

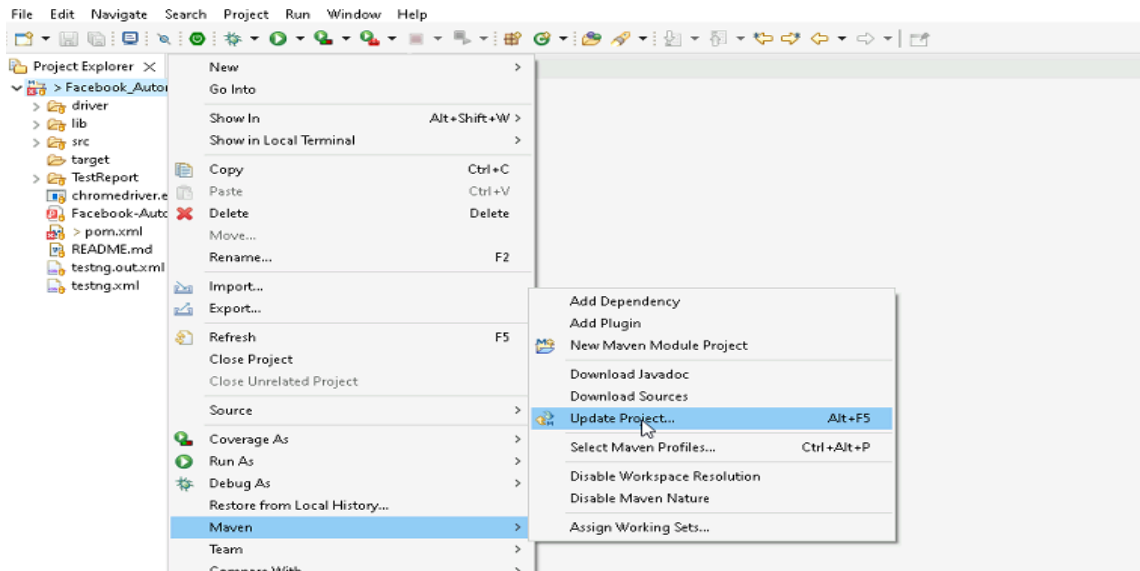
HEALTHAPP AUTOMATION

PATIENT MODULE – PL1

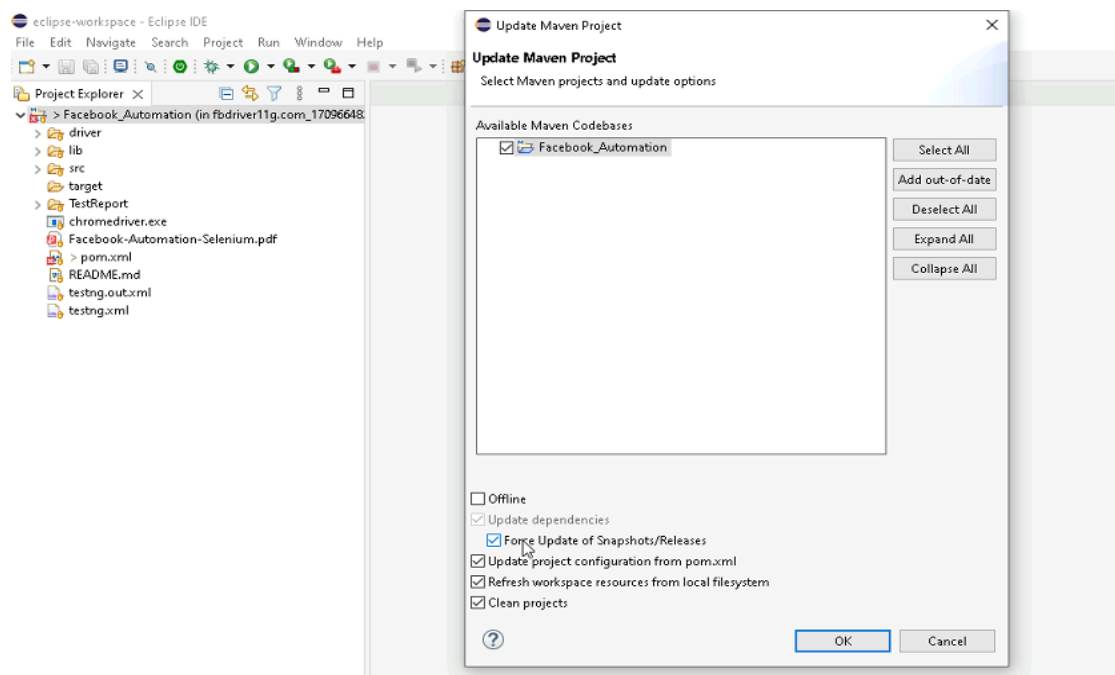
Pre-requisite:

