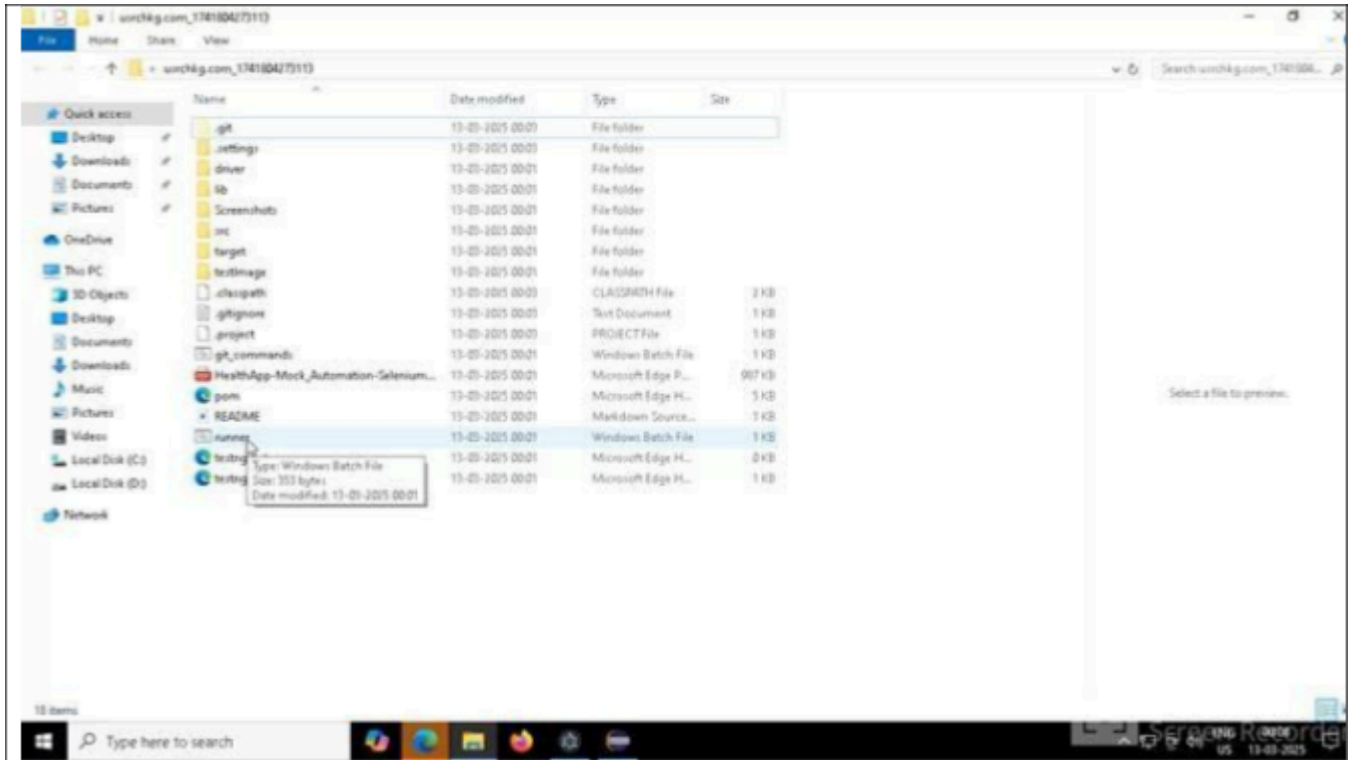


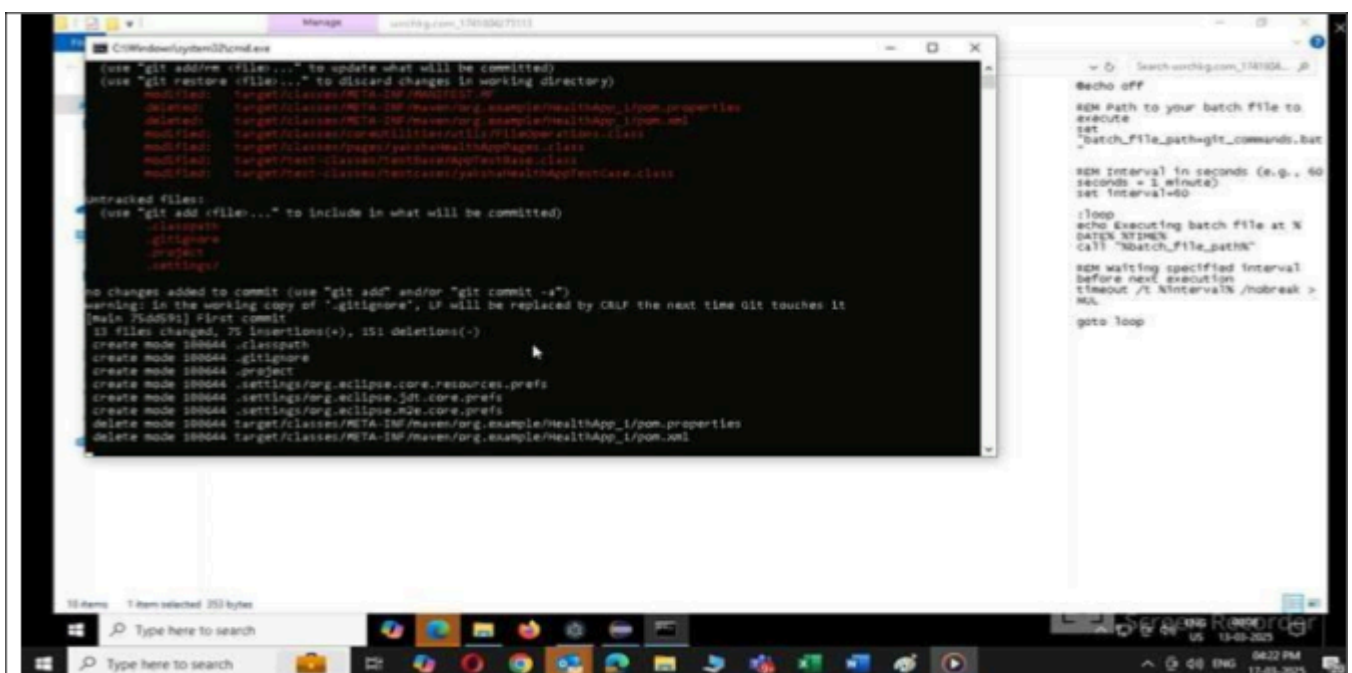
FORM AUTOMATION – PL2(9TCs)

Pre-requisite:

Before you start working on your project, execute the runner file present in your project folder (Simply by double click). **This is mandatory.**

