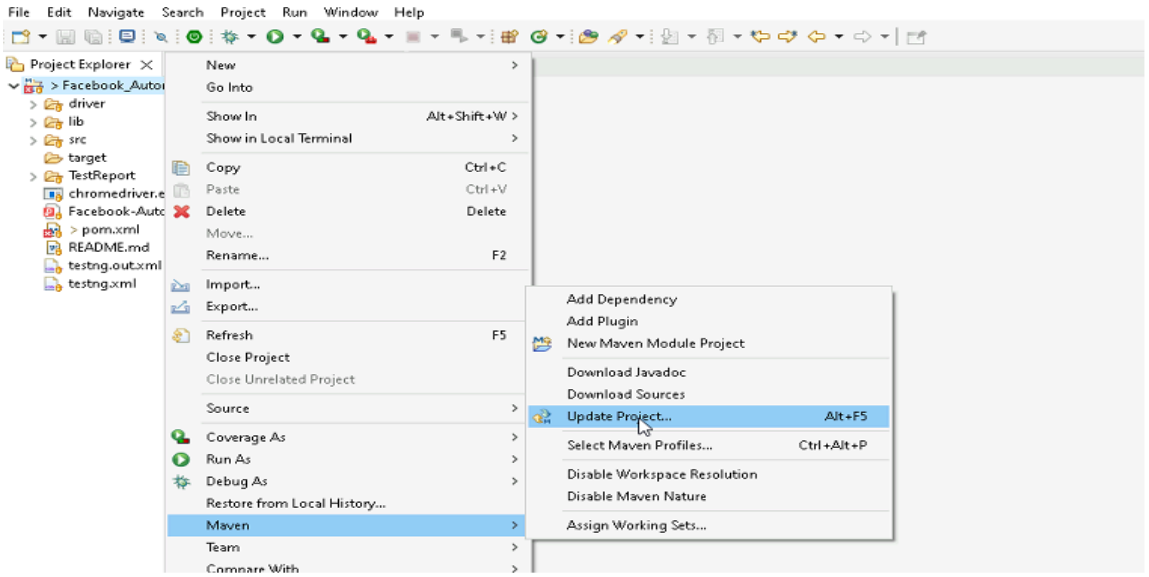
HealthApp Automation verification Module - PL1

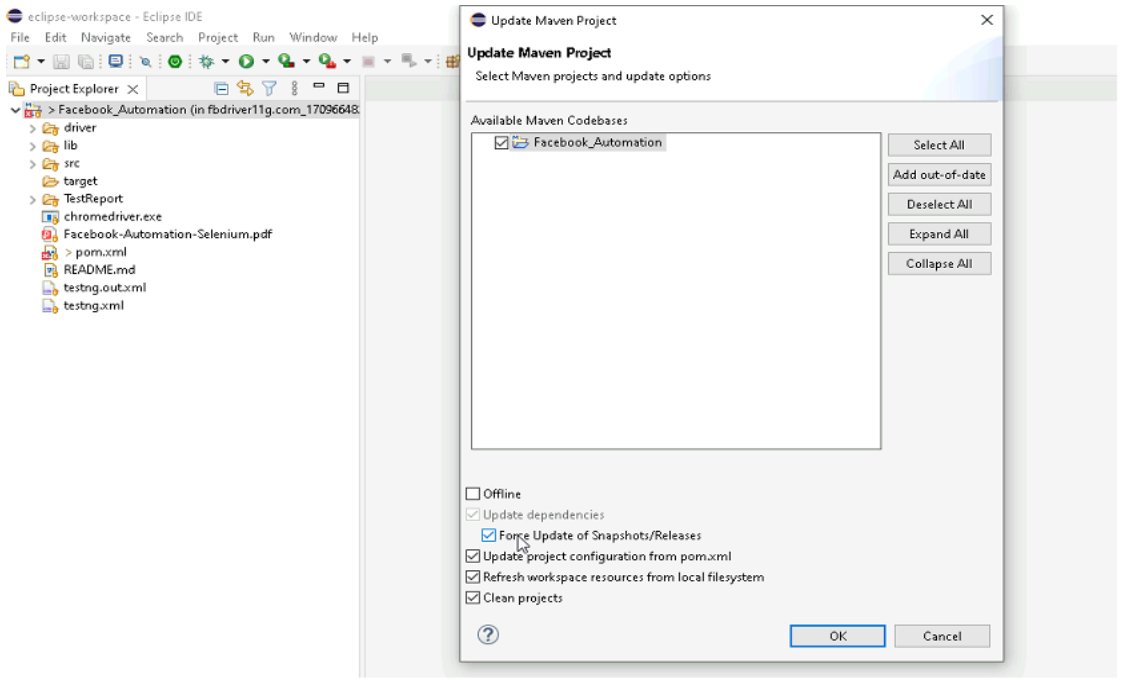
Pre-requisite:

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

1. Right-click on the project: Go to “Maven” and select “Update Project”



2. In the 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 on.

b. Other Files and packages you can ignore.

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

d. You are not required to work in the “Test” Folder. Files there are non-editable. Editing those files and trying to save them will throw errors 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 has already been implemented here.** **This method is used to fetch the required data from Excel including the URL for navigation.**
/src/main/java/pages	verification_page.java	1. All core activities (mentioned in the list above) 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 are Already defined here
	expected_data.xlsx	Contains data to fill in the form.

/src/main/java/coreUtilities/utils	CommonEvents.java	1. Contains all common activities. 2. Certain templated common methods are declared here. 3. You implement them as per your needs. 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:

#	Summary	Action	Expected Result
1	Verify whether the verification module is present or not	1. go to the URL: **https://healthapp.yaksha.com/** 2. login as a valid credential **(username: admin, password: pass123)** and click on the "**SignIn**" Button 3. Scroll down the menu till verification 4. Click on the verification	Verify the user is logged in successfully. Verify the URL of the current page should be as: "https://healthapp.yaksha.com/Home/Index#/Verification/Inventory/Requisition "
2	Ensure that all designated sub-modules are displayed in the left panel.	**Preconditions**: The user should be logged in and it is on the verification module. Make sure the user is on the verification->Inventory -> Requisition module.	Verify that sub-modules(Inventory, Pharmacy) should be displayed above the “Requistion” Tab.
3	Ensure the presence of the mentioned fields in the expected result.	**Preconditions**: The user must be logged into the health system. Make sure the user is on the verification->Inventory -> Requisition module.	The Requisition tab should contain all listed components **Tabs**: Requisition, Purchase Request, Purchase Order, and GR Quality Inspection **Radio Buttons**: Pending, Approved, Rejected, All
4	Ensure the Pharmacy tab is active.	**Preconditions**: The user must be logged in and located within the "Requisition" tab under the Inventory section of the Verification module. **Steps**: 1. Click on the pharmacy.	Verify the User is navigated to the pharmacy section from the inventory section and is active.

5	Ensure the "Purchase Request" sub-module of “Inventory” is navigable.	**Preconditions**: The user must be logged into the health system. The user is currently within the Verification module, specifically under the Inventory section. **Steps**: 1. Click on Inventory 2. Click on the Requisition tab 3. Click on the "Purchase Request" sub-tab.	Verify the user is taken to the “Purchase Request” tab.
6	Ensure that users can successfully use date-pickers to filter data.	**Preconditions**: The user must be logged into the health system. The user is currently on the Requisition tab of the Inventory sub-module of the Verification module. **Steps**: 1. Click on the Requisition tab 2. Click on the "From" date, and choose a date that is 7 days back from the current date. 3. Click on the "To" date, Choose the current date from the calendar that appears. 4. Click the "OK" button to apply the selected date range.	The 'Requested on' column date must fall within the date selected by the user.
7	Ensure the tooltip text is on the star icon.	**Preconditions:** Same as test case 6. **Steps:** 1. Hover the mouse on the "Star" Icon	Verify that the tooltip contains the exact text **"Remember this date"**
8	To verify that the system retains selected date ranges within the Verification module when navigating between the "Pharmacy" and "Inventory" tabs, ensuring that user inputs are preserved across different sections for consistent user experience.	**Preconditions:** The user must be logged into the health system. The user is located within the Requisition tab of the Inventory submodule in the Verification module **Steps**: 1. Set 'From' Date approx. 50 days older than the current date. 2. Set the 'To' Date as the current date. 3. After setting the dates, click on the "Star" icon. 4. Click on the "OK" button to apply the selected date range. 5. Navigate to the "Pharmacy" tab (just next to Inventory on top of page). 6. Navigate back to the "Inventory" tab.	Upon returning to the "Inventory" tab from the "Pharmacy" tab, the previously selected dates ("FROM" and "TO") should be remembered and displayed correctly.