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 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 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/utis/	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	<ol style="list-style-type: none">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	The URL to navigate to is Already defined here
	expected_data.xlsx	Contains data to fill in form\

/src/main/java/coreUtilities/utills	CommonEvents.java	<ol style="list-style-type: none"> 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:

Sl No.	Summary	Action	Expected Result
1	* Navigate to the URL. * Retrieve the Title and URL of the Home page. Check if the title & URL matches the expected title.	<ol style="list-style-type: none"> 1. go to the URL: https://healthapp.yaksha.com/ 2. login as a valid credential (username: admin, password: pass123) and click on the "Sign in" Button 3. get the title and URL of the Home page, post login 	Validate the title and URL of the Home page The title should be : DanpheHealth Url should be : https://healthapp.yaksha.com/Home/Index#
2	Ensure the sub-modules "Search Patient" and "Register Patient" are available.	Preconditions: The user must be logged into the health app. Steps: <ol style="list-style-type: none"> 1. Click on the expand icon (Toggle key) next to the Patient Module. 	Verify that Sub-modules "Search Patient" and "Register Patient" are displayed correctly under the Patient Module.
3	Ensure that the 'Search' textbox correctly displays its placeholder text as "Search (Minimum 3 Character)".	Preconditions: The user must be logged into the health app and on the 'Search Patient' page within the Patient module. Steps: <ol style="list-style-type: none"> 1. Obtain the placeholder text from the search textbox. 	Verify the placeholder text displayed is "Search (Minimum 3 Character)". Hint: (Placeholder text is the text value displayed inside the textbox, which will disappear when the user gives any input)

4	Ensure the presence of the "Take a Snapshot" button.	Preconditions: The user is logged into the healthApp system and on the "Search Patient" page of the Patient module. Steps: <ol style="list-style-type: none"> 1. Click on the "Register Patient" submodule. 2. Click on the camera icon to start the photo upload process. 3. Click on the "+ New Photo" button. 	Verify the presence of the "Take a Snapshot" button.
5	Ensure the error message for the "Phone number" field is "Primary Phone is required" , while we try to register a patient without filling in any information.	Preconditions: The user is logged into the health system and is on the profile picture form of the "Register Patient" submodule. Steps: <ol style="list-style-type: none"> 1. Click on the "Cancel" button to exit the photo upload interface. 2. Click on the "Basic Information" page link to access the form. 3. Without entering any data, directly click the "Register Patient" button. 	Verify that the error message for the "Phone number" field is displayed as "Primary Phone is required"
6	To ensure that data can be accurately entered into the 'Basic Information' form fields from an Excel source and that these entries are correctly displayed.	Preconditions: The user is logged into the healthApp system and is on the 'Basic Information' form of the 'Register Patient' submodule Steps (Use Expected_Data.xlsx to fetch the required values): <ol style="list-style-type: none"> 1. Fill in the 'First Name', 'Middle Name', 'Last Name', 'Age', and 'Phone Number' textbox. 	Verify that the value for 'First Name' is displayed as provided in the Expected_Data.xlsx
7	Check if the 'Blood Group' dropdown is present at the bottom of the form.	Preconditions: The user is logged into the healthApp system. The user is on the 'Basic Information' form within the 'Register Patient' submodule. Steps: <ol style="list-style-type: none"> 1. Scroll to the bottom of the form. 	Verify the presence of the field 'Blood Group' dropdown.
8	Highlight the 'Blood Group' dropdown.	Preconditions: The user is logged into the healthApp system. The user is at the bottom of the 'Basic Information' form in the 'Register Patient' submodule. Steps: <ol style="list-style-type: none"> 1. If the 'Blood Group' dropdown is present, highlight the web element using the highlight feature. 	Verify that the "Blood Group" Dropdown is highlighted.

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.

