

# YAKSHA HEALTH APP WITH TYPESCRIPT AND PLAYWRIGHT

## Usecase summary

**Project Name:** healthapp.yaksha app – Medical Record Management System

**Use Case Summary:** healthapp.yaksha is a healthcare application designed to manage Electronic Medical Records (EMR). It allows users to view, search, and manage patient records. It features functionality such as adding/editing patient records, filtering data by doctor and department, and exporting records. The primary use case is to automate the process of medical record management, ensuring efficient and reliable operations for healthcare providers.

**Technology Stack:**

- **Automation Tool:** Playwright (for testing)

**Key Features:**

- **Patient Record Management:** Add, edit, and delete patient records.
- **Filtering and Search:** Search medical records by date range, doctor, department, and more.
- **Export Functionality:** Export records for offline access.

**Expected Outcomes:**

- Automate key healthcare operations like patient record handling, filtering, and validation.
- Ensure the accurate retrieval and modification of medical records, enhancing operational efficiency.

## Overview of the application

Pages/Features that are to be focused for the application

Please use the Application URL <https://healthapp.yaksha.com>

## PROBLEM STATEMENT

Need to automate the following activities using playwright + typescript

You will be given a few Json files in the Data folder like testData.json.

Path	File	Description
src/data	<ul style="list-style-type: none"><li>• testData.json</li></ul>	<ol style="list-style-type: none"><li>1. Contains data to read from a json file.</li></ol>
src/pages	<ul style="list-style-type: none"><li>• AccountingPage</li><li>• AppointmentPage</li><li>• DispensaryPage</li><li>• LaboratoryPage</li><li>• LoginPage</li><li>• MaternityPage</li><li>• NursingPage</li></ul>	<ol style="list-style-type: none"><li>1. All core activities to be performed here.</li><li>2. The comments associated with each templated method here describe the expectation.</li><li>3. Declare any</li></ol>

	<ul style="list-style-type: none"> <li>● OperationTheaterPage</li> <li>● ProcurementPage</li> <li>● SettingsPage</li> </ul>	<p>variable/object you need to share data/status between different methods.</p> <p>4. Do not modify the signature of methods declared here.</p>
--	-----------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------

Here's a detailed table format for the test cases to be tested

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
	Verify Login with Valid Credentials	<p>1.The application should read the data from testData.json file and fetch the user's name and password.</p> <p>2. It should call the method performLogin()</p> <p>3. The login method will perform authentication with the username and password.</p> <p>4. Verify admin name is visible on the home page.</p> <p><b>It is necessary to implement this login functionality at first and then implement any other test case.</b></p>	<p><b>Reference path</b></p> <p>\src\Pages\LoginPage</p> <p><b>methods</b></p> <p>performLogin()</p>	Successfully logs in with provided credentials. The user is logged in the admin page.
1	Verify Maternity Allowance Payment Receipt Modal	<p>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Maternity/PatientList">https://healthapp.yaksha.com/Home/Index#/Maternity/PatientList</a></p> <p>2. Click on the Reports submodule.</p> <p>3. Select Maternity Allowance Report, enter the From Date as 01-01-2020, and click the OK button.</p> <p>4. From the table, click on View Details for the first record.</p> <p>5. Verify that the "Maternity Allowance Payment Receipt" modal appears.</p>	<p><b>Reference path</b></p> <p>\src\Pages\MaternityPage</p> <p><b>methods</b></p> <p>verifyReceiptModal()</p>	<p>The Reports submodule should open successfully. The Maternity Allowance Report should be displayed with data filtered based on the entered date. Clicking View Details should open the "Maternity Allowance Payment Receipt" modal without any errors.</p>

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
2	Verify Error Message When Editing an Occupied Bed	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard">https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard</a></li> <li>2. Click on the Manage Bed tab.</li> <li>3. Locate any occupied bed and click the Edit button.</li> <li>4. Verify that an error message "Cannot modify occupied beds." is displayed.</li> </ol>	<b>Reference path</b> \src\Pages\SettingsPage  <b>methods</b> verifyOccupiedBed()	Clicking Edit on an occupied bed should not allow modifications and should display the error message "Cannot modify occupied beds."
3	Verify Male Ward Bed Count Updates After Deactivating a Bed	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation">https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation</a></li> <li>2. Note down the Male Ward Bed Count from the table.</li> <li>3. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard">https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard</a></li> <li>4. Click on the Manage Bed tab.</li> <li>5. Locate an available bed in the Male Ward and click on the Edit button and update the status to available by unchecking the checkbox and save it</li> <li>6. If no bed is available, please click on Edit on any N/A and update the status to available by checking the checkbox and save it.</li> <li>7. Navigate back to <a href="https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation">https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation</a></li> <li>8. Verify that the Male Ward Bed Count is updated by 1.</li> </ol>	<b>Reference path</b> \src\Pages\SettingsPage  <b>methods</b> verifyMaleWard()	The initial Male Ward Bed Count should be recorded successfully. The system should allow editing of an available bed. Unchecking "Is Active" should deactivate the selected bed. Upon revisiting the Bed Information page, the Male Ward Bed Count should be 1 less than the initially recorded count.
4	Verify Add New Patient modal opens with Alt + Enter keyboard shortcut	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Dispensary/ActivateCounter">https://healthapp.yaksha.com/Home/Index#/Dispensary/ActivateCounter</a></li> <li>2. Activate any counter.</li> <li>3. Select the Sale tab and press Alt + N.</li> </ol>	<b>Reference path</b> \src\Pages\DispensaryPage  <b>methods</b> addNewPatientUsingShortcut()	Add New Patient modal should open.

