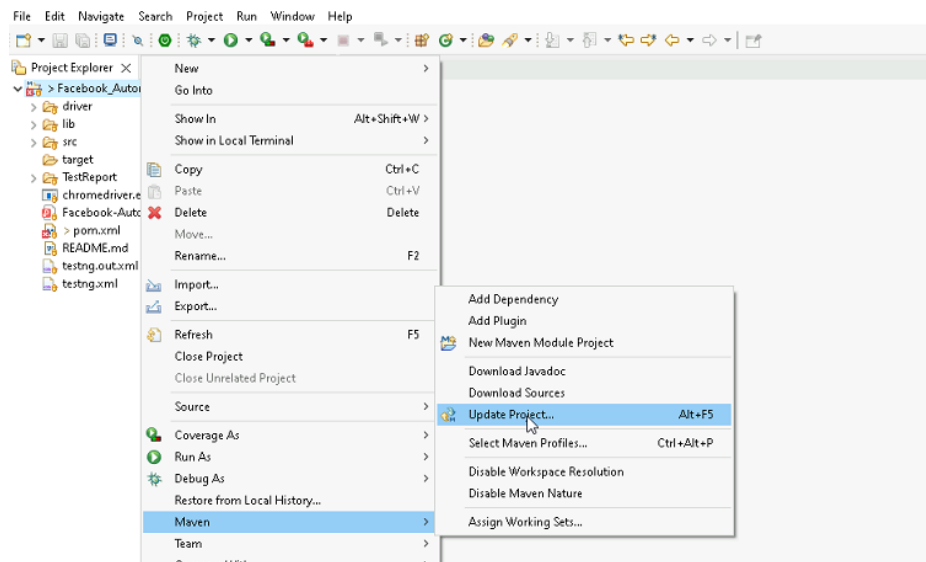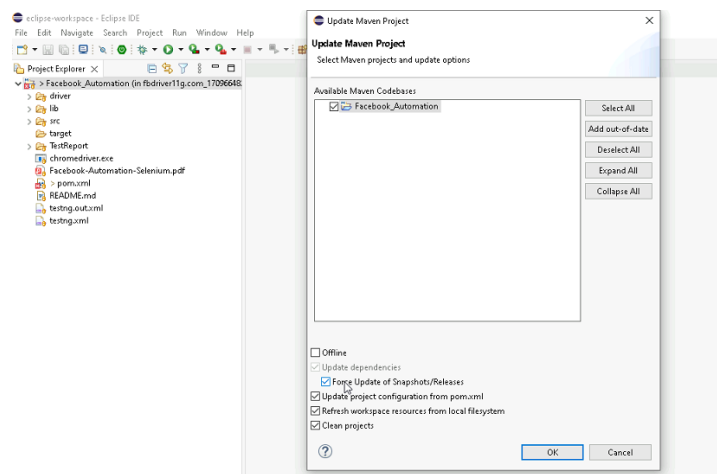# HEALTHAPP AUTOMATION DISPENSARY MODULE -PL1

Pre-requisite:

As soon as you import 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" and select "Update Project"

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

Template Code Structure:

a. Below are the packages and files you will be required to work upon.

b. Other Files and packages you can ignore.

c. In other Files and packages do not do any changes. It would affect your evaluation.

d. You are not required to work in "Test" Folder. Files there are non-editable. Editing those files and trying to save them will throw error and would affect your evaluation.

Package	Class/File	Description
src/main/java/coreUtilities/utils/	FileOperations.java	**The method for reading data as input from an Excel file have already been implemented here.** **This method is used to fetch the required data from excel including the URL for navigation.**
/src/main/java/pages	Dispensary_Pages.java	1. All core activities (mentioned in list above) 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 is Already defined here
	expected_data.xlsx	Contains data to fill in form
/src/main/java/coreUtilities/utils	CommonEvents.java	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:

Sl No.	Summary	Action	Expected Result
1	Verify the title and URL of the current page.	1. Go to URL : **https://healthapp.yaksha.com/** 2. Login as valid credential **(username : admin , password : pass123)** and click on "**Sign in**" Button 3. Get the title and URL of the Home page, post login 4. Validate the title and URL of the Home page	Title should be : **DanpheHealth** URL should be : **https://healthapp.yaksha.com/Home/Index#/**
2	Verify Presence of Dispensary Module.	**Precondition :** The User must be logged in to the HealthApp System. **Test Steps (for navigation purpose only) :** 1. Navigate to the Left Navigation Panel. 2. Check for the Dispensary Module. Verify Presence of Module.	dispensary Module should be present
3	Verify Presence of Sub-Modules under Dispensary Module.	**Precondition :** The User must be logged in to the HealthApp System. **Test Steps (for navigation purpose only) :** 1. Locate the "Dispensary" module in the left navigation panel and click the expand icon to view its contents. 2. Look for the following sub-modules within the "Dispensary" module: - Prescription - Sale - Stock - Counter - Reports - Patient Consumption 3. Ensure all listed sub-modules are present.	All mentioned sub-modules should be present under Dispensary Module.

4	Verify the presence of the following elements at the bottom of the "Sale" page: - Print Invoice Button - Discard Button - Invoice History - Credit Limits and Balances	**Precondition:** "Counter" sub module of "**Dispensary**" modules should be pre-selected. **Test Steps (for navigation and verify the presence of elements only) :** 1. Click on "Morning Counter" link text 2. It should navigate to the "Sale" page. 3. Scroll to the bottom of the page. 5. Verify the presence of the following elements at the bottom of the "Sale" page: - Print Invoice Button - Discard Button - Invoice History - Credit Limits and Balances	user should be able to scroll the bottom of the "**Sale**" page and able to see the following elements. - Print Invoice Button - Discard Button - Invoice History - Credit Limits and Balances
5	Perform the keyboard operation "Alt+N" to open the "**Add New Patient**" popup form and verify its title.	**Precondition:** "**Sale**" page must be pre-selected. **Test Steps (to verify the keyboard action) :** 1. Press **"Alt + N"** from the keyboard, 2. Then "A**dd New Patient**" form should popup. 3. Verify the title of the form is "**Add New Patient**" displayed over the top of the form.	**Add new patient** form should be open. Page title must be **"Add New Patient"**.
6	Validate the error message shown by "First Name" textfield after clicking the "**Ok"** Button without filling any information in the form.	**Precondition:** Ensure "Add New Patient" Form is Open and ready for inputs. **Test Steps (to verify the error message only) :** 1. Attempt to submit the form by clicking the "OK" button without entering any information in the form fields. 2. Check for an error message that should appear under the "First Name" textfield. 3. Confirm that the error message is appropriate (e.g., "First Name is required") and clearly visible under the "First Name" textfield.	Error message should be : **First Name is required.**
7	Fill all the text fields which are present inside the "Add New Patient" form. Check each field to ensure the information entered is accurate and correctly formatted.	**Precondition:** Ensure "Add New Patient" Form is Open and ready for inputs. **Test Steps (for the validation of entered data) :** 1. Enter Data in the First Name Field (Input a valid first name into the 'First Name' field.) 2. Enter Data in the Middle Name Field(Input a middle name (if applicable) into the 'Middle Name' field.) 3. Enter Data in the Last Name Field (Input a valid last name into the 'Last Name' field.) 4. Enter Data in the Age Field (Input the patient's age in the 'Age' field. Ensure the age is numerically valid.) 5. Enter Data in the Contact Number Field (Input a valid contact number into the 'Contact Number' field, following the format required (e.g., country code, area code)). This structured format ensures that all necessary personal details are correctly filled into the "Add New Patient" form.	Given text fields value inside the "Add New Patient" form are filled with appropriate information. The entered data should be correctly displayed in First Name, Middle Name, Last Name, Age and Contact Number Field on "**Add New Patient"** form.

| 8 | On the "**New Consumption Entry**'s page, verify the confirmation Message that is "**Are you sure you want to Proceed ?**" . | **Precondition:** User is logged in and "Add New Patient" Form is Open.