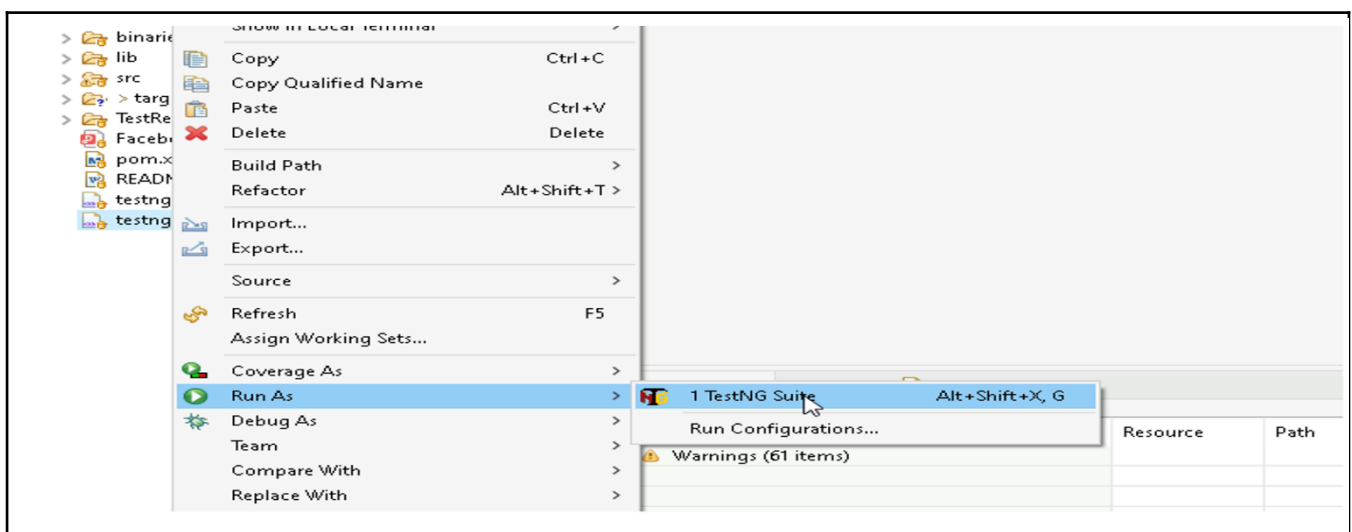
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

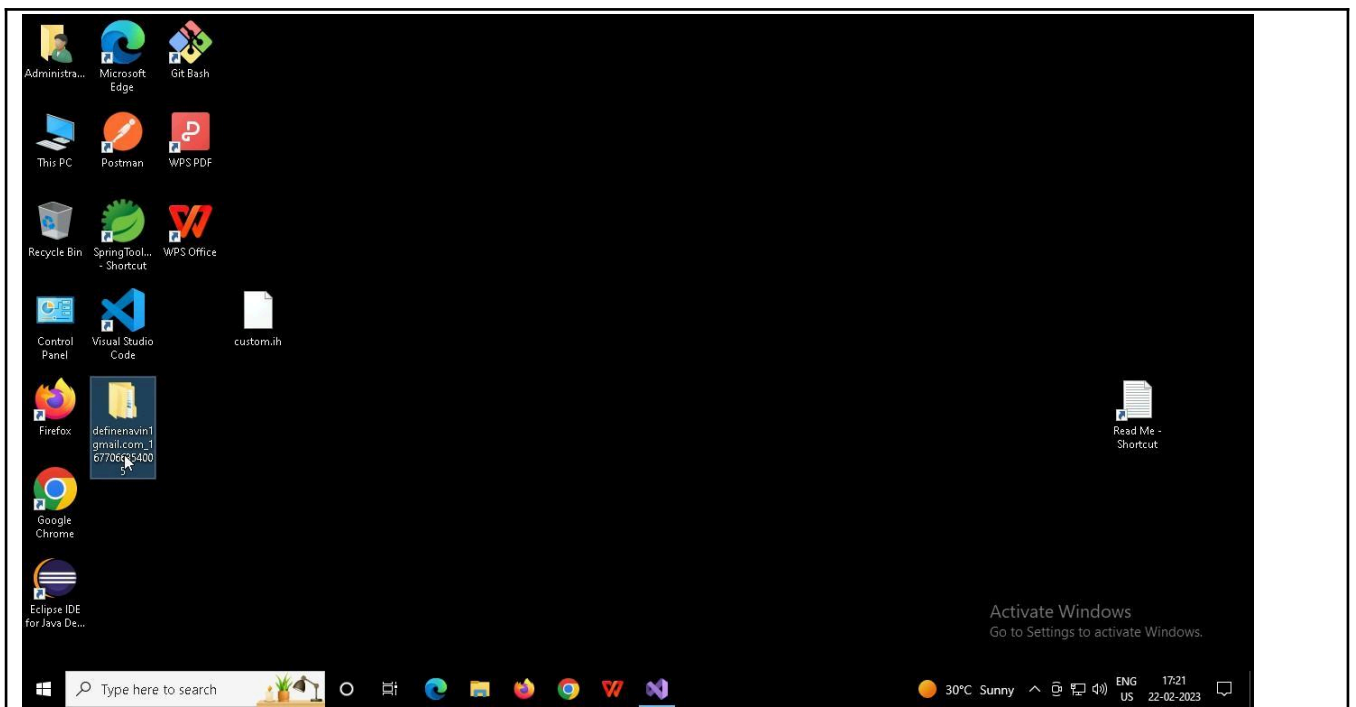
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