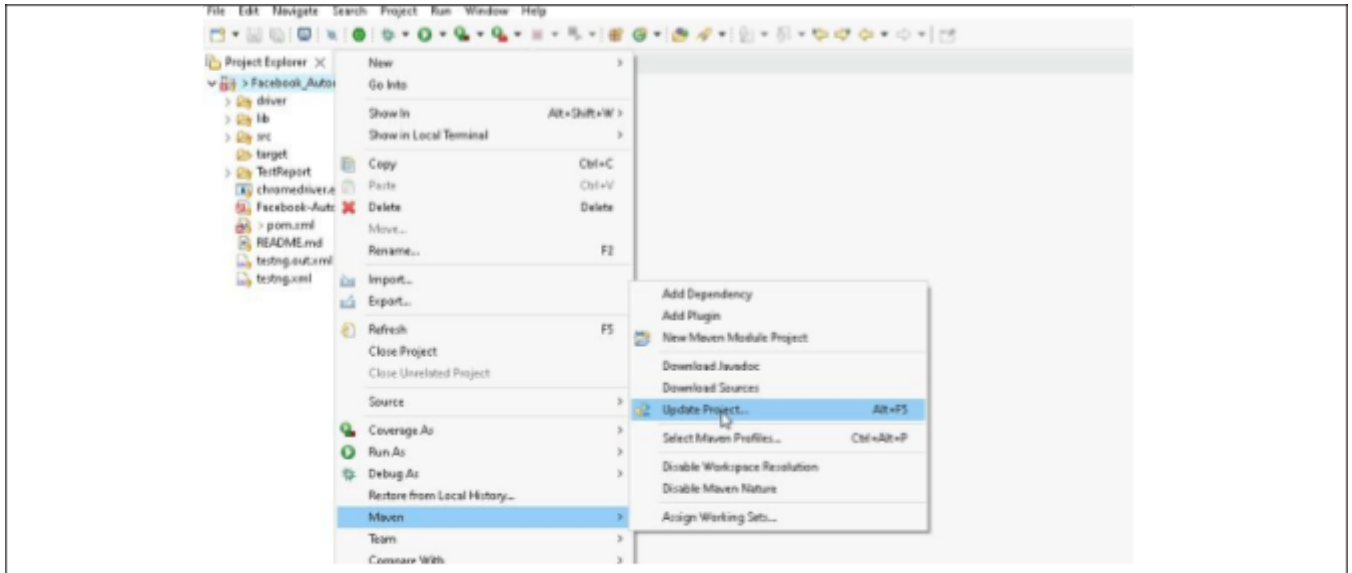


This will launch a command terminal for you where it will keep on pushing your updated code to GIT at regular intervals. Keep that command terminal open at the backend and you can continue working on your project.

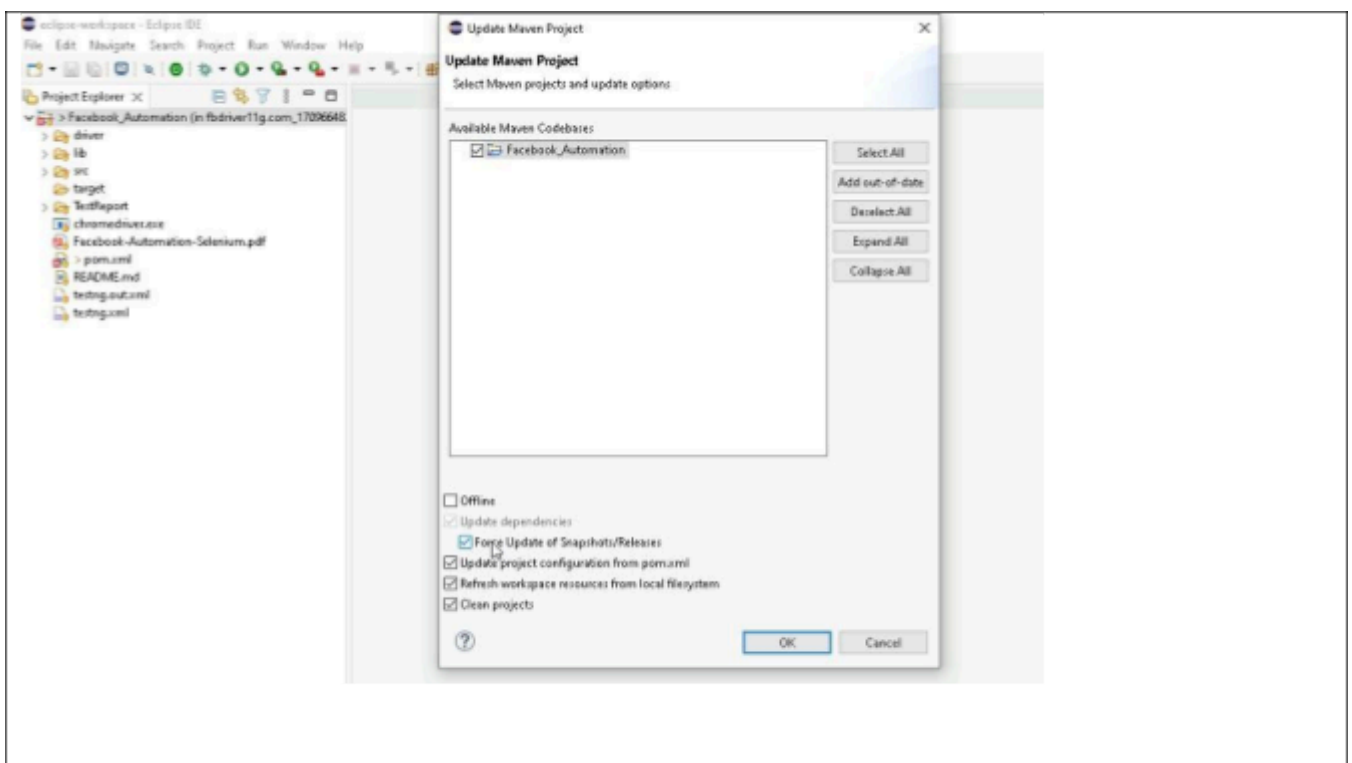


As soon as you import the project in Eclipse, update the project using maven update option as below. This is to resolve issue if any maven dependency not downloaded properly:

1. Right click on project: Go to “Maven” : Select “Update Project”



2. In Update Maven Project Box Select “Force Update of Snapshots/Releases” and click OK



Template Code Structure:

- Below are the packages and files you will be required to work on.
- Other Files and packages you can ignore.
- In other Files and packages do not make any changes. It would affect your evaluation.
- You are not required to work in the “Test” Folder. The files there are non-editable. Editing those files and trying to save them will throw errors and affect your evaluation.

| Package | Class/File | Description |
|-----------------------------------|------------------------------|--|
| src/main/java/coreUtilities/utis/ | FileOperations.java | <ol style="list-style-type: none"> 1. It contains methods to read data from Excel files. 2. The method is already implemented. |
| /src/main/java/pages | FormManagement_L1_Pages.java | <ol style="list-style-type: none"> 1. All core activities (mentioned in the list below “Key Activities to implement”) are to be performed here. 2. The comments associated with each templated method here describe the expectation. 3. You can define locators and xpath here. 4. Declare any variable/object you need to share data/status between different methods. 5. Do not modify the signature of methods declared here. 6. You can create additional supportive common methods in CommonEvents class. |
| /src/main/resources/ | Config.xlsx | URL to navigate: URL is defined here. |
| | expected_data.xlsx | Contains data to fill in the form. |
| /src/main/java/coreUtilities/utis | CommonEvents.java | <ol style="list-style-type: none"> 1. Contains all common activities. 2. Certain templated common method declared here. 3. You implement them as per your need. 4. You can add any additional method for common activity here |
| | Testng.xml | Execution needs to kick started from TestNG xml |

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

Need to automate the following activities using Selenium + Java.

Note: In this use case, you need to automate below scenarios using the “<https://demo.automationtesting.in/Register.html>” portal.

Key Activities to implement:

| # | Summary | Action | Expected Result |
|---|--|--|---|
| 1 | Validate the title of home page | Precondition : NA Steps: 1. Navigate to the URL : https://demo.automationtesting.in/Register.html | Verify that the title of the page should be as: "Register" |
| 2 | Click “Switch to” -> “Alert” Link and validate the title of the navigated page. | Precondition: User should be on the “Register” page. Steps: 1. Click on “Switch To” menu. 2. Click on “Alert” submenu. | Verify that the title of the page should be as: "Alert" |
| 3 | Click on button to display the alert box and validate its text. | Precondition: User should be on the “Alert” Page. Steps: 1. Click the button titled as “Click the button to display an alert box:” | Verify the text displayed by alert should be as "I am an alert box!" |
| 4 | Click on Register menu option to navigate to Register page and fill the form with data provided in “expected_data.json” file | Precondition: User should be on the “Alert” Page. Steps: 1. Click on “Register” menu. 2. Fetch and fill in the details for the following fields through the expected_data.xlsx file. (First Name, Last Name, Address, Email address, Phone). 3. Click on “male” radio button. 4. Click on “Cricket”, “Movies” and “Hockey” check boxes. | Verify that all the entered values are filled properly and required check boxes and radio buttons are selected. |