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.

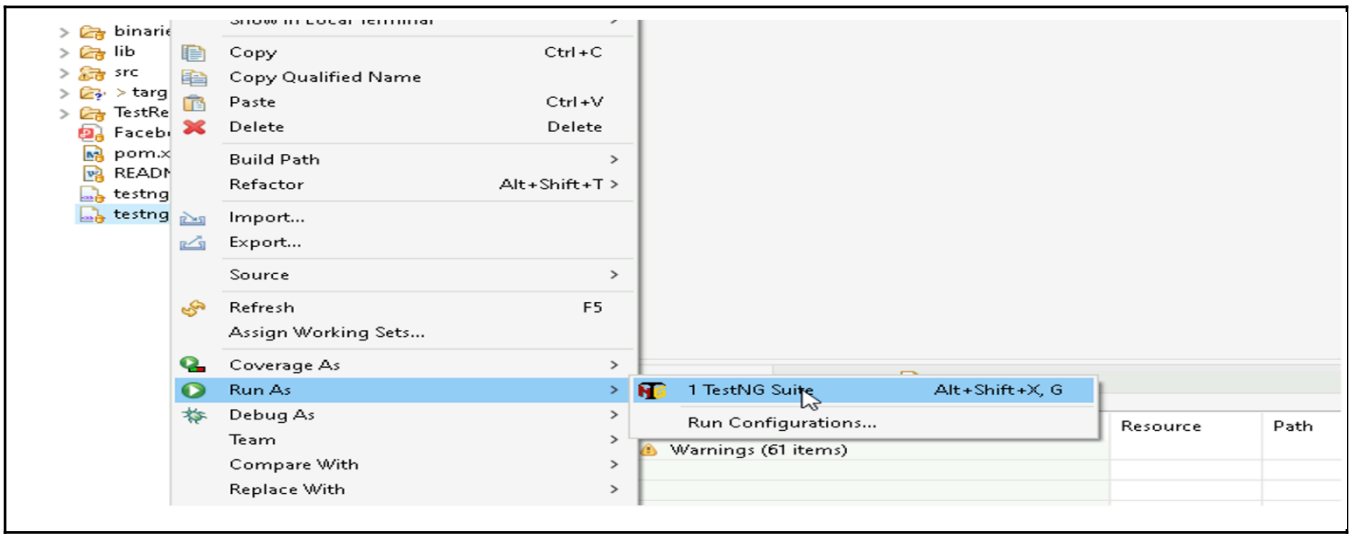
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