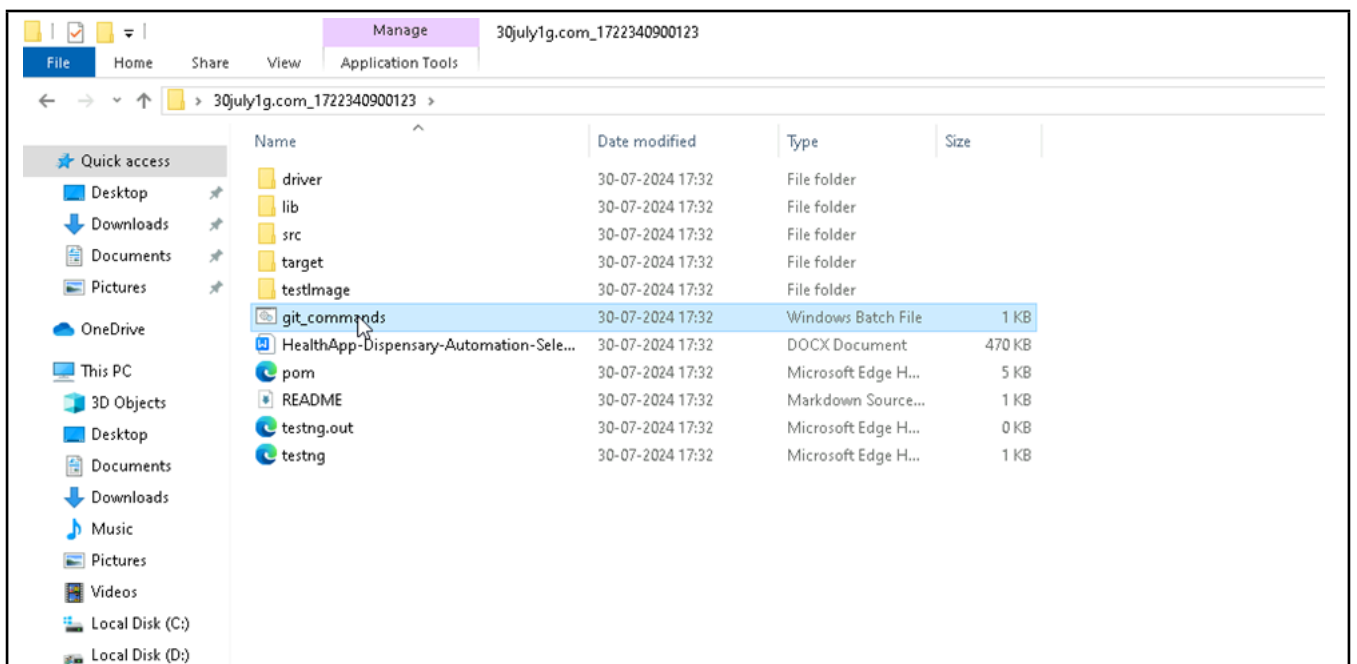
1. You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.
2. You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite.



