HealthApp Automation-Substore Module

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”



1. In Update Maven Project Box Select “Force Update of Snapshots/Releases” and click OK



Template Code Structure:

1. Below are the packages and files you will be required to work upon.
2. Other Files and packages you can ignore.
3. In other Files and packages do not do any changes. It would affect your evaluation.
4. 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 | 1. Contains methods to read from excel file. 2. Method is in templated form. 3. **You will be required to implement these methods as very first activity, because even URL to navigate to, is read using these methods.** |
| /src/main/java/pages | substore\_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 |
|  | 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 SubStore module is present or not | 1. Login in the healthapp application 2. Scroll down menu till SubStore 3. Click on the SubStore | SubStore module should be present |
| **2** | Verify the sub-module buttons present in Select your Substore modal | 1. Login in the healthapp application 2. Click on the substore  3. "Select your Substore" pop up | Expected value that should be present in "Select your Substore" modal Expected sub-modals button: Accounts,male ward SubStore,SubStore1,SubStore1 |
| **3** | Verify that the "Accounts' sub-module is clickable | Precondition: User should be logged in and on the Verification section 1. Click on the Inventory Module 2. Hover over on module signout button and get text | Verify text on hover contains "To change, you can always click here." |
| **4** | Verify all sub-modules are displayed correctly after Clicking on the "SubStore " Module. | Pre condition: User should be logged in and it is on SubStore module 1. Login in the healthapp application 2. Click on the Substore 3. "Select your Substore" pop up  4. Click on "Account" sub-modal 5. Click on the "Inventory" sub-module 6. Click on" Pharmacy" sub-module | All sub-modules should be displayed correctly.  Expected Sub modules are : Pharmacy, Inventory |
| **5** | Verify all section are displayed correctly after Clicking on the "Inventory " Sub-Module. | Pre condition: User should be logged in and it is on SubStore module 1. Click on the SubStore Module drop-down arrow  2. Click on "Accounts" from the "Select your Subdtore"  3. Click on the "Inventory" 4. Navigate to "Stock" section 5. Click on the all the section of the "inventroy" sub-module 6. Navigate back to "Stock" section | All section should be displayed correctly. Expected Sub modules are : Stock, Inventory Requisition, Consumption, Reports, Patient Consumption,Return |
| **6** | Verify to navigate to each sections which are present in the "Inventory" sub-module | Pre condition: User should be logged in and it is on Inventory sub-module 1. Naviagte to the "Inventory" sub-module 2. Click on the "Inventory Requisition" section 3. Click on the "Stock" 4. Click on the "Consumption" 5. Click on the " Reports" 6. Click on the "Patient Consumption" 7. Click on the "Return" 8. Naviaget back to the "Inventory Requisition" section | Ensure that it should navigate to each sections of the "Inventory" module |
| **7** | Take Screenshot of the current page | Under Substore module > Inventory Section Take the screenshot of the current page | Screenshot shot of the page should be saveed under screenshot folder |
| **8** | Verify the presence of Inventory Requisition section in Inventory sub-module with all fields | Pre condition: User should be logged in and it is on SubStore module 1. Click on the SubStore module drop-down arrow  2. Click on Inventory sub-module 3. Click on Inventory Requisition section | Inventory Requisition section in Inventory sub-module should be present  Expected value that are present in Inventory Requisition section are   1. Buttons: First, Previous, Next, Last, Create Requisition, Ok, Print, View. Receive items 2. Fields: search bar 3. Drop down: Filter by Store, Date range, "..."(3 dots) 4. Radio-button: All, Pending, Completed, Cancelled, Withdrawn  5. Date Picker: From, To 6. Tooltip: Star figure |
| **9** | Creating and Verifying "Create Requisition" button | Pre condition: User should be logged in and it is on Inventory sub-module 1. Click on "Inventory Requisition" section 2. Click on "Create Requisition" button 3. Click on "Target Inventory " field and Select "GENERAL-INVENTORY" option 4. Click on "Item Category" drop down and select " Consumables" option 5. Enter "tissue" in ItemName field 6. Enter the "Required Quantity" field  7. Click on "Request" button 8. Click on "Close" icon | The requisition should be successfully created and saved. It should then be visible in the list of Inventory Requisition page with the details entered.  after Clciking on the "Request" button this successful message should pop up "success Requisition is Generated and Saved" |