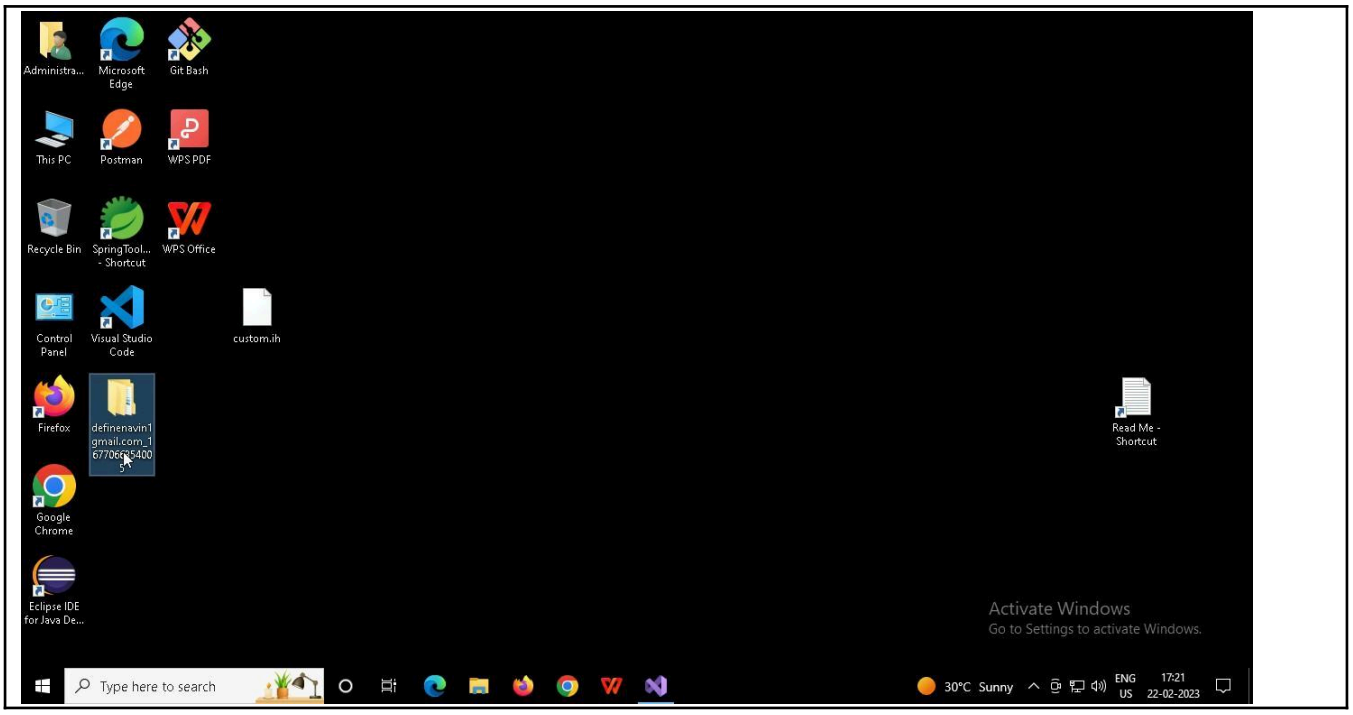
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

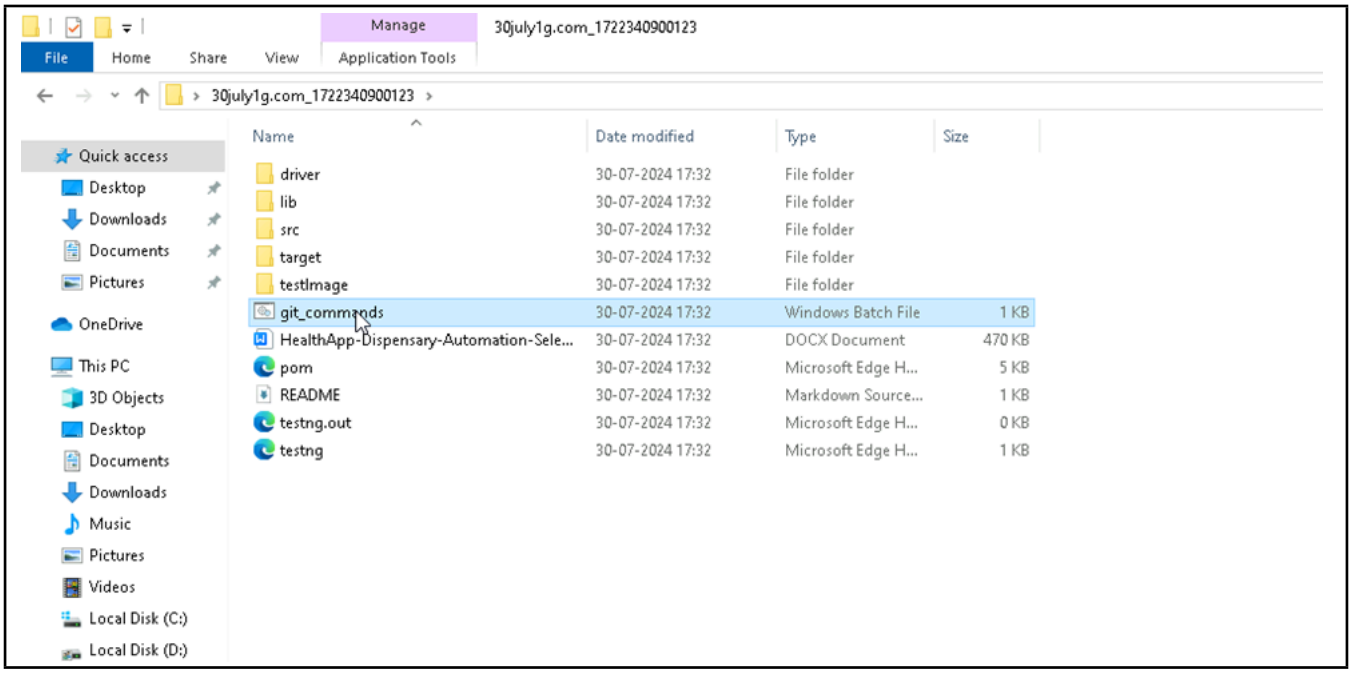
1. **You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.**
2. **You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite.**