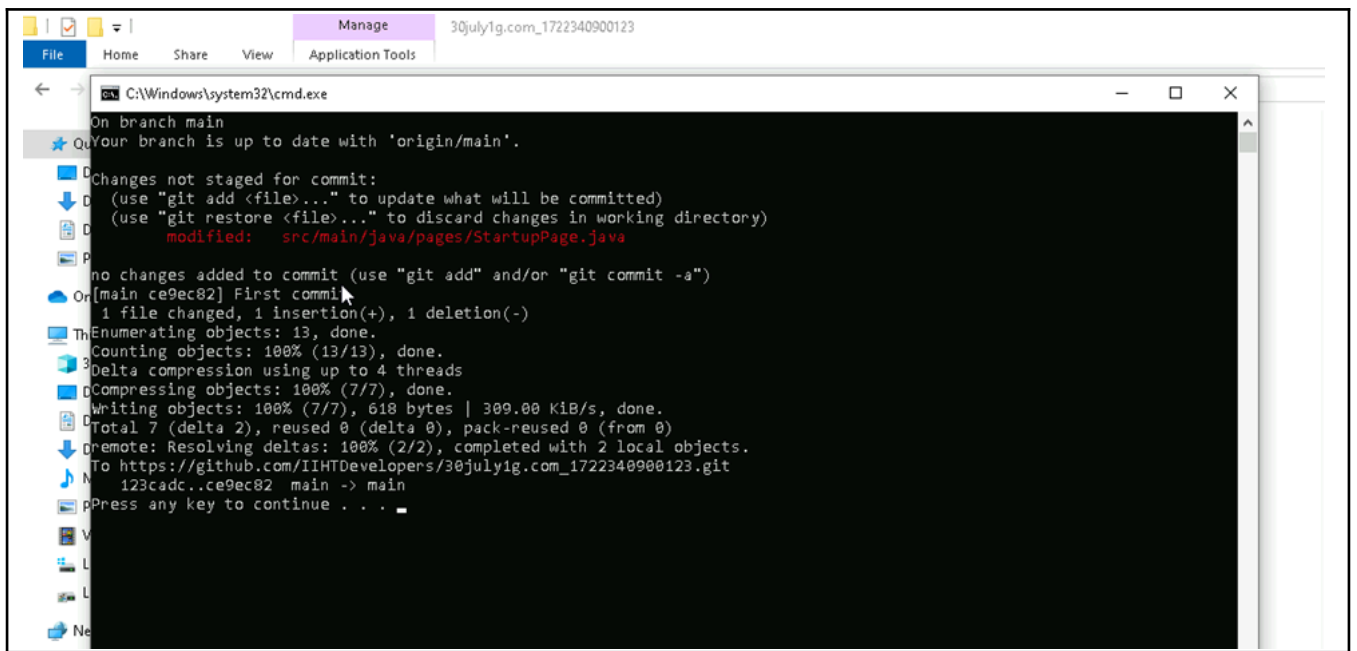
3. Before final submission, you are also required to push your code to GIT. Following are the steps to follow:



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.