| **10** | Verify to search the data by picking the date filter | Pre condition: User should be logged in and it is on Inventory Sub-module 1. Navigate to "Inventory Requisition" section 2. Click on the "From" date and select the "Jan 2022" date 3. Click on the "To" date and Select "Current date"  4. Click on "OK" button | The 'Date' column date must fall within the selected date |
| **11** | Verify data range by Select "one week" option from drop down | Pre condition: User should be logged in and it is on Inventory sub-module 1. Click on the data range button 2. select "one week" option from the drop down 3. Click on "OK" button | The 'Date' column date must fall within the selecetd date range of "one week" |
| **12** | Verify the presence of Filter by Store drop down | Pre condition: User should be logged in and it is on Inventory sub-module 1. Navigate to "Inventory Requisition" section 2. Select option ar per the choice | The 'Requested To' column data must fall within the selecetd option from the dropdown |
| **13** | Verify the presence of List by Requisition Status radio button | Pre condition: User should be logged in and it is on Inventory sub-module 1. Naviaget to "Inventory Requisition" section 2. Select "Pending" radio button 3. Select "Complete" radio button 4. Select "Canceled" radio button 5. Select "Withdrawn" radio button 6. Select "All" radio button | Ensure that all radio buttons in the "List by Requisition Status" field are functional and clickable. Verify that the "Status" column data updates correctly based on the selected radio button. |
| **14** | Verify that search field returns records matching the entered keyword. | Pre condition: User should be logged in and it is on Inventory sub-module 1. Naviagte to the "Inventory Requisition" section 2. Click on "all" radio button 3. Enter the 'pending' keyword in the search bar | Record should be present as per the entered keywords |
| **15** | Verify the presence of "View" button | Pre condition: User should be logged in and it is on Inventory sub-module 1. Navigate to " Inventory Requisition" section 2. Click on "View" button of the specfic record 3. Click on "Close" icon button | After Clicking "View" button, "REQUISITION DETAILS PRINT" page should pop up and by Clicking on close icon, user should navigate back to the Inventory Requisition page |
| **16** | Verify the presence of "..." (3 dots) drop down | Pre condition: User should be logged in and it is on Inventory sub-module 1. Navigate to the "Inventory Requisition" section 2. Click on "..." drop down of the specfic record 3. Select "create copy" from drop down | Ensure that records with a status of "Pending" can be copied.  The copied record should also display with a status of "Pending.  " After selecting "Create Copy" from the dropdown, the "Create Requisition" page should show the relevant fields  clicking the "Request" button should navigate the user back to the Inventory Requisition page. |
| **17** | Verify the presence of "Receive items" button | Pre condition: User should be logged in  1. Navigate to the "Inventory" module 2. Navigate to "Internal" sub-module  3. Navigate to "Requisition" section 4. Click on "Direct Dispatch" button 5. Enter all the required field  ( Store -> (it should match with current store of substore module), ItemCategory, ItemName , Dispatched Qty, Remark) 6. Click on the "Direct Dispatch" button 7. Click on "Close" icon  8. Navigate to "Substore" module and click on "Accounts" sub-modal 9. Navigate to "Inventory Requisition" section of "Inventory" sub-module  10. Click on "Receive item" of the specfic record 11. Click on the "Recevie" butotn 12. Click on "Back to Requisition List" button | Ensure that the selected store in the "Inventory" module matches the current substore modal.   Verify that the "Receive Item" button is highlighted.  Upon clicking the "Direct Dispatch" button, a success message should display: "Notice: Requisition is Generated and Saved."  After clicking the "Receive" button, the remarks field should disappear, and a success message should appear: "Success: Items Received Successfully. Stock Updated."  When clicking "Back to Requisition List," the "Receive Items" button should disappear. |
