

HEALTHAPP AUTOMATION PATIENT 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



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	verification_page.java	<ol style="list-style-type: none">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/utlis	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:

Sl No.	Summary	Action	Expected Result
1	<ul style="list-style-type: none"> * Navigate to the URL. * Retrieve Title and URL of the Home page. * Verify Title & URL: Check if the title & URL matches the expected title. 	<ol style="list-style-type: none"> 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 	Validate the title and URL of the Home page 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: 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. 	Sub-modules "Search Patient" and "Register Patient" should be correctly displayed under the Patient Module.
3	Ensure that the 'Search' textbox correctly displays its placeholder text as "Search (Minimum 3 Character)".	Preconditions: 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. Click on the Search textbox to focus on it. 2. Obtain the placeholder text from the textbox. 	Ensure the placeholder text matches "Search (Minimum 3 Character)". Hint: (Placeholder text is the text value displayed inside the textbox, which becomes disappear once you start to input your own value)

4	Verify the presence of "Take a Snapshot" button.	Preconditions: 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. 4. Highlight the "Take a Snapshot" button. 5. Click on the "Cancel" button to exit the photo upload interface. 	Verify the presence of "Take a Snapshot" button.
5	Ensure the error message for the "Phone number" field is "Primary Phone is required" , while we try to register patient without filling any information.	Preconditions: 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 "Basic Information" page link to access the form. 2. Without entering any data, directly click the "Register Patient" button. 	Verify the error message for the "Phone number" field, ensuring it states "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: 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. Find the "First Name" textbox and fill the value. 2. Find the "Middle Name" textbox and fill the value. Find the "Last Name" textbox and fill the value. 3. Find the "age" textbox and fill the value. 4. Find the "Phone number" textbox and fill the value. 	Each textbox ('First Name', 'Middle Name', 'Last Name', 'Age', 'Phone Number') should contain the correct information as provided by the Excel data. The 'First Name' field, in particular, should correctly display the entered data, confirming the accuracy of the input process.
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. It should be clearly visible at the bottom of the 'Basic Information' form.
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, apply a visual highlight to enhance its visibility. 	"Blood Group" Dropdown should be Explicitly 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

HealthApp Automation – Patient Module

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.

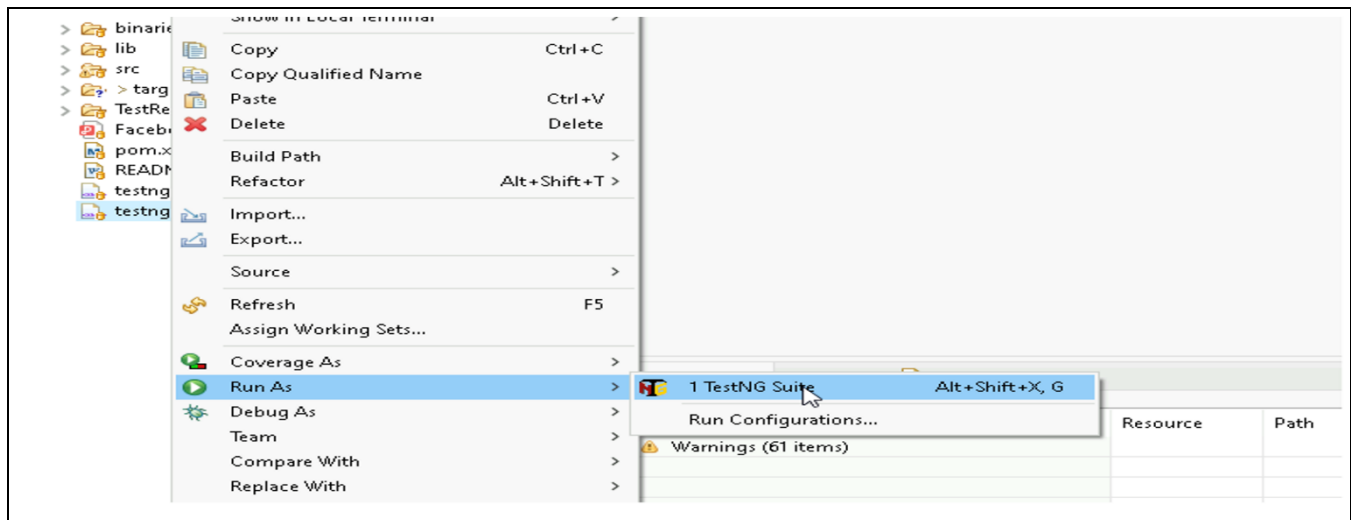
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