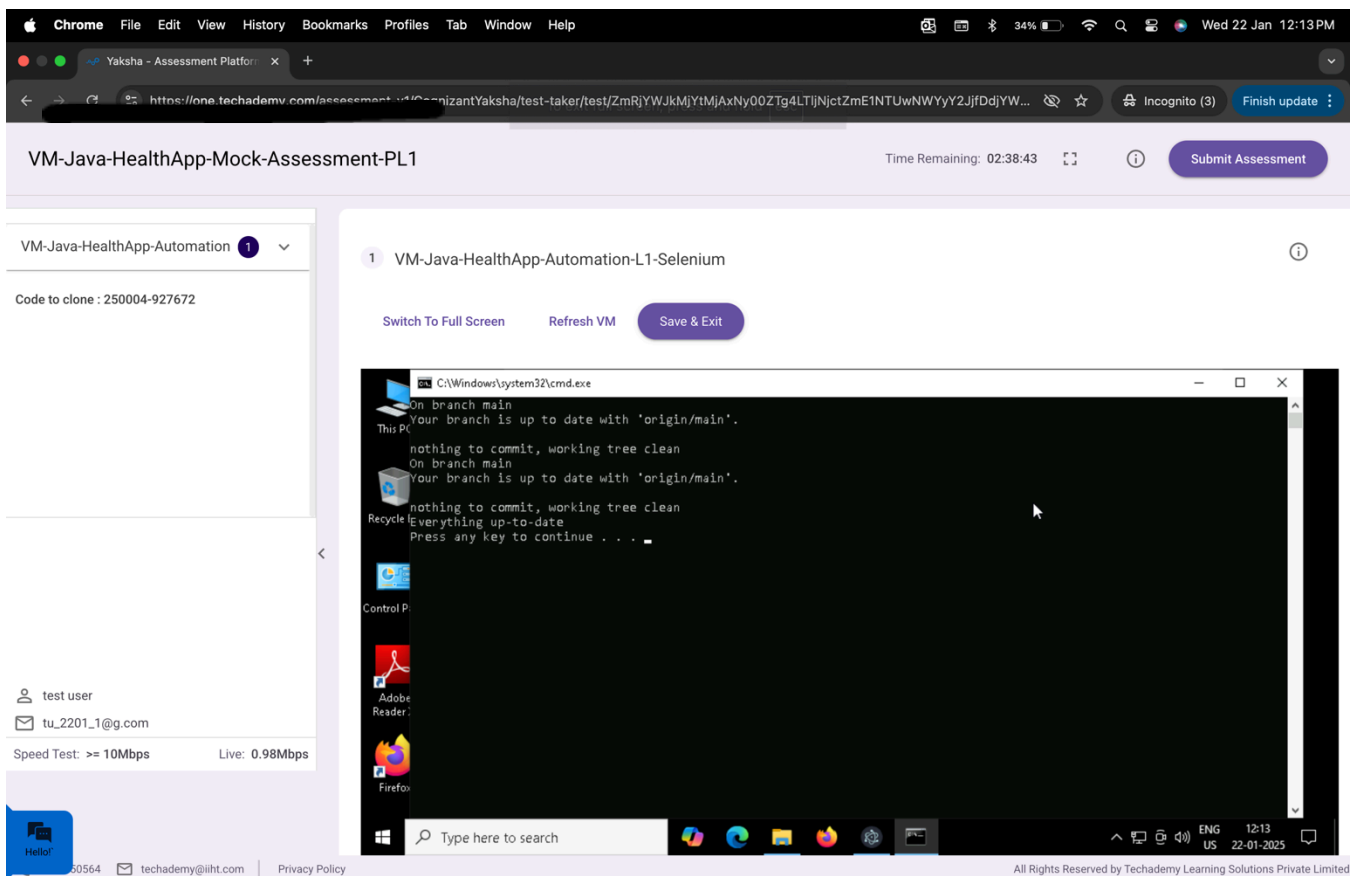
```
C:\Windows\system32\cmd.exe
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   src/main/java/pages/StartupPage.java

no changes added to commit (use "git add" and/or "git commit -a")
[master ce9ec82] First commit
1 file changed, 1 insertion(+), 1 deletion(-)
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 618 bytes | 309.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/IIHTDevelopers/30july1g.com_1722340900123.git
   123cadc..ce9ec82  main -> main
Press any key to continue . . .
```

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.



The screenshot shows the TechAdemy assessment platform interface. The browser address bar displays the URL: <https://one.techademy.com/assessment/v4/ConfigYaksha/test-taker/test/ZmRjYWJkMjYtMjAxNy00ZG4LTjNjctZmE1NTUwNWYyY2JfDdjYW...>. The page title is "VM-Java-HealthApp-Mock-Assessment-PL1". The time remaining is 02:38:43. A "Submit Assessment" button is visible.

The left sidebar shows the assessment details for "VM-Java-HealthApp-Automation" with a code to clone: 250004-927672. The user is logged in as "test user" with email "tu_2201_1@g.com". The speed test shows ">= 10Mbps" and live speed is "0.98Mbps".

The main area displays the assessment title "1 VM-Java-HealthApp-Automation-L1-Selenium". Below the title are buttons for "Switch To Full Screen", "Refresh VM", and "Save & Exit".

The "Switch To Full Screen" button is active, and a Windows desktop environment is shown. The desktop has a taskbar with icons for "Control Panel", "Recycle Bin", "Adobe Reader", and "Firefox". The command prompt window is open, showing the following output:

```
C:\Windows\system32\cmd.exe
On branch main
Your branch is up to date with 'origin/main'.

This PC
nothing to commit, working tree clean
On branch main
Your branch is up to date with 'origin/main'.

Recycle Bin
nothing to commit, working tree clean
Everything up-to-date
Press any key to continue . . .
```


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

All the Best