| **18** | Verify and creating the stock section of the "Inventory" module | Pre condition: User should be logged in 1. Navigate to the "Inventory" module 2. Navigate to "Internal" sub-module  3. Navigate to "Requisition" section 4. Click on "Direct Dispatch" button 5. Enter all the required field  ( Store -> (it should match with current store of substore module), ItemCategory, ItemName , Dispatched Qty, Remark) 6. Click on the "Direct Dispatch" button 7. Click on "Close" icon  8. Navigate to "Substore" module and click on "Accounts" sub-modal 9. Navigate to "Inventory Requisition" section of "Inventory" sub-module  10. Click on "Receive item" of the specfic record 11. Click on the "Recevie" butotn 12. Click on "Back to Requisition List" button 13. Navigate to the "Stock"section | Verify that the available quantity of items has increased in the stock. |
| **19** | Verify and creating the "Consumption" section of the "Inventory" module | Pre condition: User should be logged in and it is on Substore module 1. Naviagte to the "Consumption" section of the "Inventory" sub-module 2. Click on "New Consumption"  3. Enter all the required field  (Consumption Date ,Item Name,Consumed Qty) 4. Click on "Save" button | Upon clicking the "Save" button, a success message saying "Consumption completed" should be displayed, and the data should be accurately reflected in the record. |
| **20** | Verify and creating the "Reports" section of the "Inventory" module | Pre condition: User should be logged in and it is on Substore module 1. Naviagte to the "Reports" section of the "Inventory" sub-module 2. Navigate to "Consumption Report" sub-modal  3. Enter all the "From" and "To" field  4. Click on "Show Report" button | The 'Date' column data must align with the selected option from the dropdown, and it should display records added in the "Consumption" section for that specific date. |
| **21** | Verify and creating the "Return" section of the "Inventory" module | Pre condition: User should be logged in and it is on Inventory sub-module 1. Naviagte to the "Return" section 2. Click on "Return Items" buton 3. Enter all the required field  (Select Inventory, Item Name, ReturnQty) 4. Click on "Return" button | After clicking the "Return" button, a success message stating "Return is Generated and Saved" should be displayed. Ensure that the entered data is visible in the "Return" section of the "Inventory" sub-module. |
| **22** | Retrieve Comments for a Post and validate the response in method:    getCommentsForPost(int postId) | 1. Call the GET **https://jsonplaceholder.typicode.com**  **/posts/{postId}/comments** endpoint  2. Pass the postId as a parameter  3. Check if the response body contains an array of comments  4. Validate each comment has a postId, id, name, email, and body properties | Returns a 200 OK status with a list of comments related to the specified post.    Response body contains an array of comments with the specified properties |
| **23** | Delete a Post and validate the deletion in method:    deletePost(int id) | 1. Call the DELETE **https://jsonplaceholder.typicode.com**  **/posts/{id}** endpoint.  2. Pass the id of the post to delete.  3. Call GET /posts/{id} to verify the post no longer exists. | Returns a 200 OK status code if the post is successfully deleted.  The post should not be found in the subsequent GET request, returning 404 Not Found. |
| **24** | Update an existing post and validate the update in method:    updatePost(int id, String newTitle, String newBody, int userId) | 1. Call the PUT **https://jsonplaceholder.typicode.com**  **/posts/{id}** endpoint.  2. Provide newTitle, newBody, and userId in the request body.  3. Validate the response contains the updated title, body, and userId fields. | Returns a 200 OK status code with the updated post.  Response body contains the updated title, body, userId, and id. |
| **25** | Retrieve a post by ID and validate the response in method:    getPostById(int id) | 1. Call the GET **https://jsonplaceholder.typicode.com**  **/posts/{id}** endpoint.  2. Pass the id of the post to retrieve.  3. Validate the response body contains the correct post details. | Returns a 200 OK status code.  Response body contains the correct title, body, userId, and id. |

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. **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**

|  |
| --- |
| **A screenshot of a computer  Description automatically generated** |

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

|  |
| --- |
| A screenshot of a desktop  Description automatically generated |

In your project folder, you will find a batch file named git\_commands

|  |
| --- |
| A screenshot of a computer  Description automatically generated |

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

|  |
| --- |
| A screenshot of a computer  Description automatically generated |

===============================================================================

All the Best