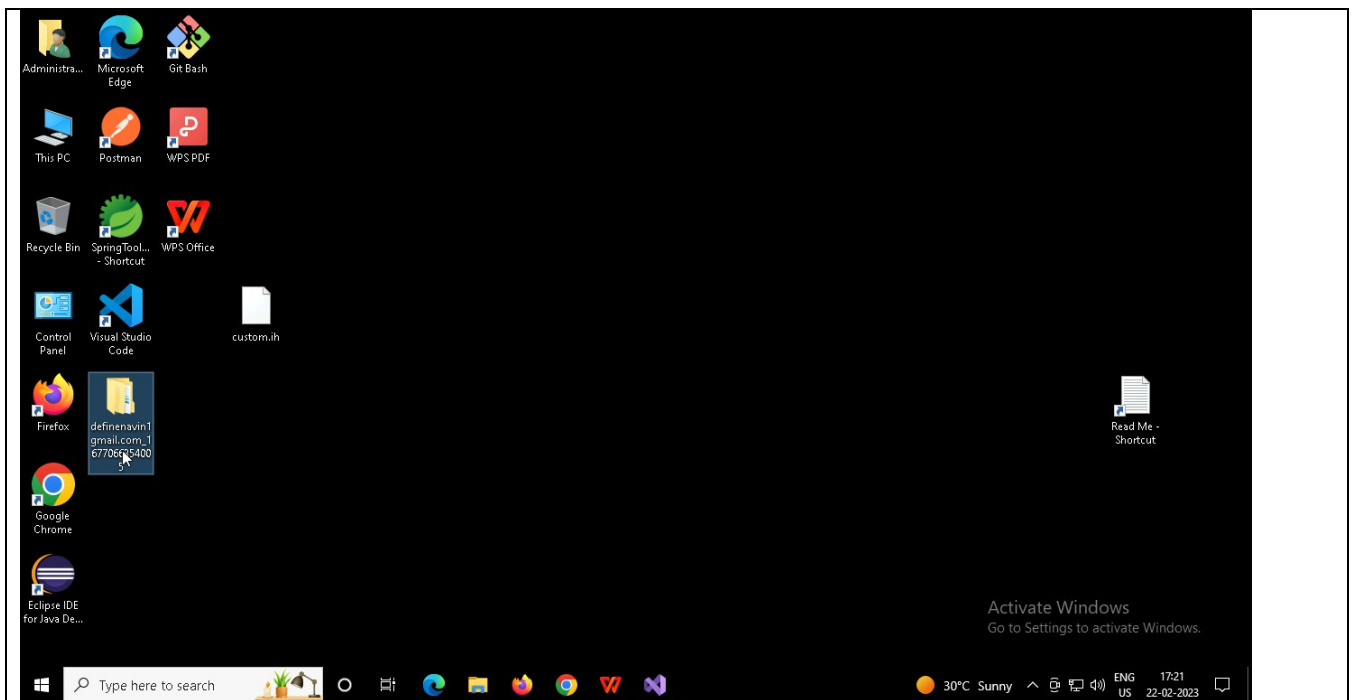
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

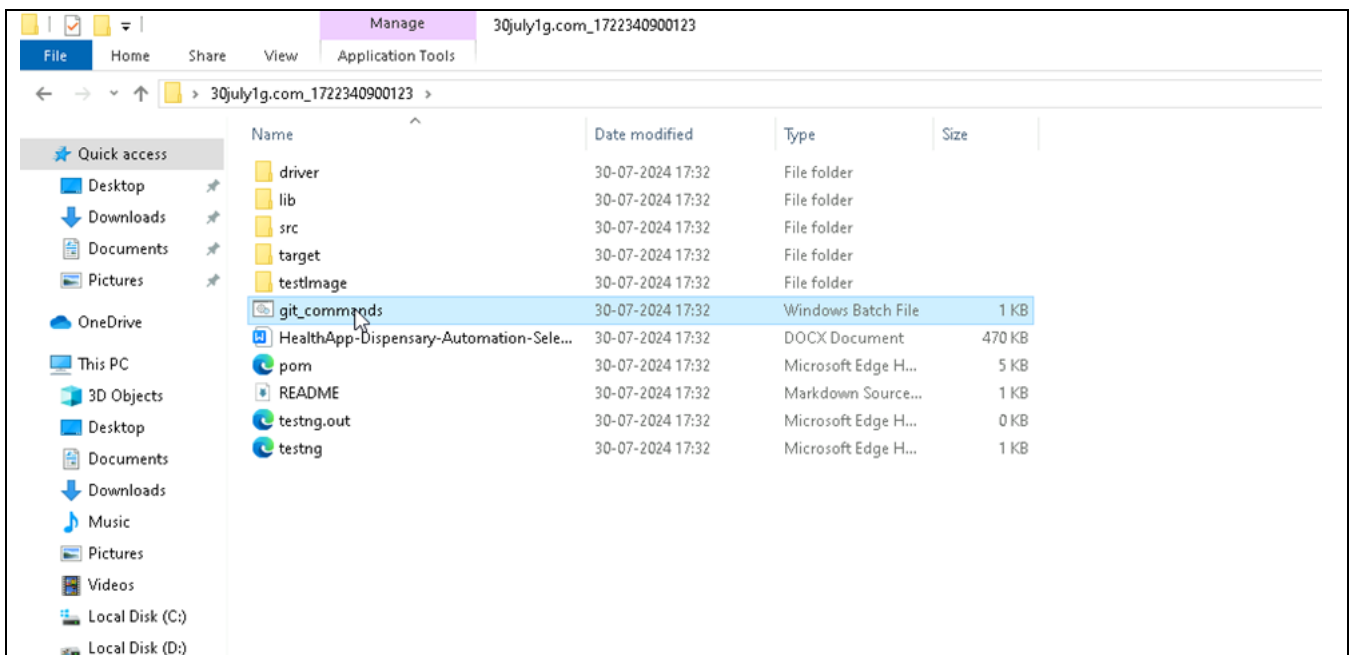
1. You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.
2. You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite.



