HEALTHAPP AUTOMATION RADIOLOGY 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" : Select "Update Project"

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

HealthApp   Automation   –   Verification

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/pages	radiology_page.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 to. Already URL is 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

HealthApp   Automation   –   Verification

	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	Ensure that the Radiology module is available and accessible within the HealthApp.	**Preconditions**: • The user must have successfully logged into the HealthApp application. **Steps**: 1. **Login Verification**: Confirm that the user is logged into the HealthApp. 2. **Locate Radiology Module**: Scroll through the main menu to find the Radiology section. 3. **Access Radiology Module**: Click on the Radiology module.	Radiology module should be present.
2	To ensure that all designated sub-modules are correctly displayed under the Radiology module.	**Preconditions**: The user must be logged into the health system. **Steps**: 1. **Access Radiology Module**: Click on the drop-down arrow or equivalent control next to the Radiology module 2. **Verify Sub-Modules**: Observe the list of sub-modules that appear under the Radiology module. Ensure that all expected sub-modules are displayed. The expected sub-modules include: ○ **List Requests** ○ **List Reports** ○ **Edit Doctors** ○ **Ward Billing**	All sub-modules should be displayed correctly. Expected Sub modules are : List Requests, List Reports, Edit Doctors, Ward Billing
3	To confirm that clicking on the Radiology module navigates to the default sub-module, which should be the 'List Requests' section, Velidate the URL.	**Preconditions**: The user must be logged into the healthApp system. **Steps**: 1. **Navigate to Radiology Module**: Click on the Radiology module in the main navigation of the health system. This action should direct the user to the default view with **"List Request"** module.	Upon clicking the Radiology module, the browser should automatically navigate to the default sub-module 'List Requests'.

HealthApp   Automation   –   Verification

4	To ensure that the 'List Request' section within the Radiology module is equipped with all necessary components for effective operation, including buttons, fields, dropdowns, date pickers, and tooltips.	**Preconditions**: The user must be logged into the healthApp system and located within the Radiology module. **Steps**: 1. **Access Radiology Module**: Click on the drop-down arrow within the Radiology module to expand and reveal its sections. 2. **Navigate to List Request Section**: Select the 'List Request' section to view its contents and available functionalities.	he 'List Request' section should be properly accessible and display the following components: **Buttons**: Ensure the presence of buttons like 'Ok', 'First', 'Previous', 'Next', 'Last', 'Scan Done', 'Add Report', and 'Edit Report'. **Fields**: Verify the inclusion of a search bar for filtering requests. **Dropdowns**: Check for dropdown menus for additional filtering options such as 'Filter' and 'Date range'. **Date Pickers**: Confirm the availability of 'From' and 'To' date pickers. **Tooltip**: A tooltip should be present on the 'Star' figure.
5	To ensure seamless navigation between different sub-modules within the Radiology module.	**Preconditions**: The user is logged into the healthApp system and is within the Radiology module. **Steps**: 1. **Navigate to List Reports**: Click on the 'List Reports' option to ensure it is active. 2. **Switch to Edit Doctors**: From 'List Requests', click on the 'Edit Doctors' link to navigate to the section. 3. **Navigate to Ward Billing**: After verifying the 'Edit Doctors' section, click on the 'Ward Billing' option to access it. 4. **Return to List Requests**: Finally, navigate back to the 'List Requests' to confirm the ability to switch back to the initial section.	Clicking on each sub-module link or button should lead to the respective section without any delays or errors.
6	Ensure that the 'Next' button is visible after users scroll to the bottom of the Radiology module, facilitating navigation to subsequent records or pages.	**Preconditions**: The user is logged into the healthApp system. The user is on "List Request" submodule of the Radiology module. **Steps**: 1. **Set Filter to Show All Records**: Select the "All" option from the "Filter" dropdown to display all records, ensuring there are enough entries to necessitate scrolling. 2. **Scroll to Bottom of Page**: Perform a vertical scroll to reach the bottom of the Radiology module page.	The "Next" button should be visible after scrolling vertically.
7	Ensure that the tooltip for the 'Star' icon in the List Requests section of the Radiology module accurately displays helpful text when hovered over by the mouse.	**Preconditions**: The user is on "List Request" submodule of the Radiology module. **Steps**: 1. **Locate and Hover Over 'Star' Icon**: Move the mouse cursor over the 'Star' icon within the List Requests interface. 2. **Observe Tooltip**: Carefully watch for a tooltip to appear when the mouse hovers over the 'Star' icon.	Tooltip should be present when hover the mouse on star with text "Remember this date"