| | | | |
|---|---|---|--|
| 5 | Select countries from the country dropdown one by one (Bangladesh, Denmark, Hong Kong, Australia), and verifies that the dropdown functions properly. | Precondition: User should be on the "Register" page. Steps: <ol style="list-style-type: none"> 1. Refresh the page 2. Click on the country dropdown and Select Bangladesh 3. Click on the country dropdown and Select Denmark 4. Click on the country dropdown and Select Hong Kong 5. Click on the country dropdown and Select Australia | Verify that the country dropdown is successfully interacted with to select Bangladesh, Denmark, Hong Kong, and Australia, and remains and functional throughout the process. |
| 6 | Click on the country dropdown and select Australia. | Precondition: User should be on the "Register" page. Steps: <ol style="list-style-type: none"> 1. Click on the country dropdown and Select Australia. | Verify that "Australia" is selected. |
| 7 | Check and uncheck each hobby checkbox and validate that the checkboxes are responding correctly. | Precondition: User should be on the "Register" page. Steps: <ol style="list-style-type: none"> 1. Click on the Cricket checkbox. 2. Uncheck the Cricket checkbox. 3. Click on the Movies checkbox. 4. Uncheck the Movies checkbox. 5. Click on the Hockey checkbox. 6. Uncheck the Hockey checkbox. 7. Click on the Cricket checkbox again. 8. Click on the Movies checkbox again. 9. Click on the Hockey checkbox again. | Toggle the cricket, movies, and hockey checkboxes multiple times (checking and unchecking). At the end verify that the checkboxes remain selected at the end of the test. |
| 8 | Select each radio button option for gender and validate that only the female radio button option is selected finally. | Precondition: User should be on the "Register" page. Steps: <ol style="list-style-type: none"> 1. Click the male radio button to select it. 2. Click the female radio button to select it | Verify that the female radio button is selected. |
| 9 | The method selects a year, month, and date from their respective dropdowns and verifies that all dropdowns are displayed after the selections. | Precondition: The User should be on the "Register" page. Steps: <ol style="list-style-type: none"> 1. Locate the Year dropdown and select the year "1996" from it. 2. Locate the Month dropdown and select the month "June" from it. | Return true if all dropdowns are displayed and function properly, otherwise return false. |

| | | | |
|--|--|---|--|
| | | 3. Locate the Date dropdown and select the date "25" from it. | |
|--|--|---|--|

NOTE: "Please do not delete any file in the src folder. But you are free to add any other file".

Expectations:

- 1) Learners should write automation script using Java and selenium to automate all the steps in the above question. In other words, automation script should perform all mentioned steps.
- 2) Learners should not use any tools to create the xpath. They should develop the xpath/cssselector on their own.

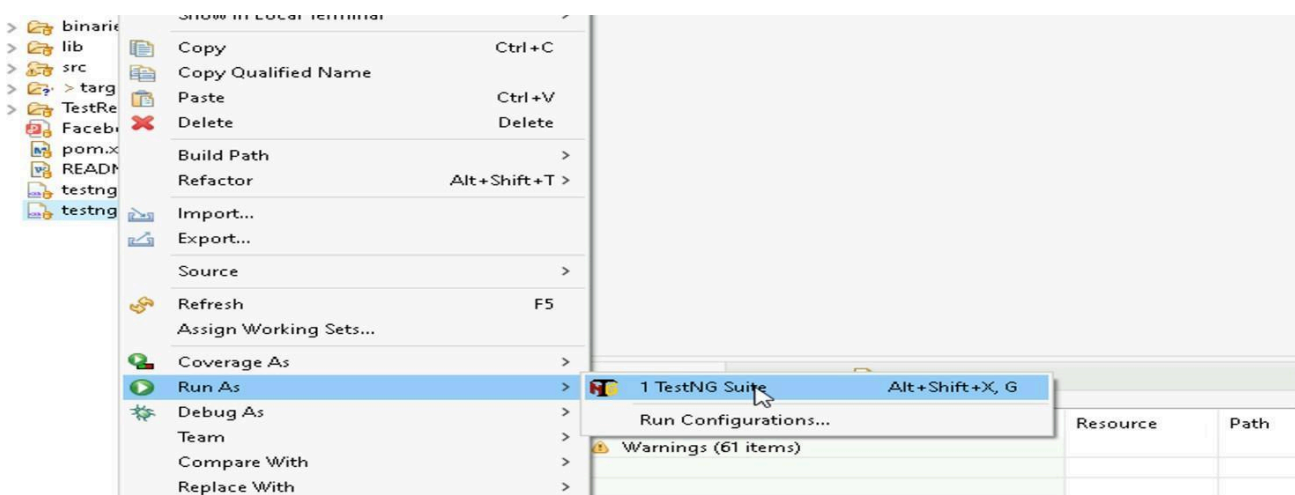
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