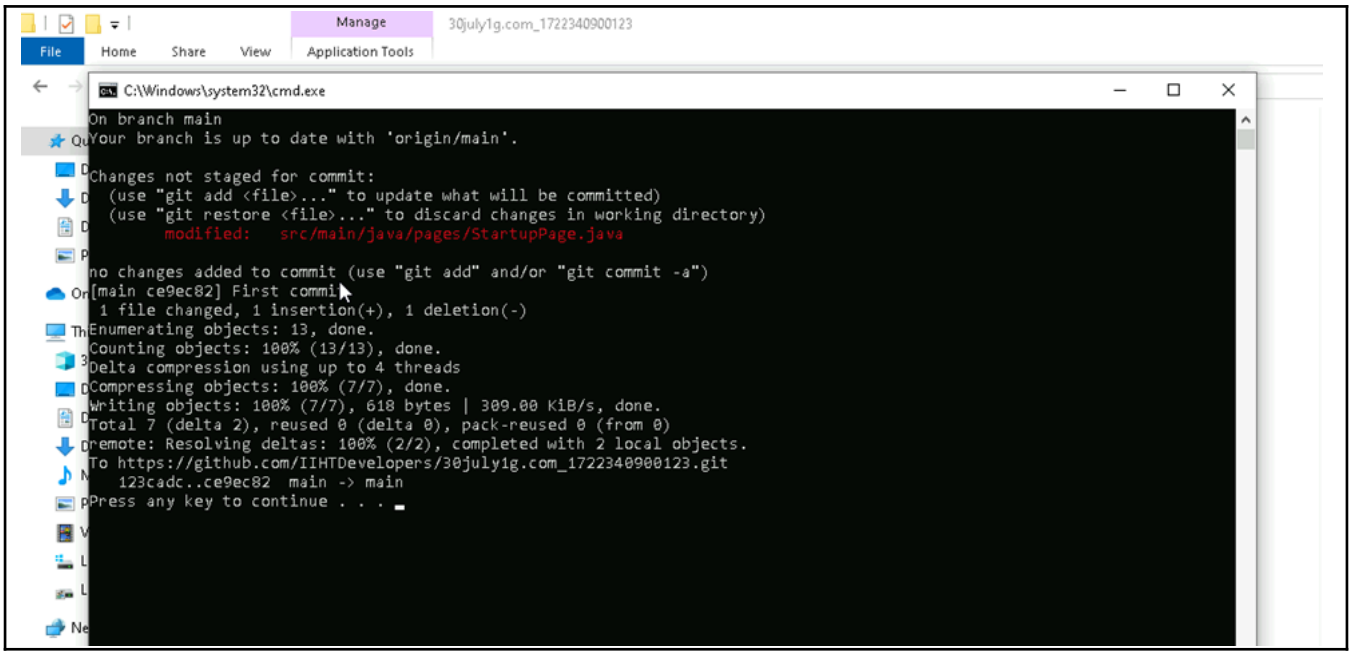
3. **Before final submission, you are also required to push your code to GIT. Following are the steps to follow:**