HealthApp Automation – Verification

8	Ensure that after setting date filters and interacting with the 'Star' tooltip in the List Requests section of the Radiology module, these settings remain unchanged even after refreshing and navigating back to the page.	**Preconditions**: The user must be logged into the health system. The user is within the Radiology module, specifically in the List Requests section. **Steps**: 1. **Set Date Filters**: Navigate to the Radiology module and access the "List Requests" section. Set the "FROM" date to "Jan 2020" and the "TO" date to "March 2024". 2. **Interact with Star Tooltip**: Click on the 'Star' tooltip to observe any functionalities or messages it might display. 3. **Confirm Settings**: Click the "OK" button to confirm the date settings. 4. **Refresh and Re-Navigate**: Refresh the page and navigate back to the "List Requests" section. 5. **Verify Tooltip and Dates**: After navigating back, check the 'Star' tooltip for any changes and verify if the "FROM" and "TO" dates remain set to "Jan 2020" and "March 2024", respectively.	The date settings in the "FROM" and "TO" fields should be unchanged, maintaining the user's previous selections despite the page refresh.

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.
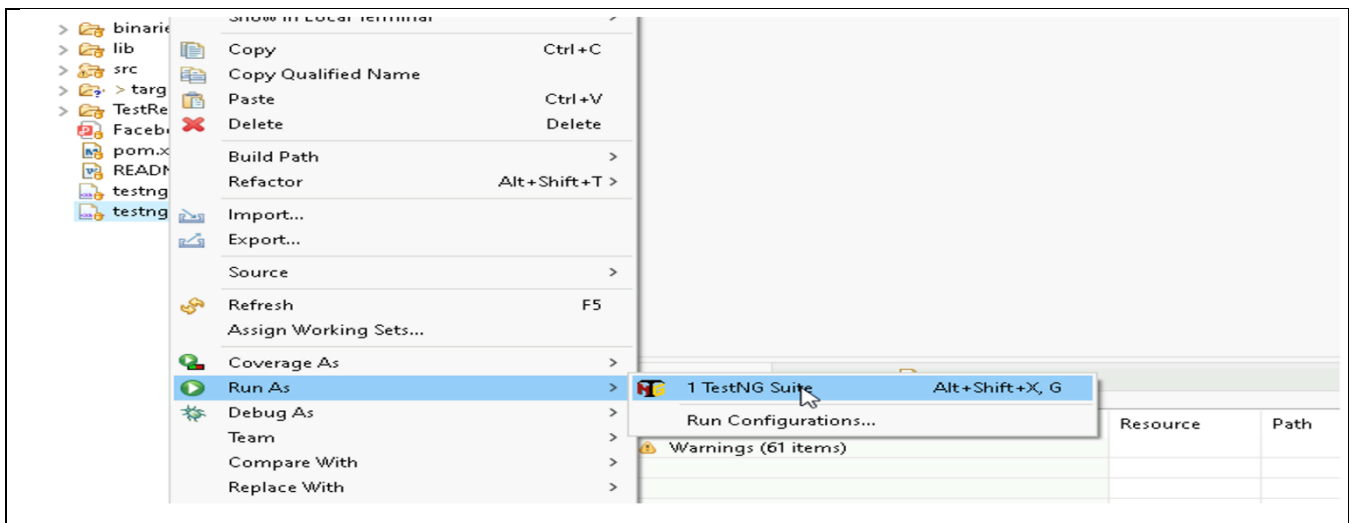