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
5	Verify Remarks text field placeholder value	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/OperationTheatre/OtBookingList">https://healthapp.yaksha.com/Home/Index#/OperationTheatre/OtBookingList</a></li> <li>2. Click on the new OT booking button.</li> <li>3. Verify Remarks Text area is enable</li> <li>4. Verify the placeholder name of the Remarks text area.</li> </ol>	<b>Reference path</b> \src\Pages\OperationTheaterPage  <b>methods</b> verifyRemarksPlaceholder()	Verify the placeholder name of the Remarks text area.
6	Capture screenshot of Inventory Requisition section	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard">https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard</a></li> <li>2. Click on the Manage Bed tab.</li> <li>3. Click on the Add Bed button.</li> <li>4. Enter the required details to create a new Male Ward bed.</li> <li>5. Unselect the "Is Active" checkbox.</li> <li>6. Click Save to create the bed.</li> <li>7. Use the Search Box to search for the newly created bed.</li> <li>8. Verify that the bed appears in the list with inactive status.</li> </ol>	<b>Reference path</b> \src\Pages\SettingsPage  <b>methods</b> captureInventoryRequisitionSection()	The newly created bed should be displayed in the Manage Bed list. The bed should be marked as inactive (Is Active = False).
7	Verify Daily Transactions Report for Fiscal Year 2023	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Accounting/Reports">https://healthapp.yaksha.com/Home/Index#/Accounting/Reports</a></li> <li>2. Select Daily Transactions Report from the available reports.</li> <li>3. Choose 2023 as the Fiscal Year.</li> <li>4. Click on the Load button.</li> </ol>	<b>Reference path</b> \src\Pages\AccountingPage  <b>methods</b> verifyTable()	Clicking on Load should populate the table with relevant transaction data. The table should not be empty if records exist for 2023.
8	Verify Appointment Status Change After Check-in and Invoice Print	<ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Appointment/ListAppointment">https://healthapp.yaksha.com/Home/Index#/Appointment/ListAppointment</a></li> <li>2. Select counter i.e NEW-1 and then navigate to link.</li> <li>2. Select "New Patient" from the dropdown.</li> <li>3. Select "All Doctors" option from the Doctor field.</li> </ol>	<b>Reference path</b> \src\Pages\AppointmentPage  <b>methods</b> verifyAppointmentStatusChangeAfterInvoicePrint()	Upon searching for the same patient again using the saved name and app id, the appointment status should update to "CheckedIn".

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
		<p>4. Click on the "FROM" date picker and select 1st January 2020.</p> <p>5. Click on the "Show Patient" button.</p> <p>6. Locate any patient having "Check-in" button and save its "Name" and "App. Id" for future use. Click on "Check-in" button.</p> <p>7. In the modal that appears, select Religion as Others, Department as Cardiology, and Doctor field as Dr Pooja Mishra.</p> <p>8. Click on Print Invoice.</p> <p>A confirmation modal "Are you sure you want to Print Invoice?" should appear, click Confirm.</p> <p>9. If a pop-up comes with heading "we have found existing patients with similar information". Click on the "Register as new patient" button.</p> <p>10. A new pop-up modal should appear. Please close the invoice modal by clicking on X.</p> <p>11. Navigate to Appointment Booking List tab.</p> <p>12. Select "New Patient" from the dropdown.</p> <p>13. Select "All Doctors" option from the Doctor field.</p> <p>14. Select the FROM date as 1<sup>st</sup> January 2020 and search for the same patient and check its status.</p>		
9	Verify Adding Triage for an Outpatient	<p>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient">https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient</a></p> <p>2. Locate the same patient used in the previous test cases.</p> <p>3. Select the checkbox for that patient.</p> <p>4. Click on the Triage button.</p> <p>5. Enter "Temp value" in Chief</p>	<p><b>Reference path</b></p> <p>\src\Pages\NursingPage</p> <p><b>methods</b></p> <p>verifyAddingTriage()</p>	A success confirmation popup with the message: " <b>Chief Complaint Added Successfully</b> " should appear.

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
		Complaint field. 6. Click on Add Triage.		
10	Verify Nursing Check-in for an Outpatient	1. Perform all points of TC#8 and then navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient">https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient</a> 2. Select the checkbox for same patient used in above TC and click on checkin button. 3. Nursing Checkin Details modal opens and click on save button.	<b>Reference path</b>  \src\Pages\AppointmentPage  \src\Pages\NursingPage  <b>methods</b>  verifyAppointmentStatusChangeAfterInvoicePrint()  verifyNursingCheckinForOutPatient()	A success confirmation popup with the message: " <b>Nursing CheckIn Added Successfully</b> " should appear. The status should also change from initiated to Checkedin for that same patient.
11	Verify 'Morning Counter' selection and report generation for the specified date	1. Navigate to the dispensary module. 2. Click on Reports. 3. Click on the User Collection report to navigate to the respective report section. 4. Read the fromDate from testData.json file and enter in From Date input field. 5. Select the morning counter from the dropdown. 6. Click on the Show Report button to generate the report. 7. Verify the generated report correctly displays the morning counter in the counter column.	<b>Reference path</b>  \src\Pages\DispensaryPage  <b>methods</b>  generateMorningCounterReport()	Verify the generated report correctly displays the morning counter in the counter column.
12	Verify the tooltip text on hover of Star icon in Laboratory	1. Navigate to Laboratory module. 2. Hovers over star icon and waits for tooltip to appear. 3. Verifies the visibility	<b>Reference path</b>  \src\Pages\LaboratoryPage	Verifies the visibility of the star icon and retrieves the tooltip text. Result should be "Remember this date".