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

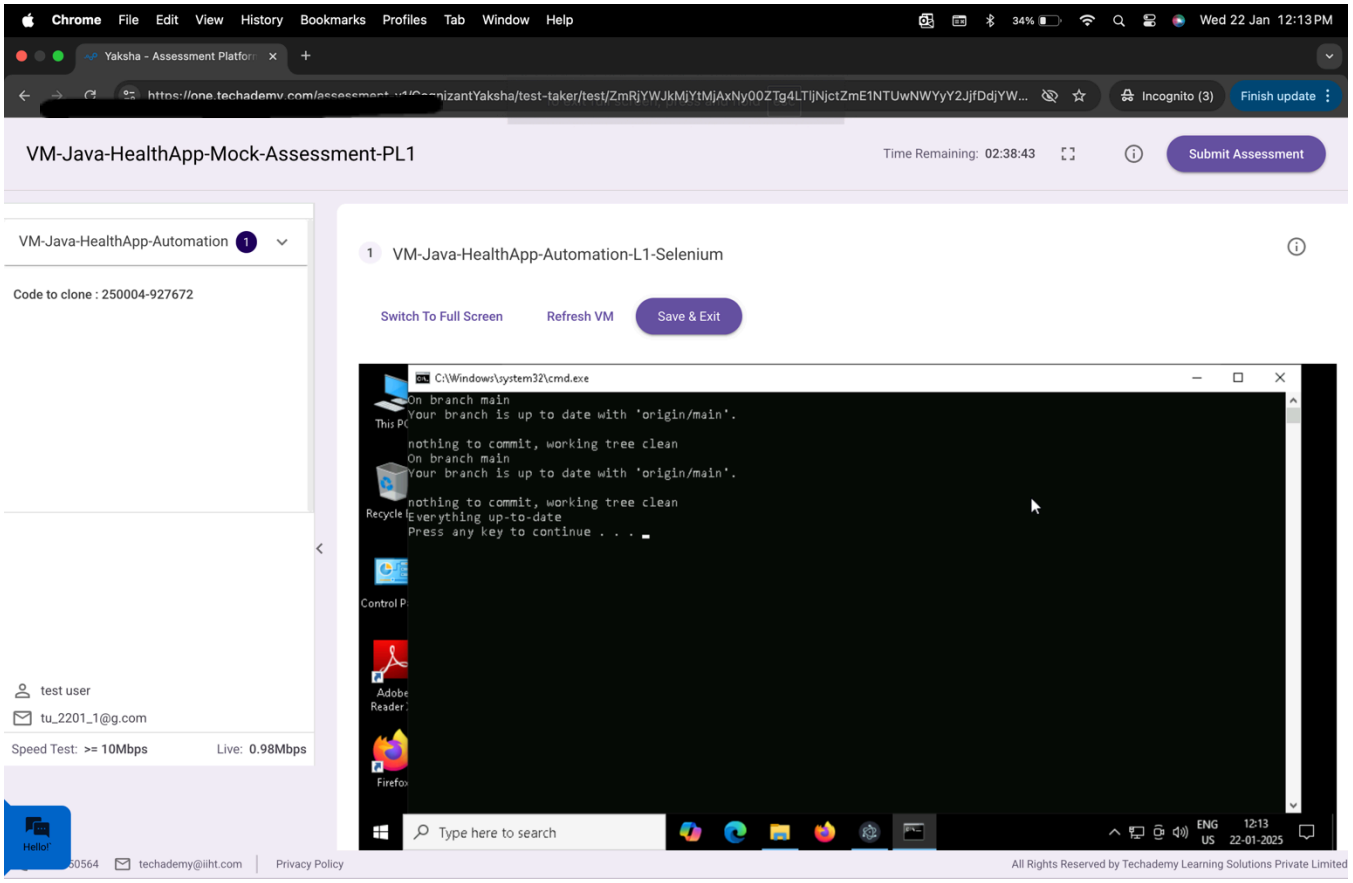


Double-click the batch file to run it. It will run the commands to push your code to GIT.



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