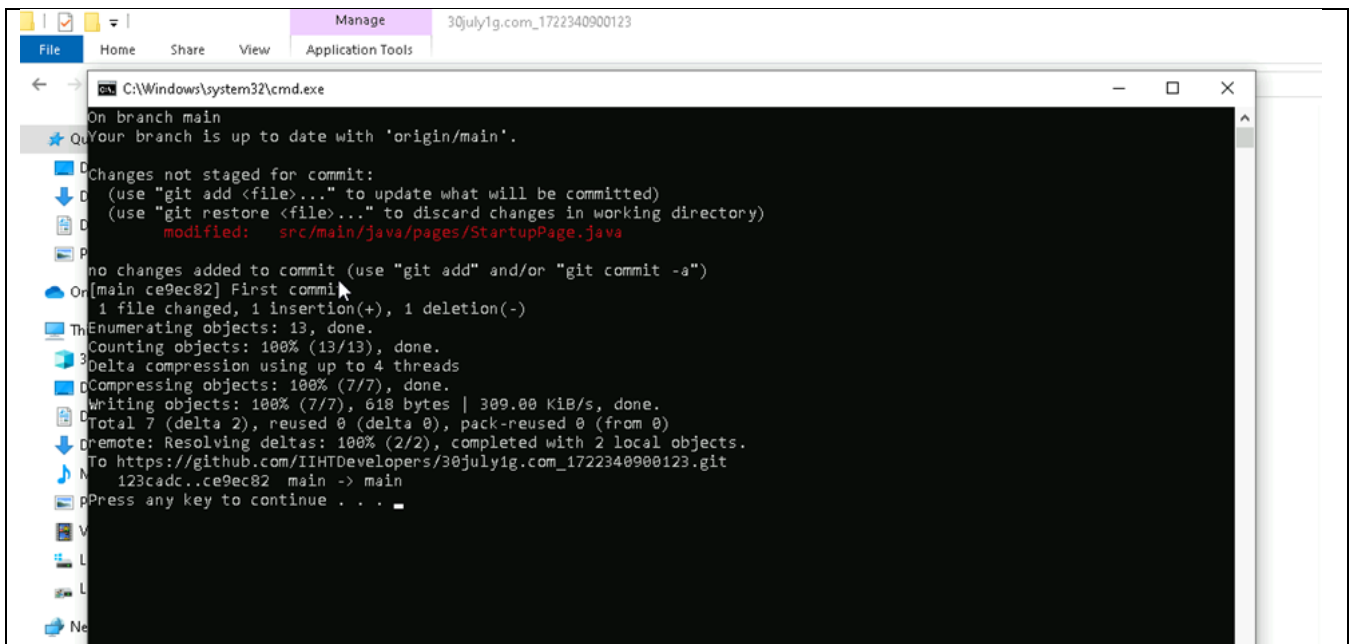
3. Before final submission, you are also required to push your code to GIT. Following are the steps to follow:



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

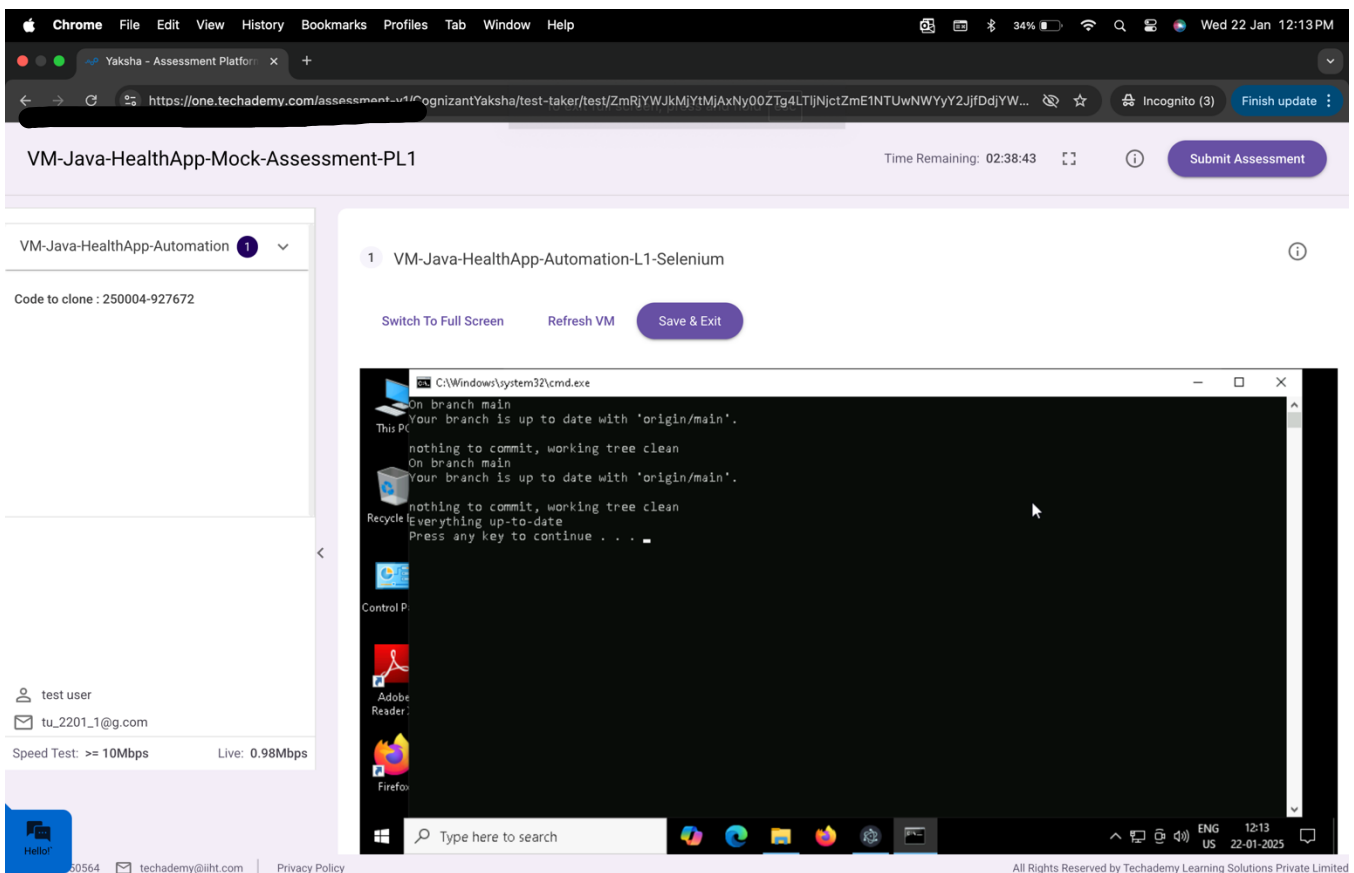
On branch main
Your branch is up to date with 'origin/main'.

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/30julyig.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 a web browser window with the URL <https://one.techademy.com/assessment-v4/IncognizantYaksha/test-taker/test/ZmRjYWJkMjYtMjAxNy00ZG4LTjNjctZmE1NTUwNWYyY2JfDdjYW...>. The page title is "VM-Java-HealthApp-Mock-Assessment-PL1". A sidebar on the left shows "VM-Java-HealthApp-Automation" with a code to clone: "250004-927672". The main area displays "1 VM-Java-HealthApp-Automation-L1-Selenium" with buttons for "Switch To Full Screen", "Refresh VM", and "Save & Exit". A "Submit Assessment" button is in the top right. A terminal window is open, showing the same Git status and commit output as the first image. The bottom of the page shows a footer with "All Rights Reserved by Techademy Learning Solutions Private Limited".

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

HealthApp Automation – Patient Module

All the Best