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
		of the star icon and retrieves the tooltip text.	<b>methods</b> verifyStarTooltip()	
13	Add and Verify New Imaging Type in Radiology	<ol style="list-style-type: none"> <li>1. Navigates to the Settings module and clicks on the Radiology submodule.</li> <li>2. Clicks on the "Add Imaging Type" button to open the modal for adding a new imaging type.</li> <li>3. Fills the "Imaging Item Name" field with a random name (Test-{random4digitnumber}) and clicks "Add".</li> <li>4. Verifies that the newly added imaging type appears in the list of imaging types.</li> </ol>	<b>Reference path</b>  \src\Pages\SettingsPage  <b>methods</b>  addAndVerifyNewImagingType()	Verifies that the newly added imaging type appears in the list of imaging types.
14	Web Element Handling for Dropdowns in Purchase Request	<ol style="list-style-type: none"> <li>1. Navigate to Purchase Request List.</li> <li>2. Fetch the fromDate and toDate from testData.json file.</li> <li>3. Apply date range filter.</li> <li>4. Retrieve dates from the "Requested Date" column and validate each date within range.</li> <li>5. Wait for success or failure.</li> </ol>	<b>Reference path</b>  \src\Pages\ProcurementPage  <b>methods</b>  verifyRequestedDateColumnDateWithinRange()	It should be able to retrieve dates that are within range.
15	Verify logout functionality from Admin dropdown	<ol style="list-style-type: none"> <li>1. Navigate to index page i.e <a href="https://healthapp.yaksha.com/Home/Index#/">https://healthapp.yaksha.com/Home/Index#/</a></li> <li>2. Click on the "Admin" dropdown</li> <li>3. Click on "Log Out" option.</li> <li>4. Verify the user is redirected to the login page.</li> </ol>	<b>Reference path</b>  \src\Pages\LoginPage  <b>methods</b> verifyLogoutFunctionality()	User is logged out successfully and the login page is displayed.



Students will gain experience in building strongly-typed applications and managing data flow with **TypeScript**. They'll learn how to define interfaces, use types for error prevention, and improve code maintainability.

With **Playwright**, learners will learn to write and execute automated tests for the <https://healthapp.yaksha.com>

pp. Key skills include:

- **Browser Automation:** Interacting with web elements and testing multiple browsers.
- **Assertions & Validations:** Ensuring app behaviour meets expected results.
- **End-to-End Testing:** Automating real user interactions and validating overall app functionality.

