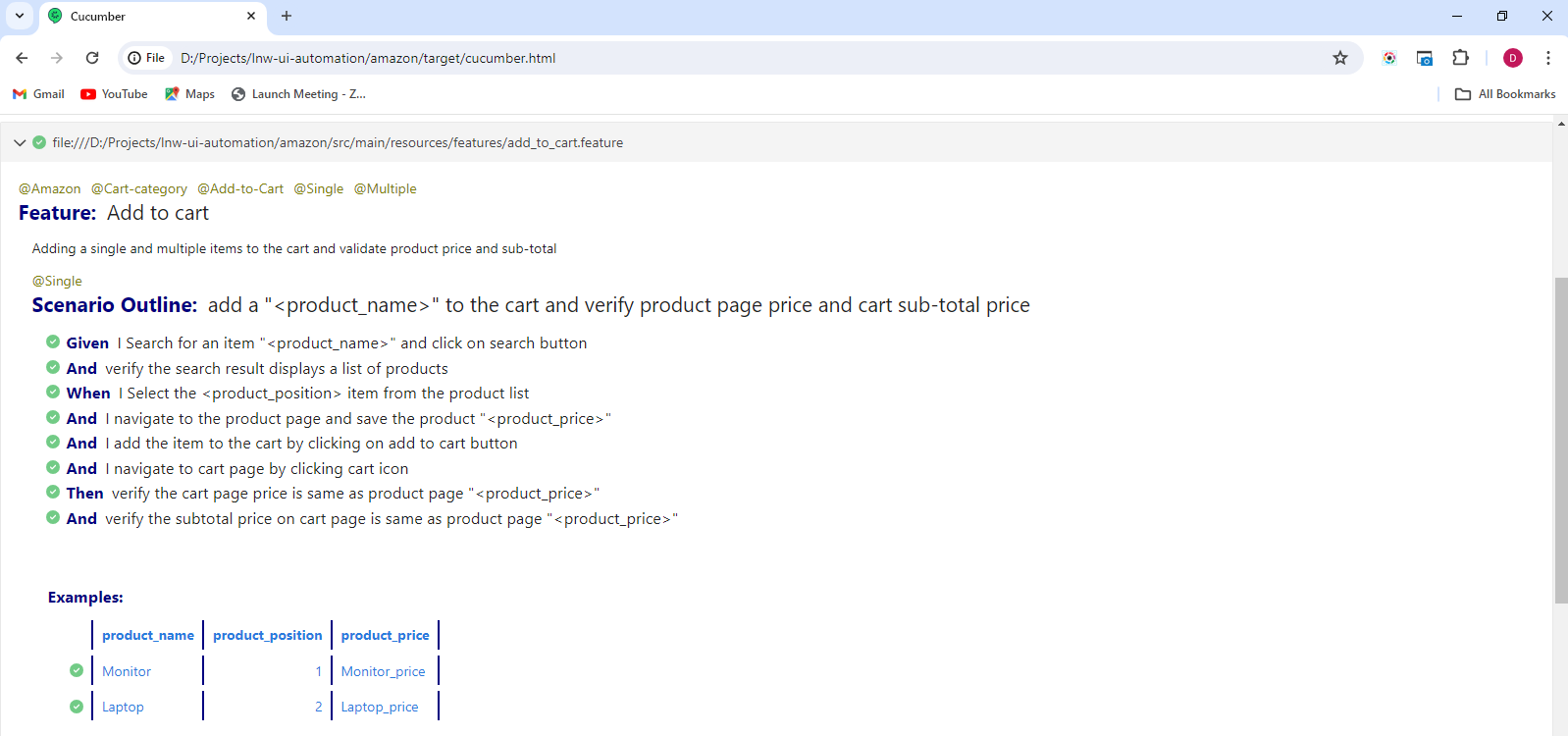
## Sample Test Report:

Cucumber TestReport screenshots below:

**Test results status and execution time:**



**Test Scenario 1 & 2:**



**Test Scenario 3:**