IMPLEMENTATION/FUNCTIONAL REQUIREMENT
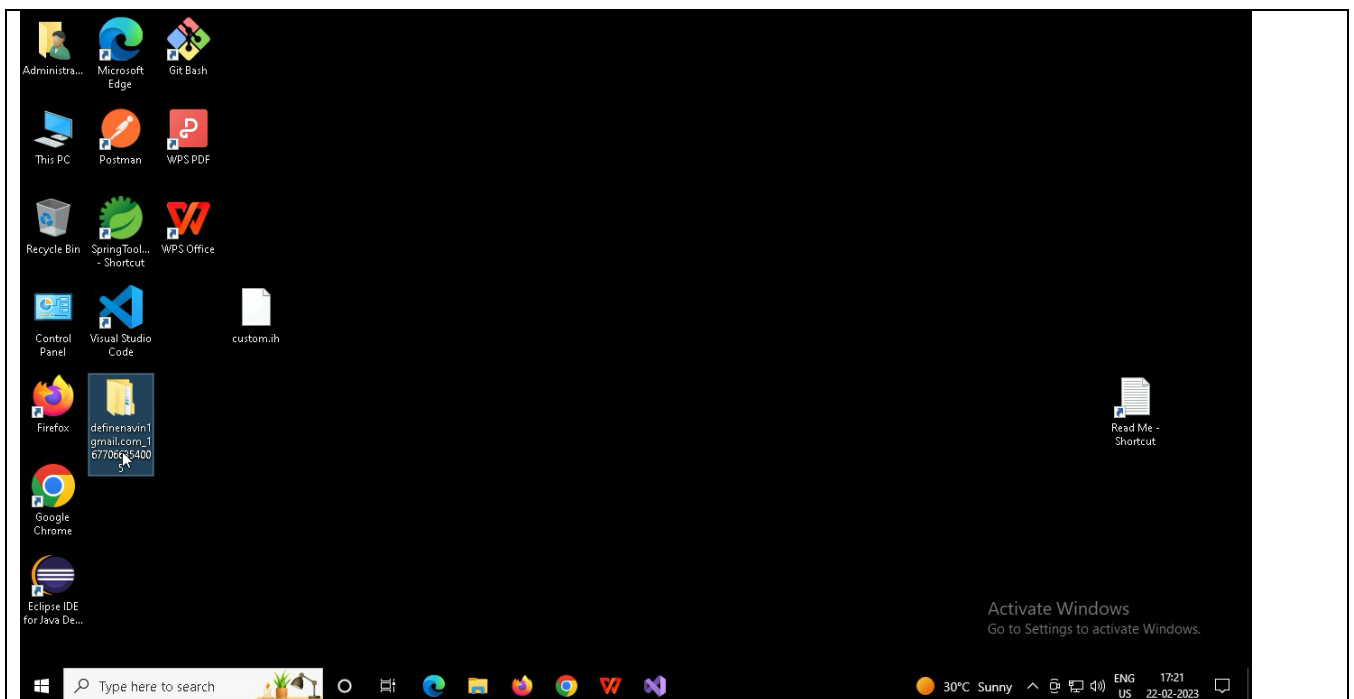
1.1 CODE QUALITY/OPTIMIZATIONS