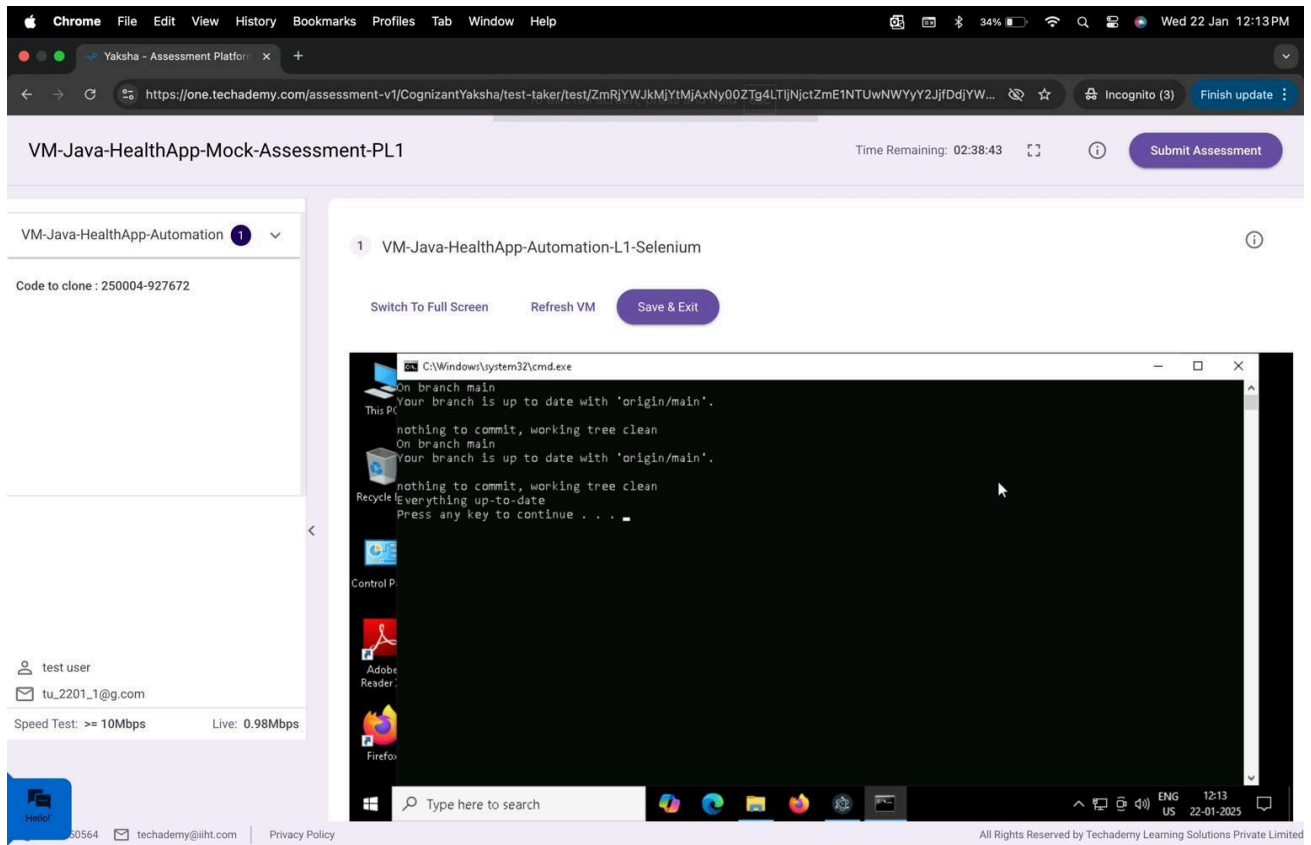
EXECUTION STEPS TO FOLLOW

1. You are mandatory required to run test cases for applications before final submission. Without this project, evaluation will not happen.
2. You can launch test cases any time as follows: Right-click on testng.xml and run TestNGSuite.



3. To do the final submission of the assessment :

- Press escape to come out of Fullscreen mode.
- Submit the assessment.



After the successful submission of the assessment, you will get a confirmation message displayed on your screen.

All the Best