## IMPLEMENTATION/FUNCTIONAL REQUIREMENT

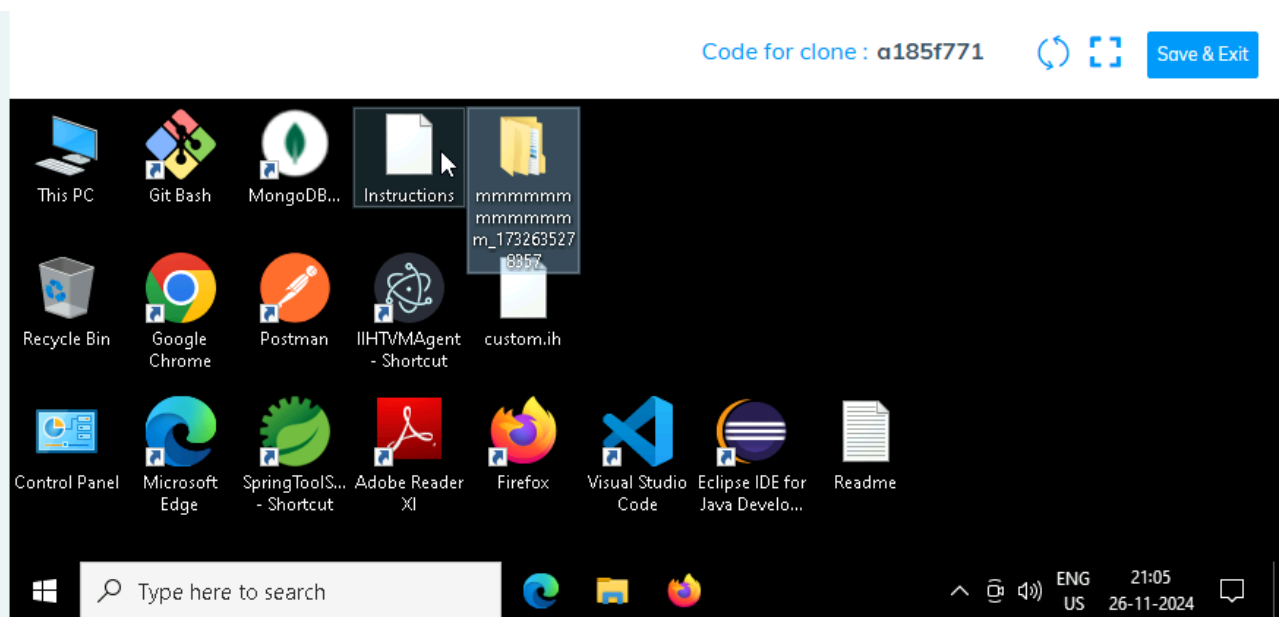
## 1.1 CODE QUALITY/OPTIMIZATIONS

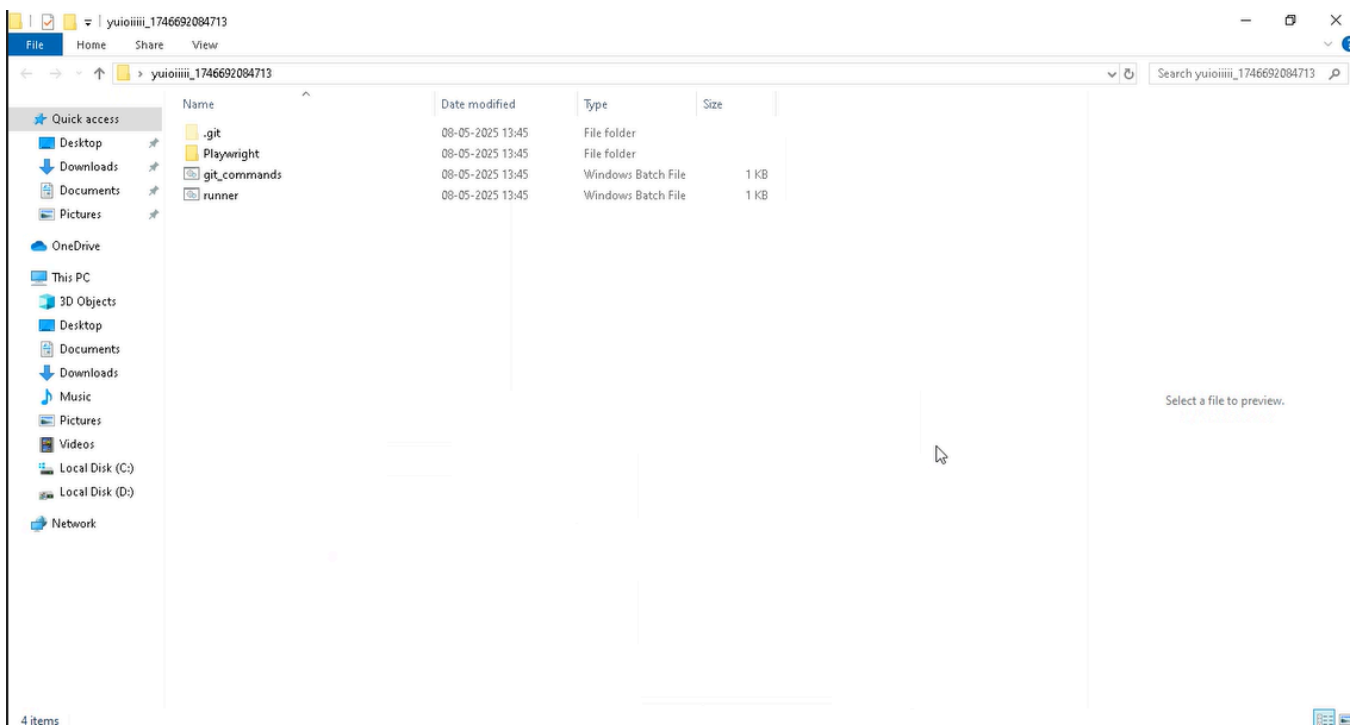
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

### Execution Steps:

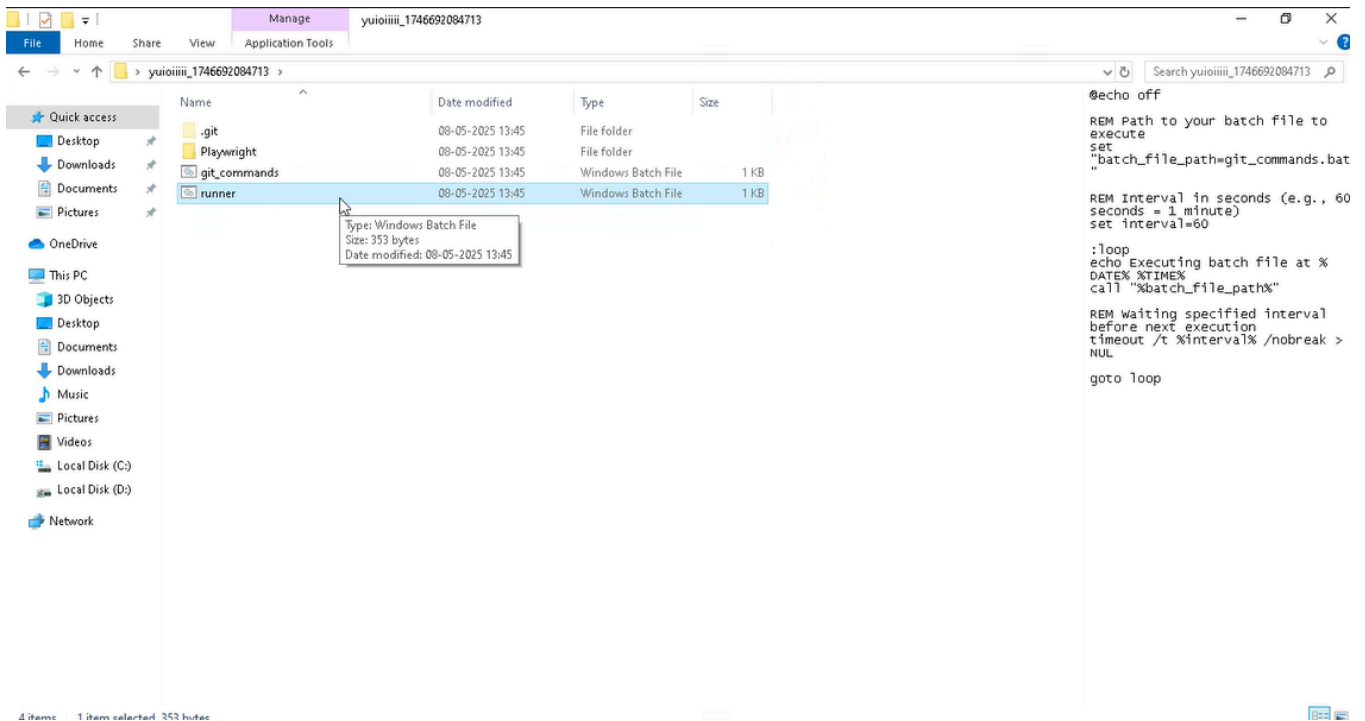
### Steps for Execution:

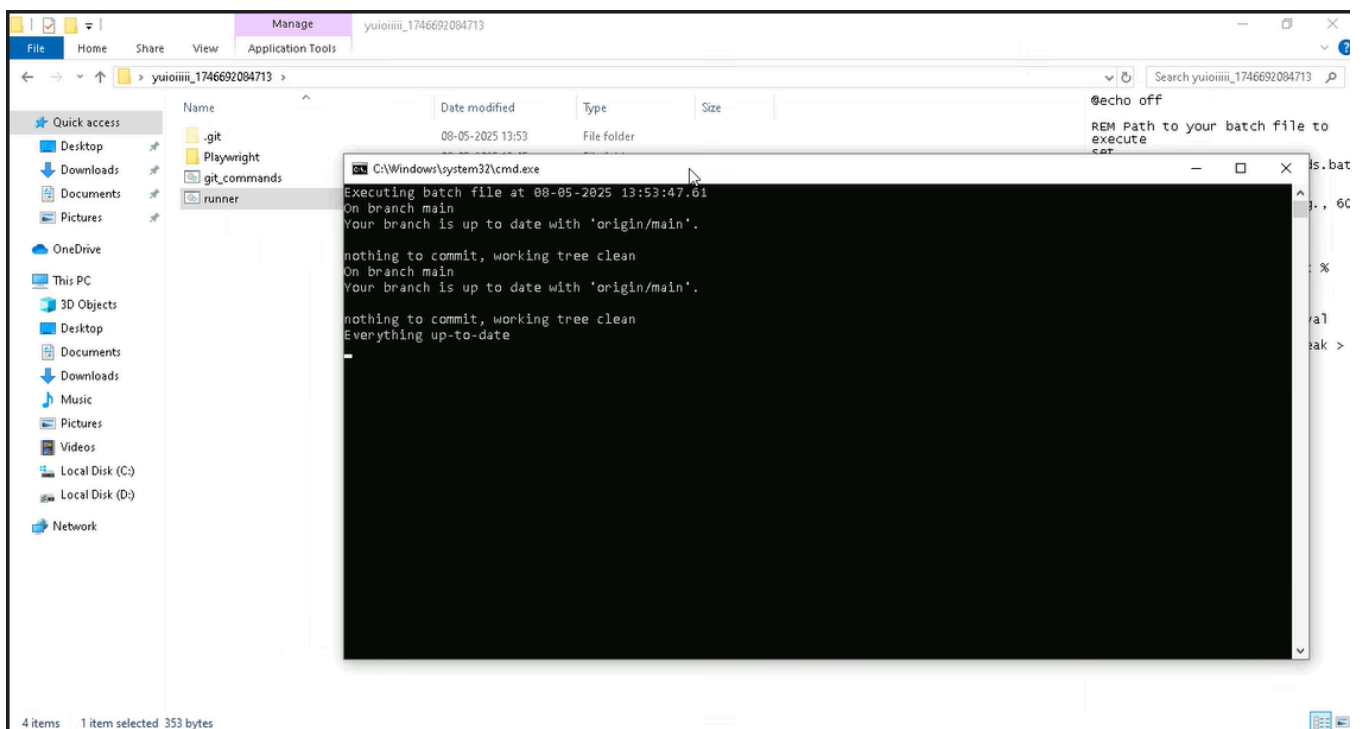
1. Please open the folder created on the desktop with the email name you used to login.