1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.
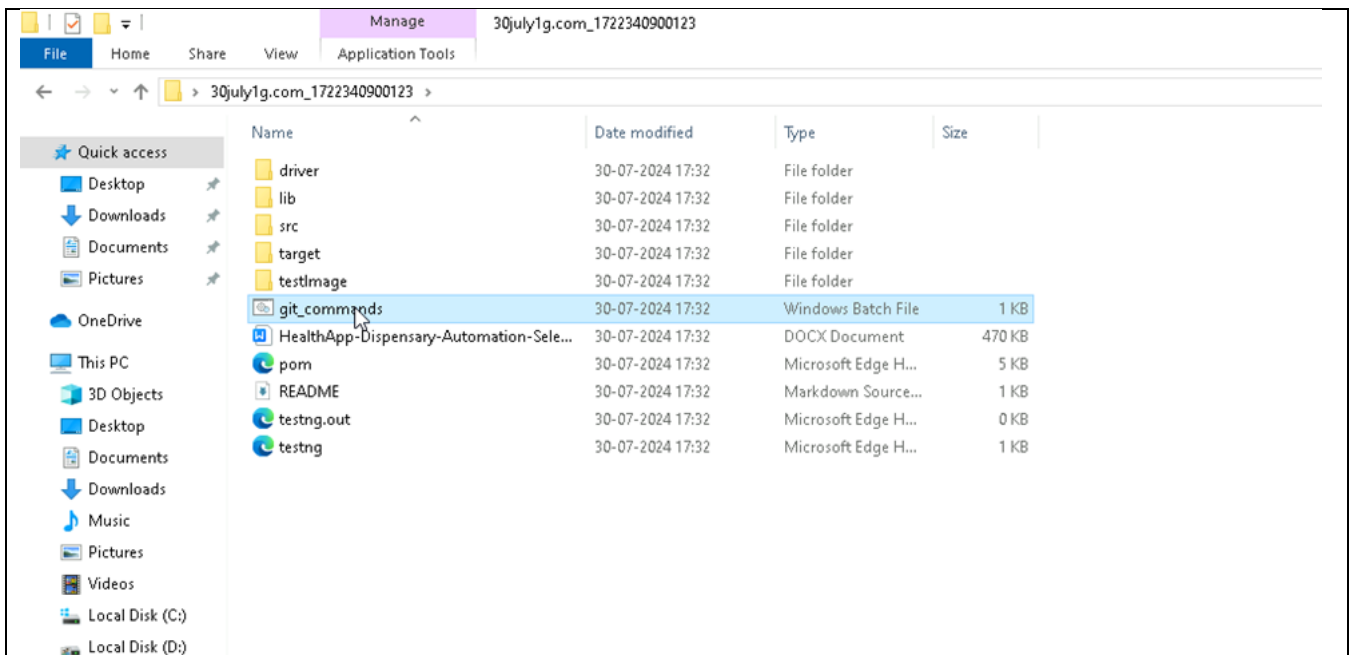
EXECUTION STEPS TO FOLLOW
1. **You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.**
2. **You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite.**
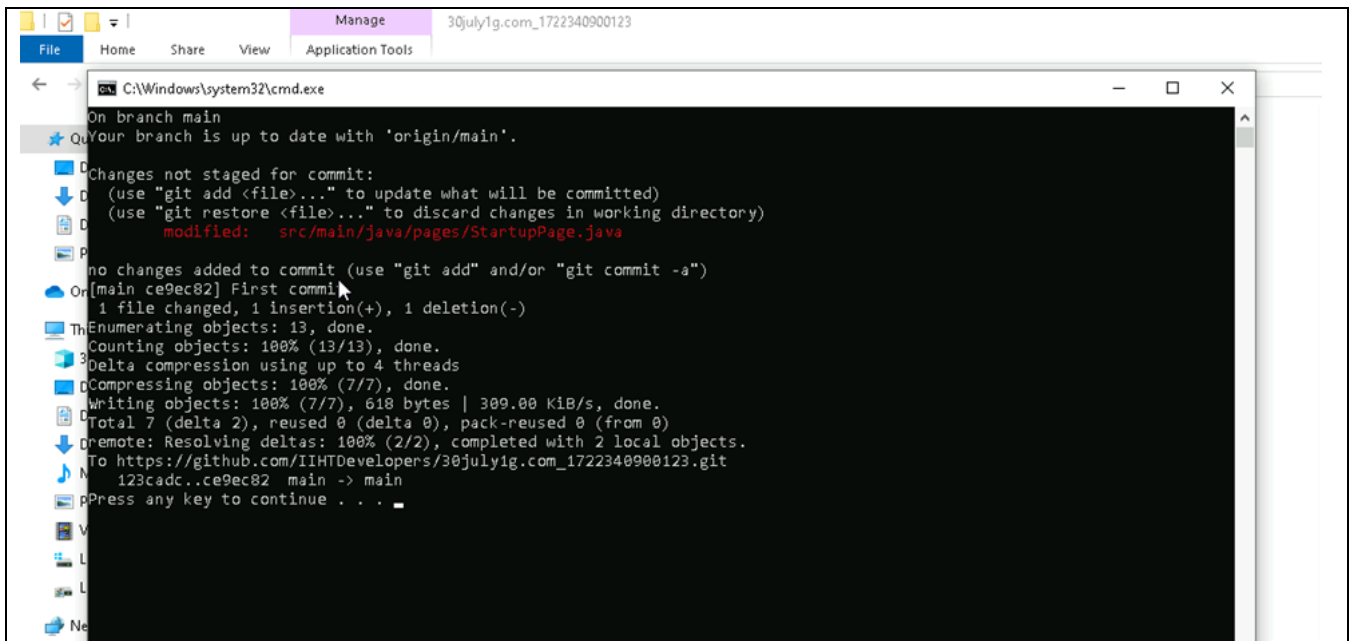
HealthApp   Automation   –   Verification