Test Steps (for the validation of popup message) :
1. Click on the cross "X" of Add New Patient popup.
2. Click on the "**Patient Consumption**" sub modules of Dispensary module.
3. Click on "**New Consumption**" button, then "New Consumption Entry" form will popup.
3. Directly click on "Save Consumption" button without fill any information.
4. Then a confirmation message will popup (**Message : "Are you sure you want to Proceed ?").**
5. Validate that confirm message.
6. Then click on confirm button.
7. Then close the "New Consumption Entry's page. | After the certain steps defined, Confirmation message displayed should be "**Are you sure you want to Proceed ?".** |

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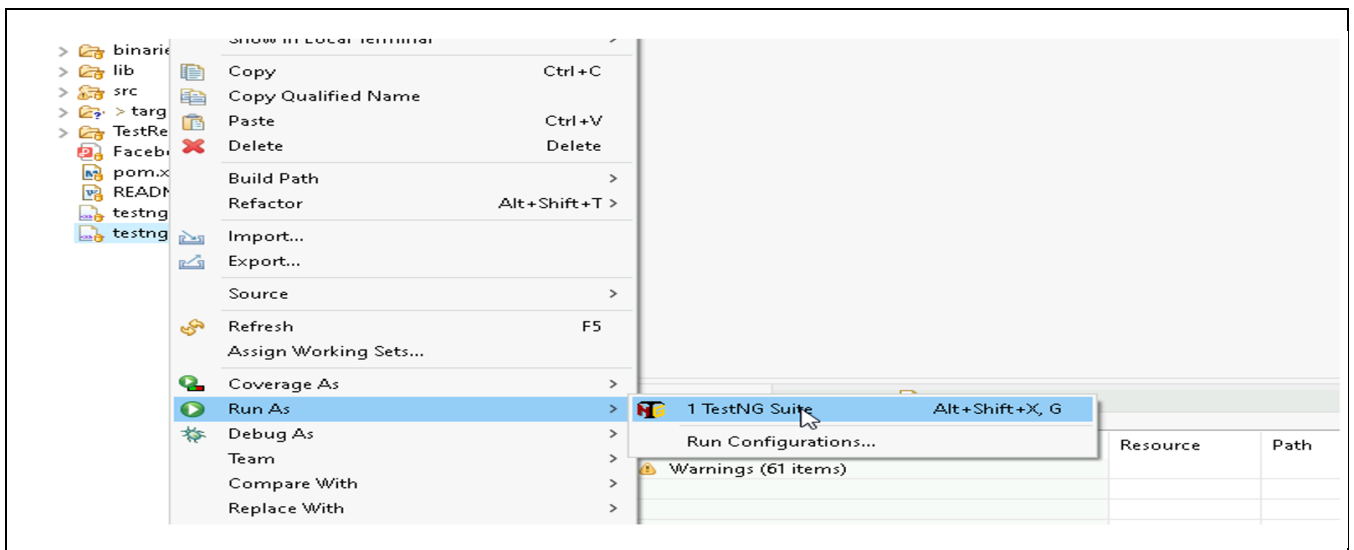2) Learners should not use any tools to create the xpath. They should develop the xpath/cssselector on their own.