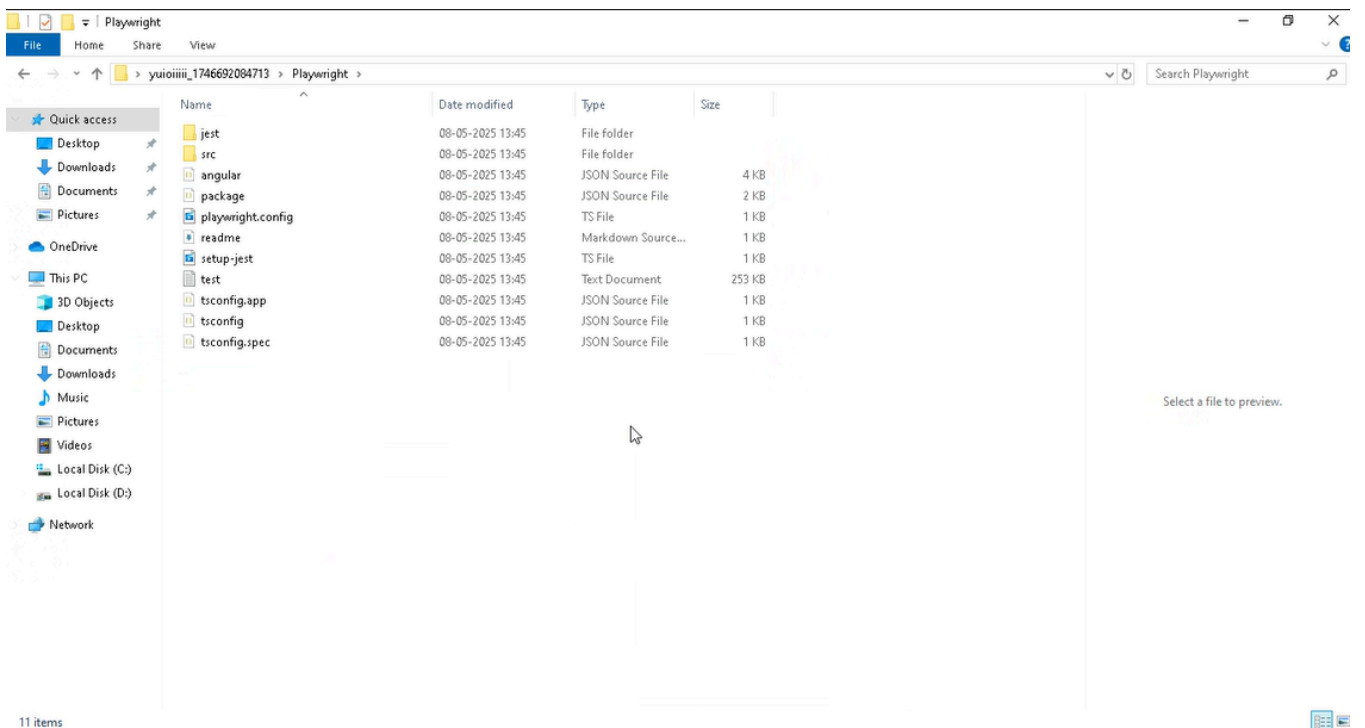


## 2. Execute this “runner” file. This will keep pushing the code at regular intervals.

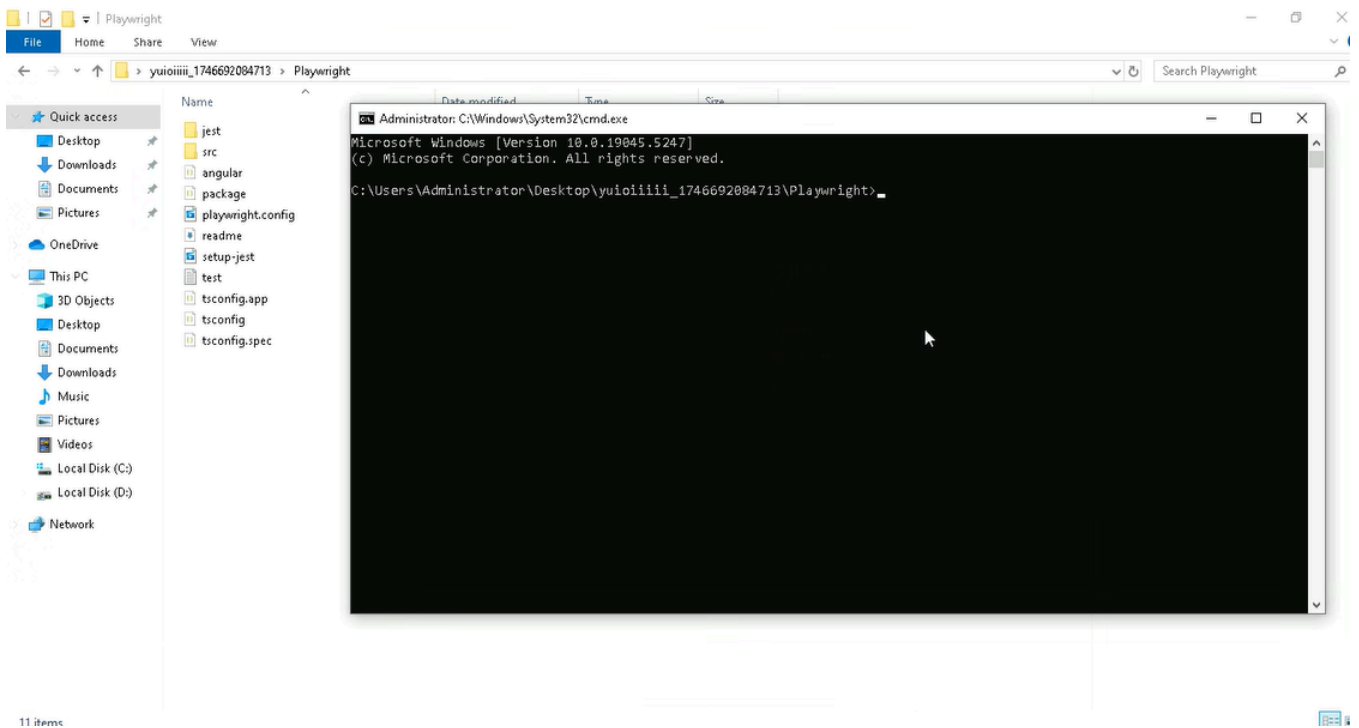
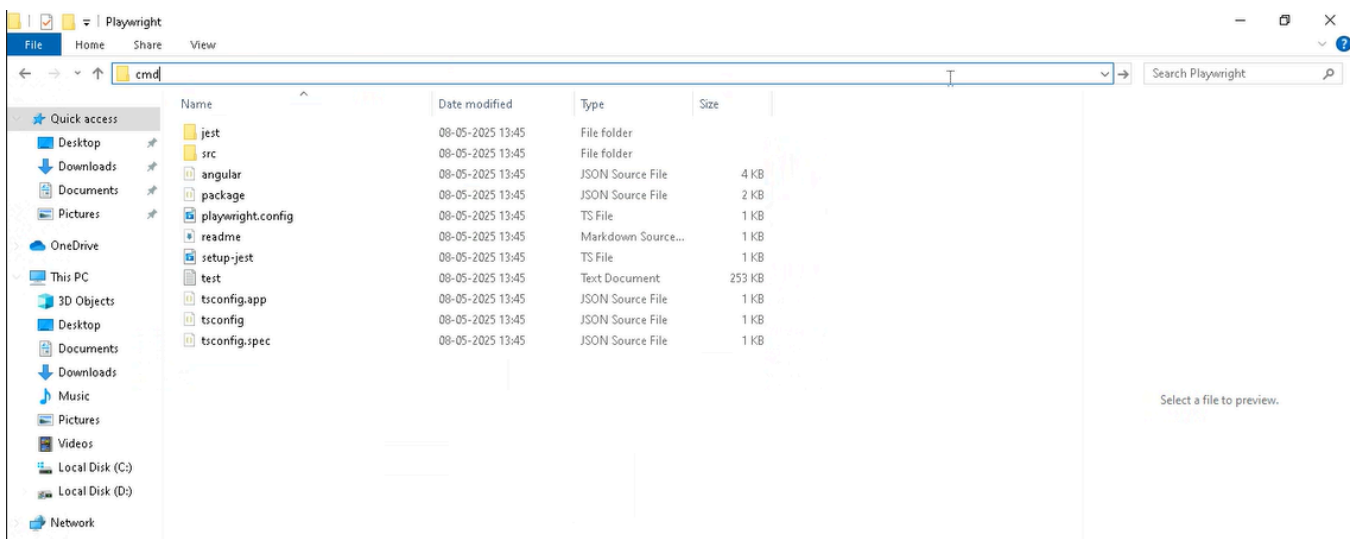




### 3. Go into the Playwright folder

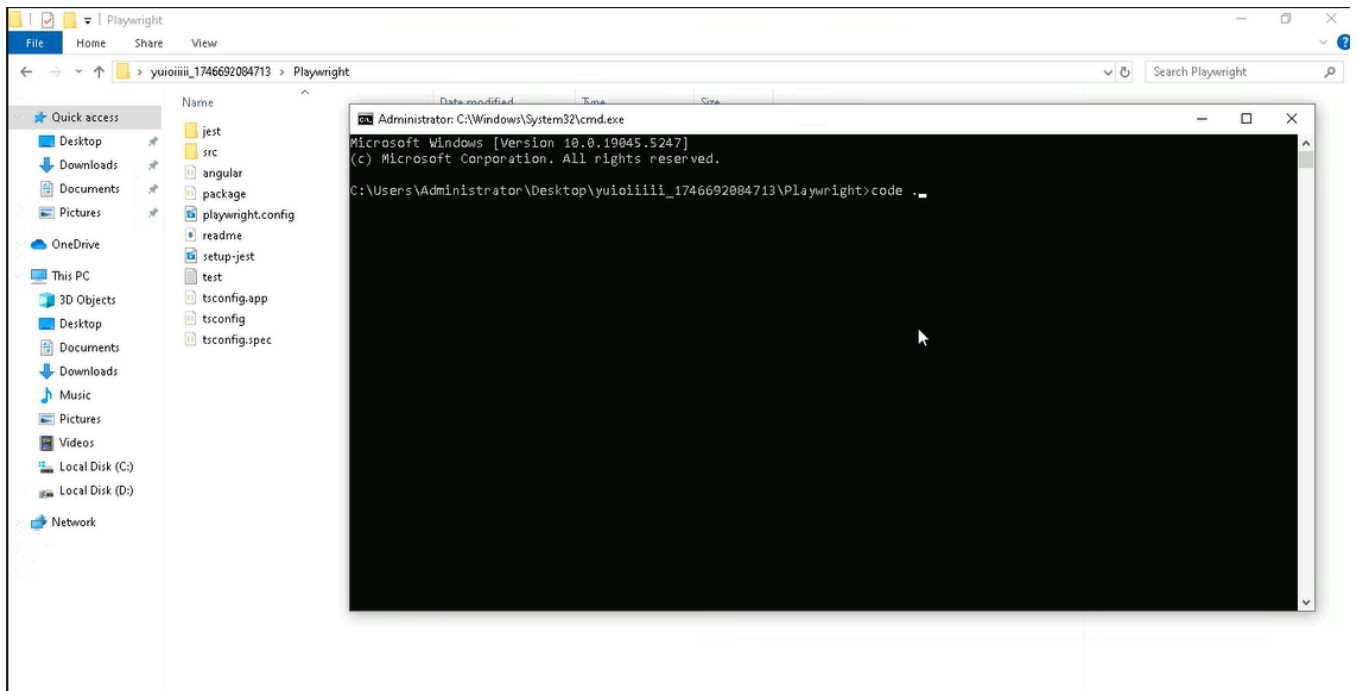


### 4. Open command prompt with its location using "cmd" in search bar

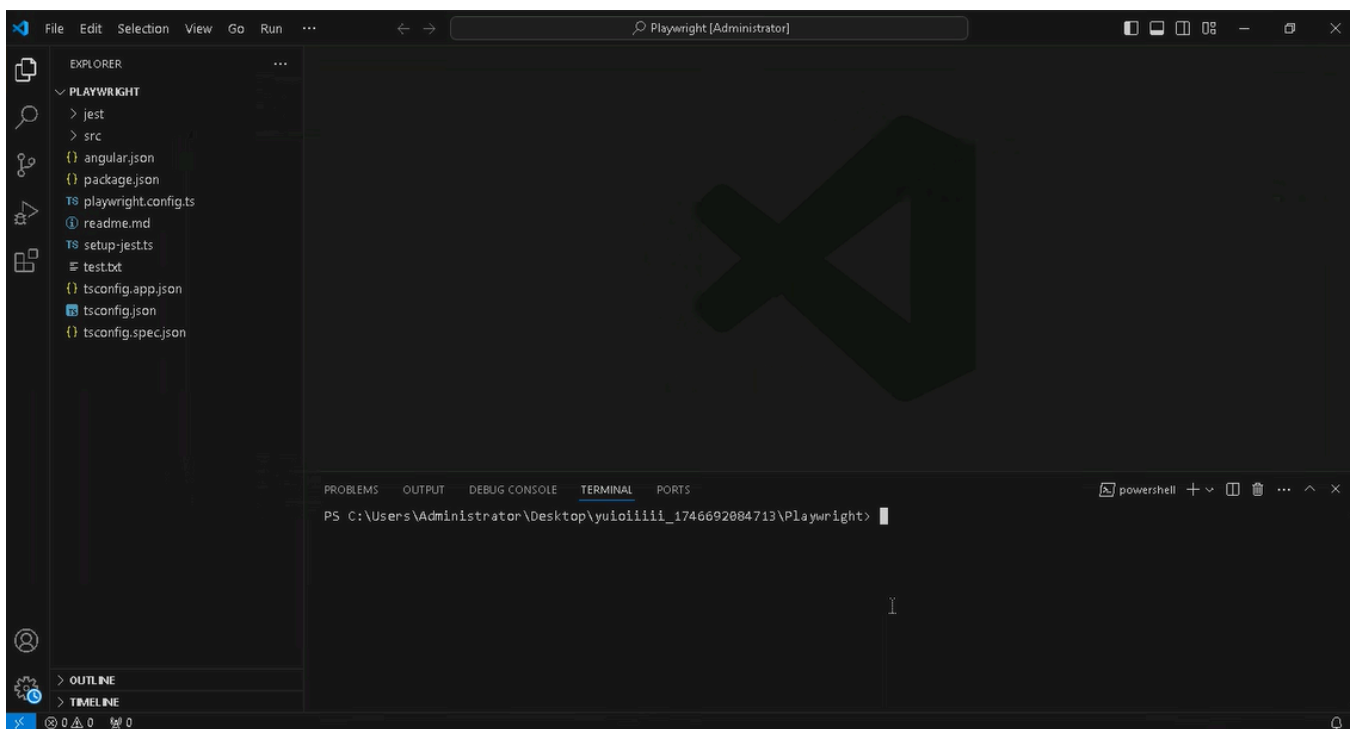


5. Open VS Code through cmd using below command:

code .



**6. Once VsCode is open. Please open the terminal in Playwright folder:**



1. Install all dependencies in the Playwright folder path using:

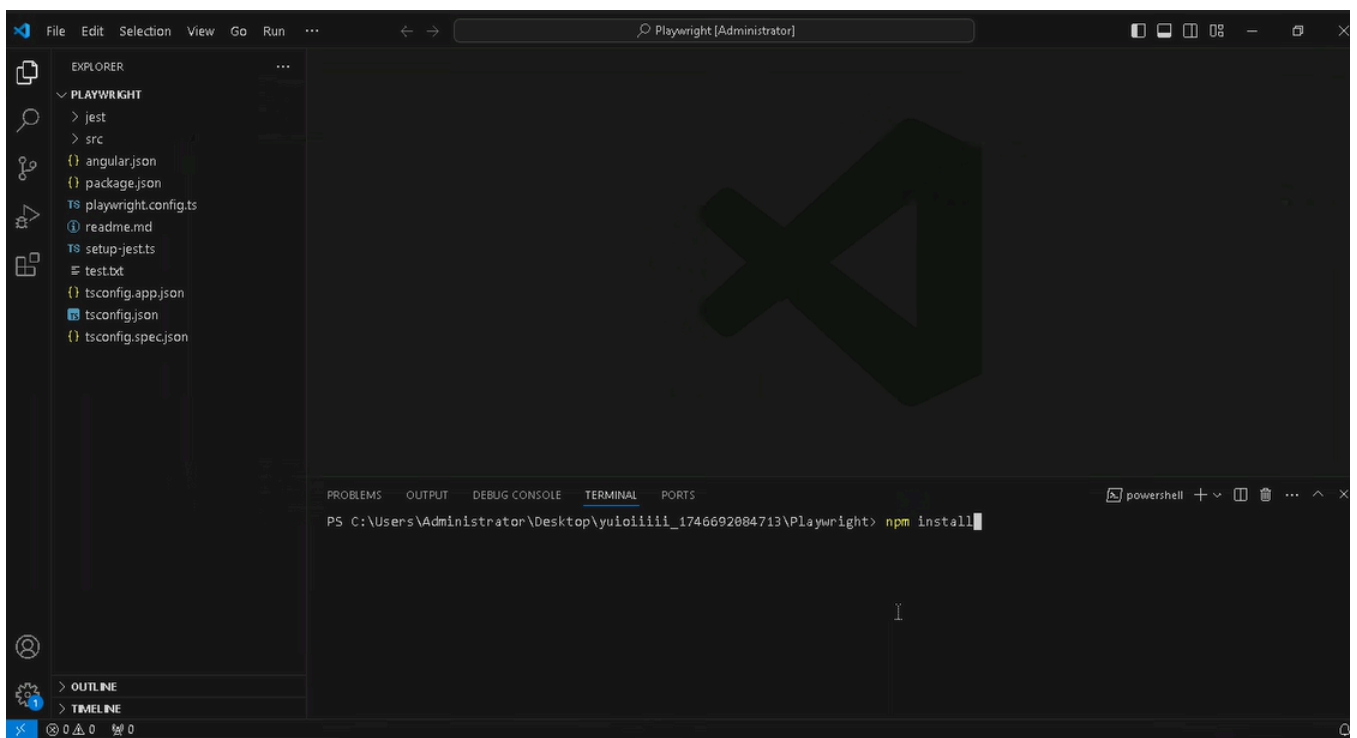
**npm install**

2. Install playwright in the Playwright folder path:

**npx playwright install**

3. Run the Tests in the Playwright folder path:

**npx playwright test ./src/tests/PL2\_testcases/yaksha.spec.ts**



Similarly you can execute other commands in the terminal.