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

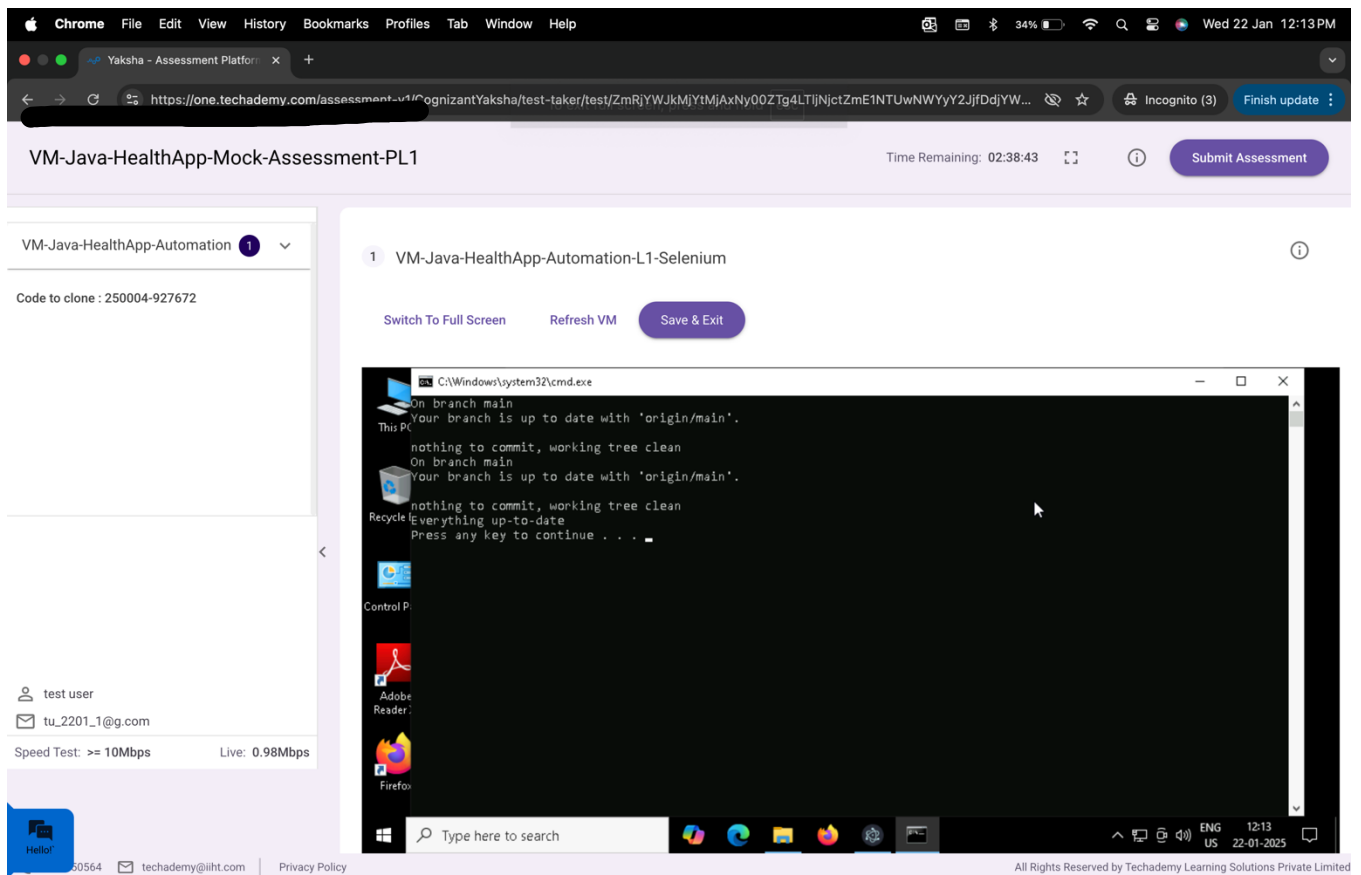In your project folder, you will find a batch file named git_commands

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.

HealthApp Automation – Verification

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

===

All the Best

HealthApp Automation – Verification