

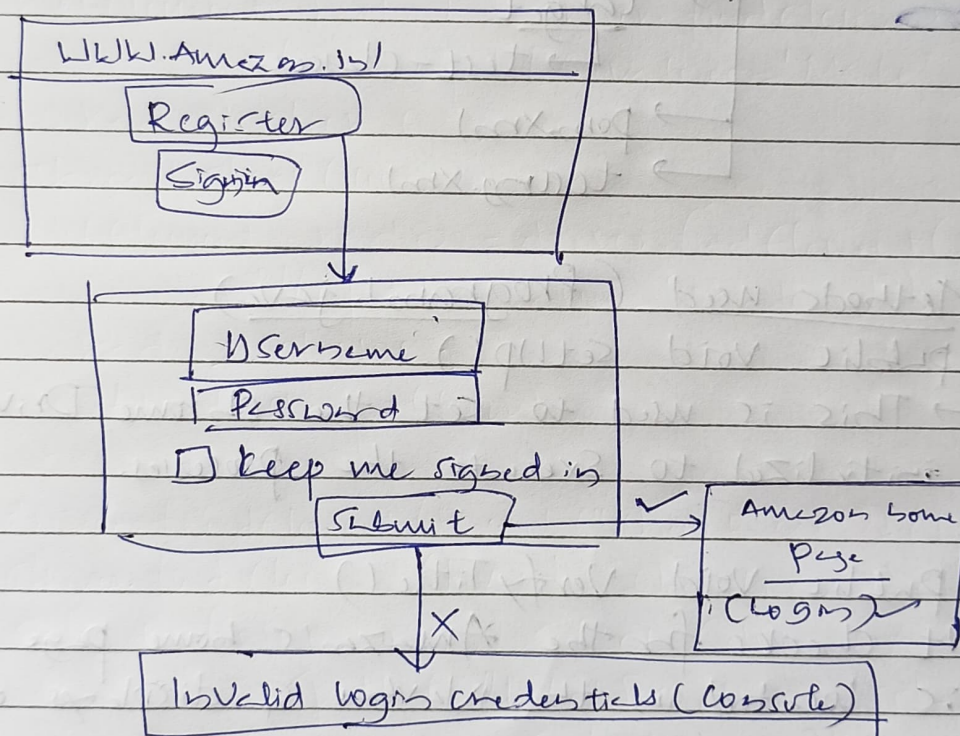
PRACTICE PROJECT

AUTOMATE WEB APPLICATION - (2.15)

Requirements

- 1) Registration form, login, search bar, add-to-cart
- 2) Handle invalid login & show appropriate error message
- 3) Fetch test reports over HTML & Excel
- 4) Document the steps

Steps :-

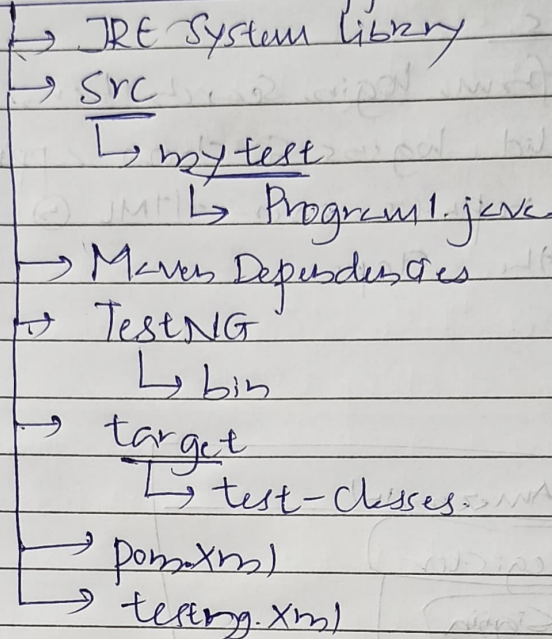


Processes involved

- + The Title of Amazon web page is checked, if match is found, then the value is returned
- + The logo is matched for the Amazon website, & the result is displayed in the console
- + Registration & login pages of Amazon validated.
- + Adding of a product to the cart is done

Project structure

Automation-Web-App



Methods used (Program1.java)

- 1) public void setup()
→ This is used to set the Chrome Driver path & initialized to execute the processes.
- 2) public void VerifyTitle()
→ It checks for the Amazon's home page title that is located in the Navber & displays output.
- 3) public void VerifyLogo()
→ It checks for the Amazon's home page logo that is located in the Navber & displays output.
- 4) public void VerifyRegistration()
→ This defines all the processes of automation to be tested for Registering a user for Amazon's web application (site).

5) ~~Void~~ Public Void VerifyLogin()

→ This involves in testing the functionalities of Amazon's login page.

6) Public Void VerifyZCert()

→ This Searches for a product in Amazon website locates it & adds the product to the cart.

All these methods are processed & automated using TestNG.

★ Pom.xml

→ This involves the modelVersion, groupId, artifactId and the dependencies used to process the project.

★ testng.xml

→ It is used for define name of suite, number of threads count.

output

Results of running suite

☒ All Tests

▼ ☒ Suite (S/O/O/O) (SS-266S)

▼ ☒ Test

▼ ☒ mytest.Program1

☒ VerifyLogo

☒ VerifyRegistration

☒ Verify Title

☒ Verify Y Login

☒ Verify Z Cert