IMPLEMENTATION/FUNCTIONAL REQUIREMENT
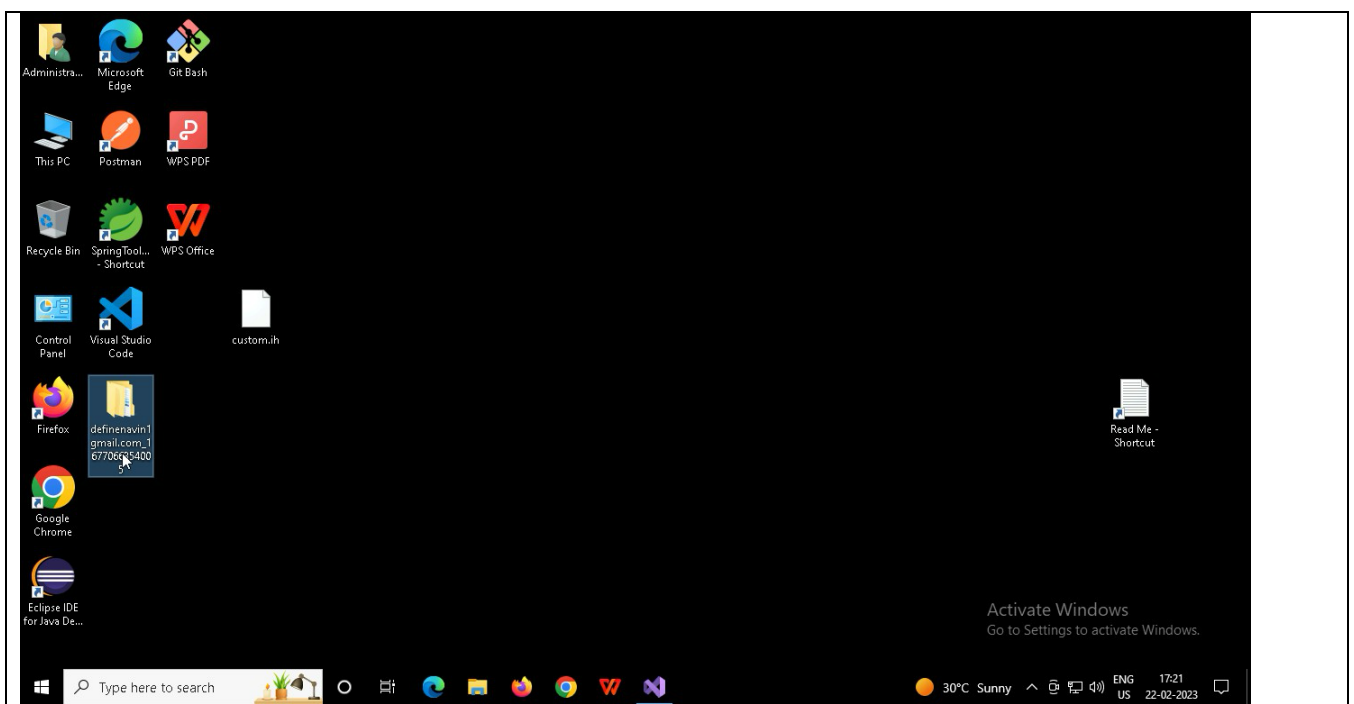
1.1   CODE QUALITY/OPTIMIZATIONS

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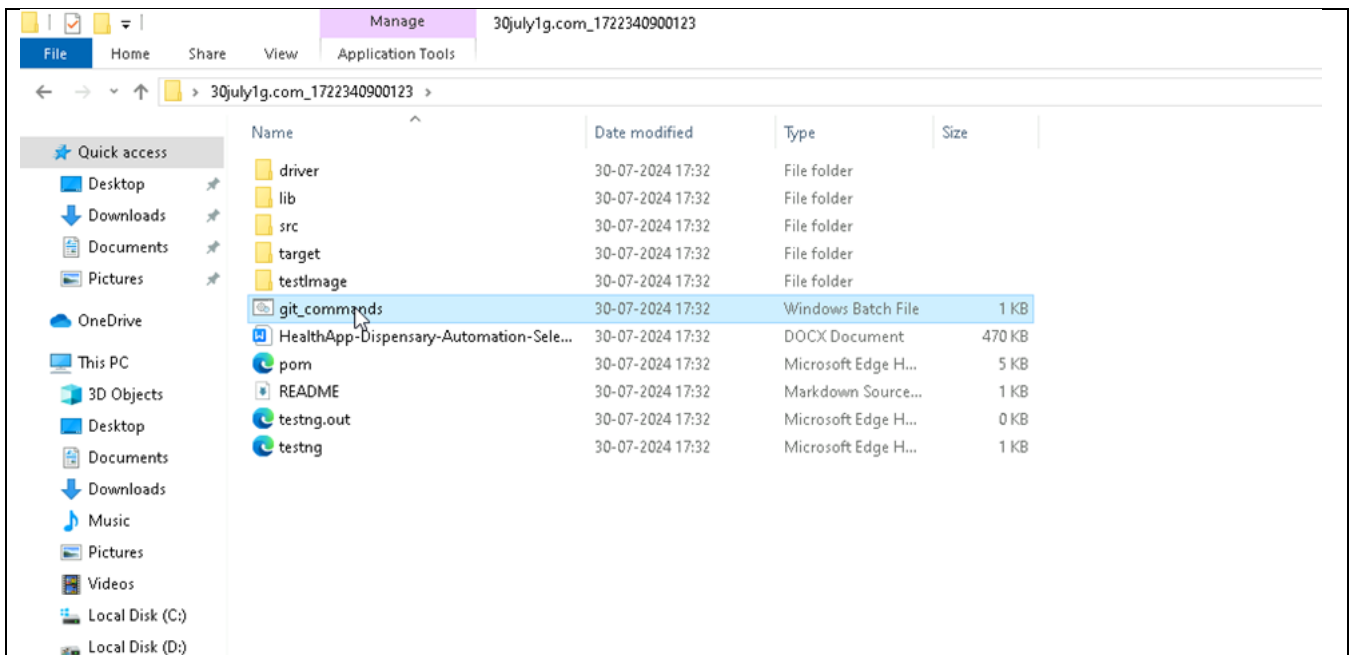
EXECUTION STEPS TO FOLLOW

1. **You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.**

2. **You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite**
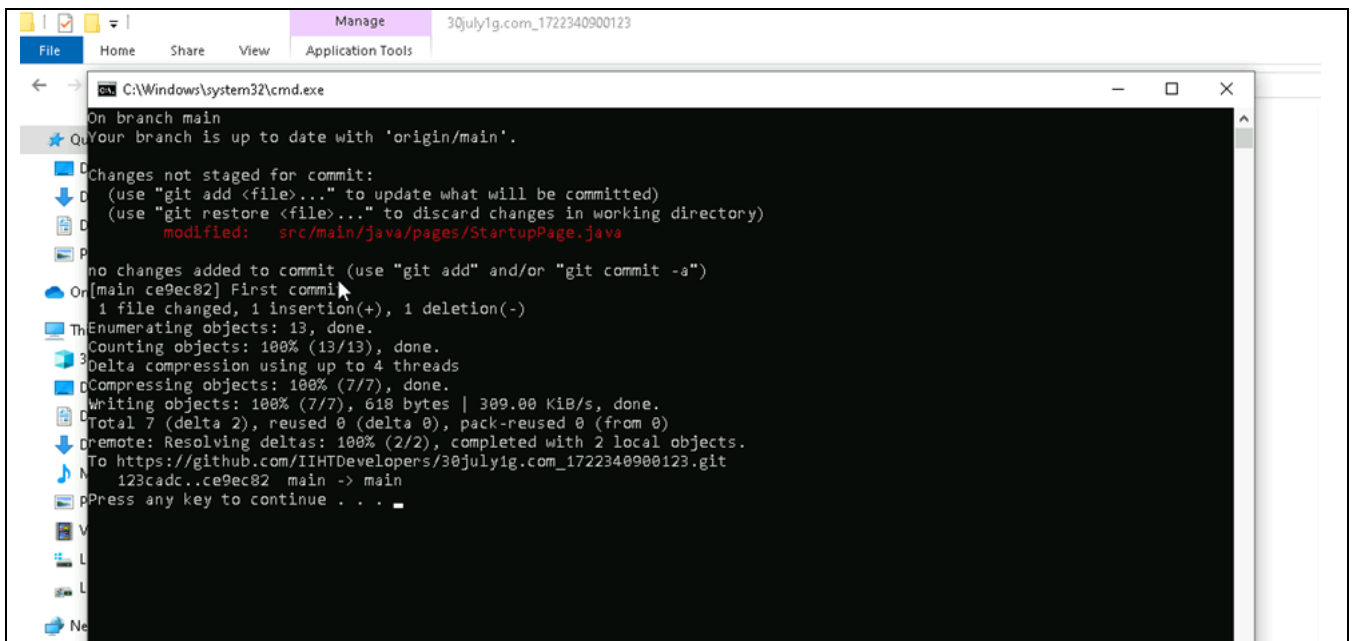
3. Before final submission, you are also required to push your code to GIT. Following are the steps to follow:

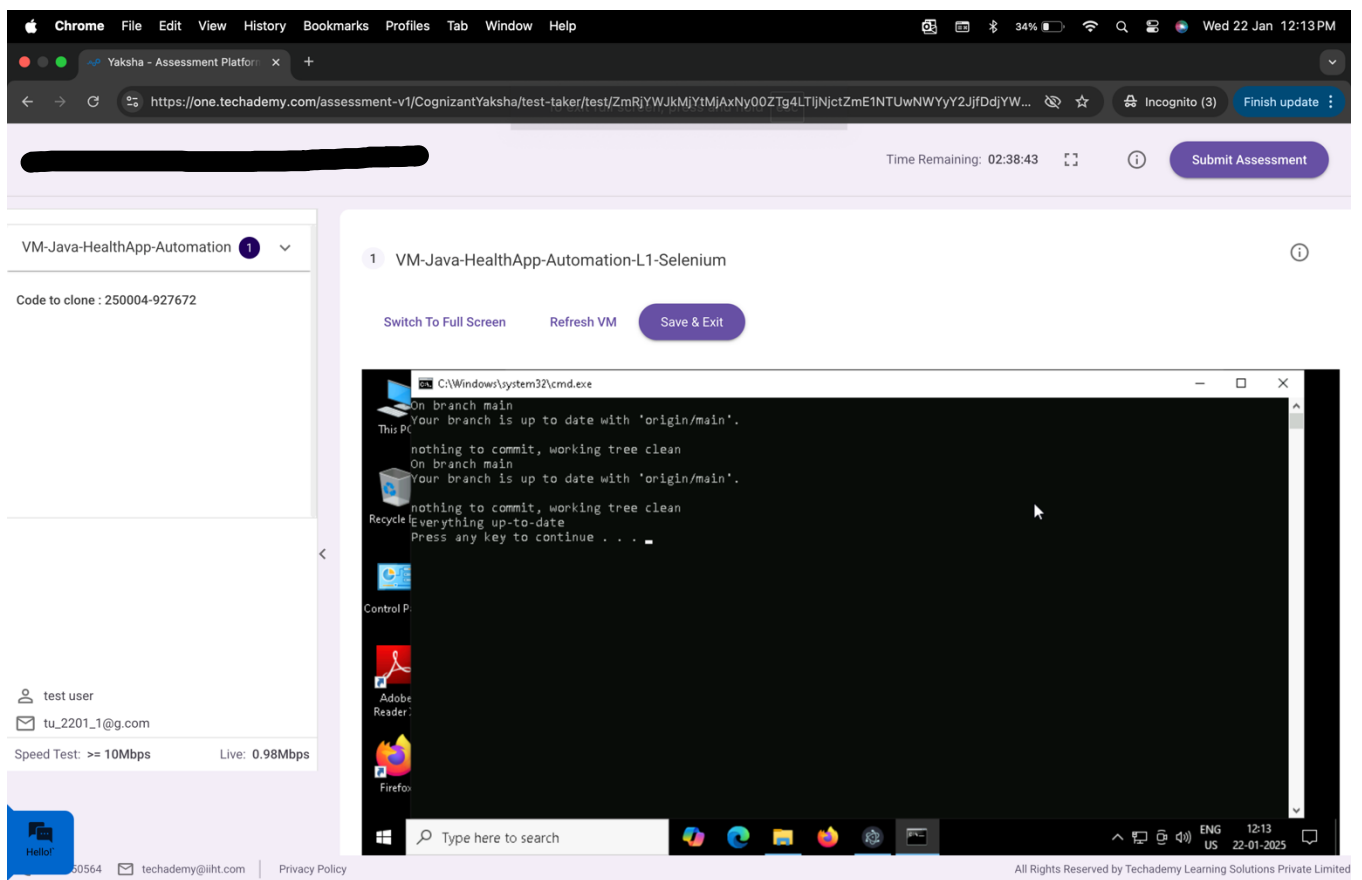In your project folder, you will find a batch file named git_commands

Double-click the batch file to run it. It will run the commands to push your code to GIT.

Once the code is pushed to git, you can go for 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