SDET Capstone Project

Environment Setup Details:

Software to be Installed on your laptop/desktop

1. Open JDK 8 or higher version
2. Browsers (Chrome & Edge latest version)
3. Eclipse IDE latest version
4. TestNg and Cucumber plugins to be added to Eclipse IDE
5. Jenkins
6. Git

To raise a request, please log in to https://wiprocio.service-now.com/sp--> Create Request (Raise a Service request -> Click on Software request -> Select the catalogue as per your requirement [Commercial / Freeware]

General Instructions:

1. Write the solutions for each of the assignments in a separate maven project.
2. You can include the applicable and relevant user defined files as necessary to implement your solution.
3. Avoid hardcoding of the data, driver path, or the file path used for the screenshots or the file path used for the data driven testing
4. Avoid using Thread.sleep(), instead use different Selenium waits.
5. Avoid using absolute xpath if there are alternative ways of locating the element.
6. If driver needs to be setup for a browser, prefer using WebDriverManager, than manually downloading the driver and adding to the project.
7. When you write solution and run the test cases, capture output screenshots and submit it along with the solutions.
8. Follow the best practices while implementing the solution like proper formatting, indentation, usage of comments, meaningful variable and function names etc.,
9. Follow the Page Object Model design pattern
10. Schedule selenium test scripts in Jenkins to run at a specific time and every time your software changes and deploy the software to a new environment.[OPTIONAL]
11. Push your project into Wipro provided gitlab repository and share the URL of the repository.[OPTIONAL]

Use Case 1:

Create a maven project. Using Selenium WebDriver+TestNG+Page Object Model design pattern automate the following actions. Do the required validations and create the extent report with failure test logs and screenshots. [Use Chrome browser]

a. Go to page : https://magento.softwaretestingboard.com/
b. Create an account in Luma e-commerce application and Sign In by getting data from excel sheets[Use @DataProvider to make the script data driven]
c. After successful login, assert the welcome message visible on top right corner of the Home page. Take a screenshot of home page and save it in output screenshots folder
d. Click on sale menu→ tees on sale section to purchase the tees.
e. Filter the tees by Pattern, Climate and Material in Tees page and then add your favorite tees to the cart by choosing size and color
f. Click on view or edit the cart on top right corner to go to Shopping cart page
g. In Cart page click on water bottle reviews from More choices and assert the review comment and then click on Add to Cart button to add water bottle to the cart
h. Go to Shopping cart page and click on proceed to checkout

i. Select shipping address
j. Select and validate shipping method and then click on Next
k. In payment method page click on apply discount button by adding TEES20 discount code and validate the error message
l. Click on place order button to place the order, capture the screenshot of order confirmation, and then write order number and other details to a text file.
m. Go to MyAccount→My Orders page and validate the order details by clicking on View Order hyperlink
n. Click on Sign Out option on top right corner and validate the message after successful logout from Luma. Take a screenshot and save it in output screenshots folder.

Use Case 2:

Create a maven project. Using Cucumber+ Selenium WebDriver+TestNG+Page Object Model automate the following XYZ bank actions by writing valid scenarios. Do the required validations and create the extent report with failure test logs and screenshots. Make use of scenario outline, tags, background, and hooks wherever applicable. [Use Edge browser]

a. Go to page: https://www.globalsqa.com/angularJs-protractor/BankingProject/
b. Click on Customer Login button in XYZ Bank landing page and validate the login page url.
c. Select the name from drop down list in login page and click on Login button to validate the welcome message, account number and balance.
d. Click on Deposit button in home page, fill in input box with amount and click on deposit button to deposit the amount and validate the result message.
e. Validate deposit functionality with different set of inputs [Ex: negative, zero, positive numbers]
f. Click on Withdraw button in home page, fill in input box with amount to be withdrawn and click on withdraw button and validate the result message.
g. Validate withdraw functionality with different set of inputs [Ex: negative, zero, positive numbers]
h. Click on Transactions button in home page to see the transactions page and validate the URL, transaction details displayed in a table.
i. Click on logout button on top right corner and validate the login page url
j. Click on home button on top left corner to go back to landing page and click on Bank Manager Login button to go to Manager page and validate the url
k. Click on Add Customer button in manager page and fill the input elements by getting the data from excel sheet to create the customer and validate the customer id shown in alert box along with success message.
l. Validate Open Account functionality in manager page by filling up customer name and currency from drop down list.
m. Click on Customers button in manager page and validate data table with customer details. In addition, validate search input box functionality in manager page.
n. Click on delete button in data table to remove newly added customer. [Scroll the page vertically to see newly added customer]

Note: Execute the scenarios using @Dataprovider using the data from excel file.

Use Case 3:

Create a maven project to automate the REST API https://petstore.swagger.io/v2/ using BDD-Cucumber and Rest Assured Framework. Validate response status code, headers and body for every endpoint.

a. Automate Pet, Store and User endpoints (refer below screenshots) by writing valid BDD scenarios.
b. Use api-key=”special-key” as header for authorization
c. For implicit OAuth2 authorization use

i. client-id=”test”
ii. Username : test
iii. Password: abc123
iv. Scopes: Read & Write

Note: Please refer to https://petstore.swagger.io/#/ documentation for more details about the endpoints.

Petstore Rest API endpoints: