

# YAKSHA HEALTH APP WITH JAVASCRIPT AND CYPRESS

## 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:** Cypress (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 cypress+javascript

You will be given few Json files in e2e/Data folder which you have to use like

PatientName.json, ValidLogin.json, appointment.json, dispensary.json, doctor.json, lab.json, maternity.json, medicalRecord.json, nursing.json, pharmacy.json, procurement.json, radiology.json, settings.json, subStore.json and testData.json.

| Path      | File   | Description                              |
|-----------|--|--|
| /e2e/Data | <ul style="list-style-type: none"><li>• PatientName.json</li><li>• ValidLogin.json</li><li>• appointment.json</li><li>• dispensary.json</li><li>• doctor.json</li><li>• lab.json</li></ul> | 1. Contains data to read from json file. |

Mymedic automation using cypress

|                   |   |   |
|-------------------|---|---|
|                   | <ul style="list-style-type: none"> <li>● <b>maternity.json</b></li> <li>● <b>medicalRecord.json</b></li> <li>● <b>nursing.json</b></li> <li>● <b>pharmacy.json</b></li> <li>● <b>procurement.json</b></li> <li>● <b>radiology.json</b></li> <li>● <b>settings.json</b></li> <li>● <b>subStore.json</b></li> <li>● <b>testData.json</b></li> </ul>         |   |
| PageObjects/Pages | <ul style="list-style-type: none"> <li>● AccountingPage</li> <li>● AdtPage</li> <li>● CustomPage</li> <li>● DispensaryPage</li> <li>● LabPage</li> <li>● LoginPage</li> <li>● MaternityPage</li> <li>● NursingPage</li> <li>● PharmacyPage</li> <li>● ProcurementPage</li> <li>● SettingsPage</li> <li>● SubStorePage</li> <li>● UtilitiesPage</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 variable/object you need to share data/status between different methods.</li> <li>4. Do not modify the signature of methods declared here.</li> </ol> |

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 | 1.The application should read the data from login.json file and fetch the user's name and password.<br>2. It should call the method performLogin()<br>3. Perform login method will perform authentication with the username and password.<br>4. Verify admin name is visible on the home page. | <b>Reference path</b><br><br>\src\Pages\LoginPage<br><br><b>methods</b><br><br>performLogin() | Successfully logs in with provided credentials. The user is logged in the admin page. |

Mymedic automation using cypress

| Test Case No. | Test Case Name   | Test Steps to be performed  | Path & Method Used  | Expected Result   |
|---------------|--|---|---|---|
|               |  | <b>It is must to implement this login functionality at first and then implement any other test case.</b>  |   |   |
| 1             | Verifies tooltip message is displayed on dispensary signout button | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Dispensary">https://healthapp.yaksha.com/Home/Index#/Dispensary</a><br>2. Navigate to the Dispensary link in the application.<br>3. Hover over the SignOut button<br>4. Verify the presence of a tooltip text when hovering over the SignOut button.   | <b>Reference path</b><br>\src\Pages\DispensaryPage<br><br><b>methods</b><br>hoveroverMainDispensary()                     | The method verifies that an tooltip occurs over the signout button and it is "You are currently in Main Dispensary dispensary. To change, you can always click here." |
| 2             | Verify Navigation to Patient Overview from Past Days Records       | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient">https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient</a><br>2. Click on the Past Days tab.<br>3. Enter the From Date as 01-01-2020 and click the OK button.<br>4. Search for the patient "Deepika Rani" in the list.<br>5. Locate the patient's record and click on Overview from the Actions column.   | <b>Reference path</b><br>\src\Pages\NursingPage<br><br><b>methods</b><br>verifyPatientOverviewNavigationFromPastRecords() | Clicking on Overview should navigate the user to the Patient Overview page successfully.  |
| 3             | Verify File Upload for a Past Patient Record                       | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient">https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient</a><br>2. Click on the Past Days tab.<br>3. Enter the From Date as 01-01-2020 and click the OK button.<br>4. Search for the patient "Deepika Rani" in the list.<br>5. Locate the patient's record and click on "Upload files" from the Actions column.<br>6. Verify that the Upload Files modal opens.<br>7. Select the Department as "Pathology".<br>8. Upload an image file.<br>9. Click on the Submit button. | <b>Reference path</b><br>\src\Pages\NursingPage<br><br><b>methods</b><br>verifyFileUpload()                               | A success confirmation popup with the message: "File Uploded" should appear.  |

| Test Case No. | Test Case Name  | Test Steps to be performed   | Path & Method Used  | Expected Result   |
|---------------|---|--|---|---|
| 4             | Verify Activation of BANK A/C # Ledger  | <ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Accounting/Transaction/VoucherEntry">https://healthapp.yaksha.com/Home/Index#/Accounting/Transaction/VoucherEntry</a></li> <li>2. Click on the Settings tab.</li> <li>3. Select Ledger from the available options.</li> <li>4. Search for "BANK A/C #" in the ledger list.</li> <li>5. Locate the ledger and click on Activate from the Actions column.</li> <li>6. Accept the confirmation alert "Are you Sure want to Activate BANK A/C # ?".</li> </ol>   | <p><b>Reference path</b></p> <p>\src\Pages\AccountingPage</p> <p><b>methods</b></p> <p>verifyBankActivation()</p>   | A success confirmation popup with the message: "BANK A/C # is now activated." should appear.    |
| 5             | Verify Deactivation of BANK A/C # Ledger  | <ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Accounting/Transaction/VoucherEntry">https://healthapp.yaksha.com/Home/Index#/Accounting/Transaction/VoucherEntry</a></li> <li>2. Click on the Settings tab.</li> <li>3. Select Ledger from the available options.</li> <li>4. Search for "BANK A/C #" in the ledger list.</li> <li>5. Locate the ledger and click on Activate from the Actions column.</li> <li>6. Accept the confirmation alert "Are you Sure want to Deactivate BANK A/C # ?".</li> </ol> | <p><b>Reference path</b></p> <p>\src\Pages\AccountingPage</p> <p><b>methods</b></p> <p>verifyBankDeactivation()</p> | A success confirmation popup with the message: "BANK A/C # is now Deactivated.." should appear. |
| 6             | Verify all sub-modules are displayed correctly after Clicking on the "SubStore" Module. | <ol style="list-style-type: none"> <li>1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/WardSupply">https://healthapp.yaksha.com/Home/Index#/WardSupply</a></li> <li>3. "Select your Substore" pop up</li> <li>4. Click on "Account" sub-modal</li> <li>5. Click on the "Inventory" sub-module</li> <li>6. Click on "Pharmacy" sub-module</li> </ol>   | <p><b>Reference path</b></p> <p>\src\Pages\SubStorePage</p> <p><b>methods</b></p> <p>verifySubModulesVisible()</p>  | All sub-modules should be displayed correctly. Expected Sub modules are : Pharmacy, Inventory   |
| 7             | Verify the tooltip and it's text present on hover the                                   | <p>Pre condition: User should be logged in and it is on Maternity module</p> <ol style="list-style-type: none"> <li>1. Navigate to "Maternity list"</li> </ol>   | <p><b>Reference path</b></p> <p>\src\Pages\MaternityPage</p> <p><b>methods</b></p>                                  | Tooltip should be present when hover the mouse on star with text "Remember this date"           |

| Test Case No. | Test Case Name                          | Test Steps to be performed  | Path & Method Used   | Expected Result  |
|---------------|---|---|--|--|
|               | mouse on "Star"                         | sub-module<br>2. Hover the mouse in star  | verifyTooltipTextIsVisible()   |  |
| 8             | Verify Request for Quotation Generation | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/ProcurementMain/PurchaseRequest/PurchaseRequestList">https://healthapp.yaksha.com/Home/Index#/ProcurementMain/PurchaseRequest/PurchaseRequestList</a><br>2. Click on the Quotation tab.<br>3. Click on the Request for Quotation button.<br>4. Enter the following data in the respective fields:<br>Subject: test subject<br>Description: test description<br>Select Vendor: Ashar & Company<br>Item Name: Soap<br>Quantity: 2<br>5. Click on the Request button. | <b>Reference path</b><br><br>\src\Pages\ProcurementPage<br><br><b>methods</b><br><br>verifyRequestForQuotationGeneration() | A success confirmation popup with the message: "Request For Quotation is Generated and Saved" should appear. |
| 9             | Verify table filtering for "Male Ward"  | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Lab/Dashboard">https://healthapp.yaksha.com/Home/Index#/Lab/Dashboard</a><br>2. Select the Sample Collections tab.<br>3. Enter From Date as 01-01-2020 and click OK.<br>4. Hover over the Requesting Department column and click Hamburger Menu.<br>5. Select Starts with from the dropdown.<br>6. Enter Male Ward in the text field.  | <b>Reference path</b><br><br>\src\Pages\LabPage<br><br><b>methods</b><br><br>verifyTableFilteringForMaleWard()             | Table should display only rows with "Male Ward" in the Item column.  |
| 10            | Verify to export the order section data | 1. Navigate to <a href="https://healthapp.yaksha.com/Home/Index#/Pharmacy/Dashboard">https://healthapp.yaksha.com/Home/Index#/Pharmacy/Dashboard</a><br>2. Click on order section<br>3. Click on "Export" button  | <b>Reference path</b><br><br>\src\Pages\PharmacyPage<br><br><b>methods</b><br><br>verifyExportAndDownload()                | User should able to export the order section data  |
| 11            | Verify Warning Popup for                | 1. Navigate to the Utilities module and select the "Scheme Refund" tab.   | <b>Reference path</b><br><br>\src\Pages\UtilitiesPage  | A warning popup should appear with the message: "Please fill all the mandatory fields."                      |

| Test Case No. | Test Case Name                                       | Test Steps to be performed  | Path & Method Used   | Expected Result  |
|---------------|--|---|--|--|
|               | Mandatory Fields in Scheme Refund                    | 2. If required, please select any counter value and then select the "Scheme Refund" tab.<br>3. Click on the "New scheme Refund Entry" button.<br>Now click on save without entering value in any field.   | <b>methods</b><br>verifyWarningPopupForMandatoryFiels()  |  |
| 12            | Verify Price Category Enable/Disable                 | 1. Navigate to the "Settings" module.<br>2. Click on more... and select the "Price Category" tab.<br>3. Click on the "Disable" button to disable any Code in the table.<br>4. Verify a success message appears with the message "Deactivated."<br>Activate the same code by clicking the "Activate" button and verify the success message as "Activated". | <b>Reference path</b><br><br>\src\Pages\CustomPage<br><br>\src\Pages\SettingsPage<br><br><b>methods</b><br><br>navigateToCustomUrl()<br><br>verifyPriceCategoryEnableDisable() | A success message is displayed for both actions: "Deactivated." for disabling and "Activated." for enabling. |
| 13            | Verify Navigation Between Different Tabs             | 1. Navigate to the "SubStore" module.<br>2. Click on the "Accounts" button.<br>3. Select the "Inventory" tab.<br>4. Click and navigate between different tabs like "Stock", "Inventory Requisition", "Consumption", "Reports", "Patient Consumption" and "Return".  | <b>Reference path</b><br><br>\src\Pages\SubStorePage<br><br><b>methods</b><br><br>verifyNavigation()   | Ensure that it should navigate to each sub tab of the "Inventory" tab.                                       |
| 14            | Capture screenshot of Inventory Requisition section. | 1. SNavigate to "SubStore" module<br>2. Click on "Accounts" button.<br>3. Select "Inventory" tab and then "Inventory Requisition" sub tab.<br>4. Capture a screenshot of the page and save it in the Screenshots folder.  | <b>Reference path</b><br><br>\src\Pages\SubStorePage<br><br><b>methods</b><br>captureScreenshotOfInventoryRequisitionSection()   | Screenshot of the page is captured and saved successfully.   |

| Test Case No. | Test Case Name   | Test Steps to be performed   | Path & Method Used  | Expected Result   |
|---------------|--|--|---|---|
| 15            | Verify to navigate to each section which are present in the "Inventory" sub-module | <ol style="list-style-type: none"> <li>1. Navigate to ADT module.</li> <li>2. Click on "Admitted Patients".</li> <li>3. Search for any patient name which data to be read from PatientName.json file.</li> <li>4. Click on "..." button from table and select "Change Doctor".</li> <li>5. Change doctor modal will open and then click on update button without filling any value.</li> </ol> | <b>Reference path</b><br><br>\src\Pages\ADTPage<br><br><b>methods</b><br>verifyInventorySubModuleNavigation() | Verify a field level error message appears "Select doctor from the list." |

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

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

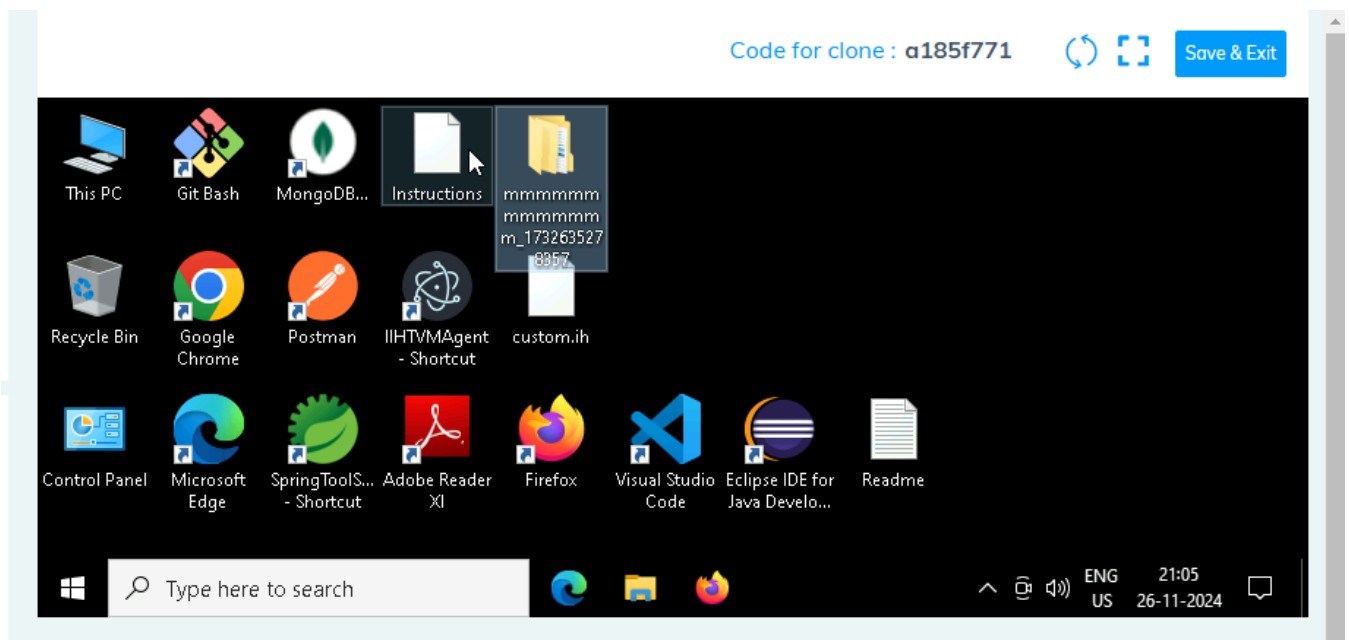
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

#### Execution Steps:

#### Steps for Execution:

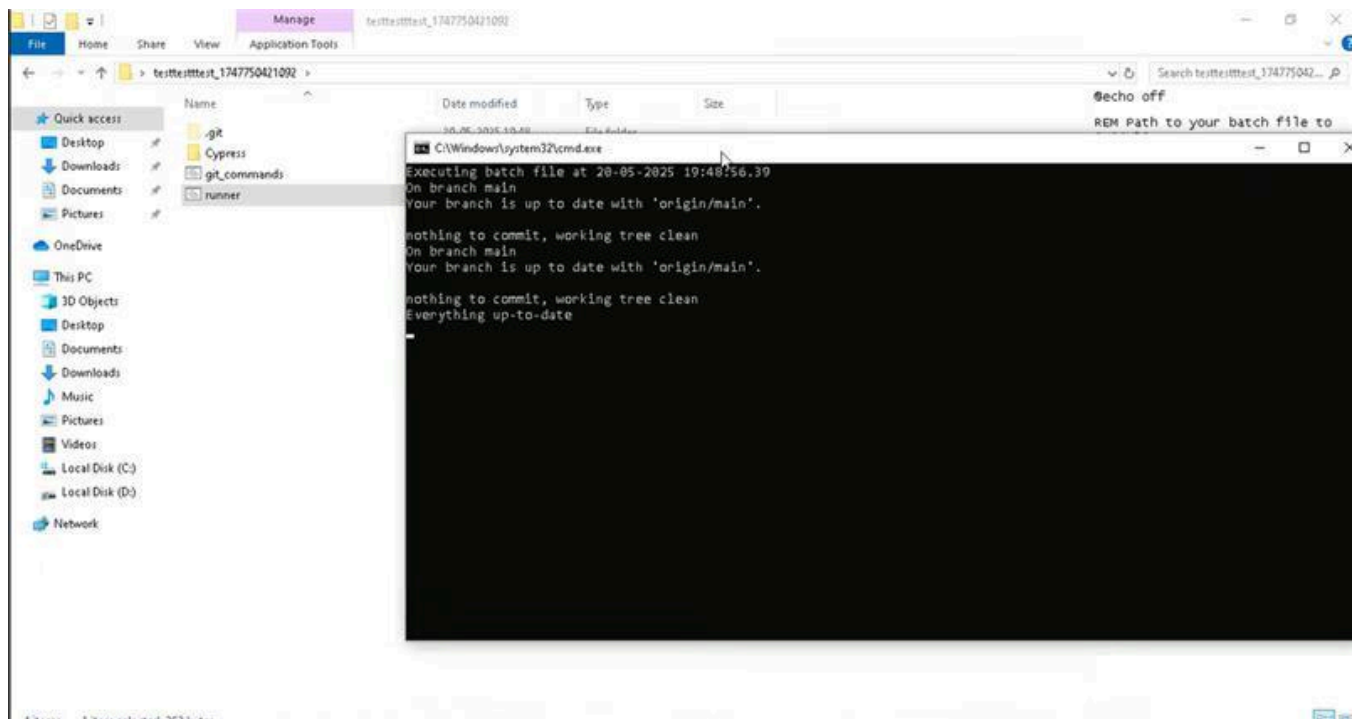
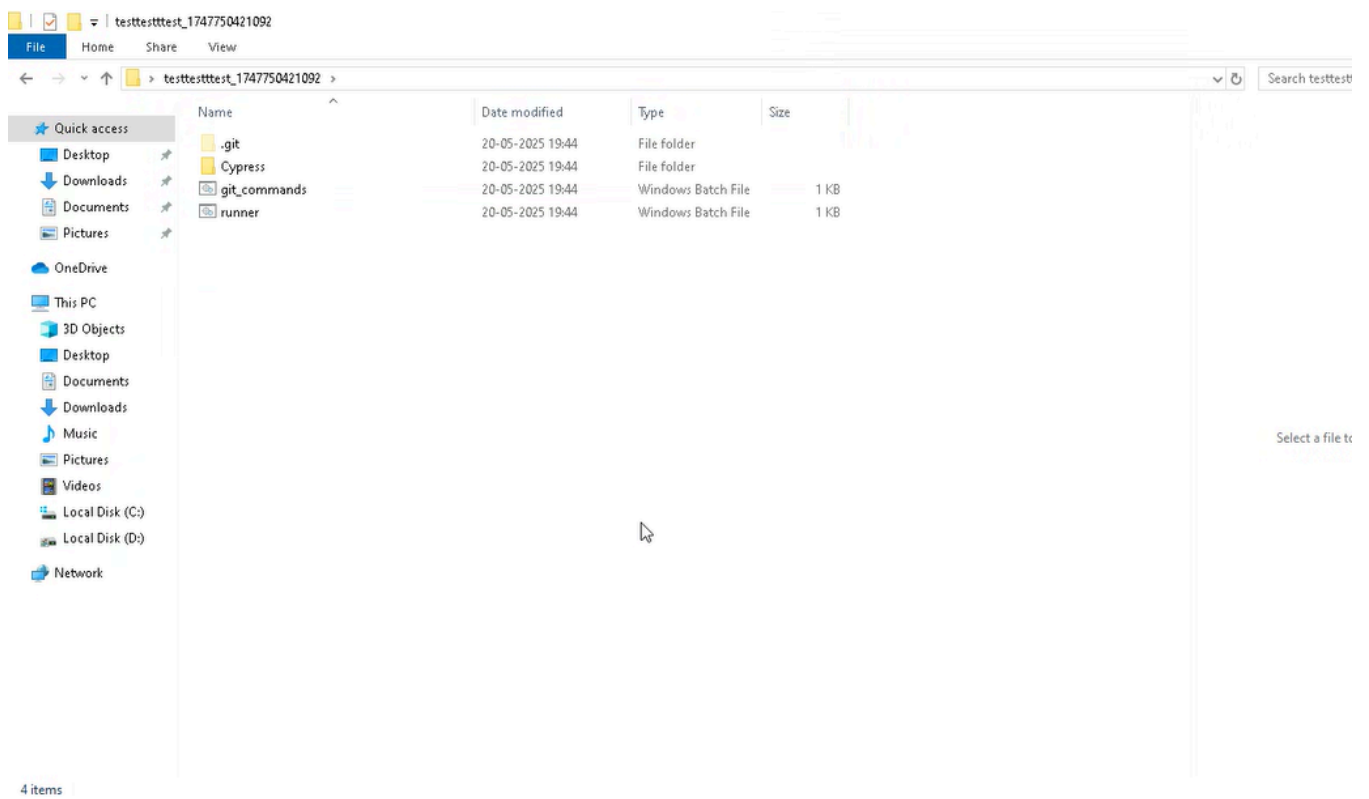
1. Please open the folder created on the desktop with the email name you used to login.





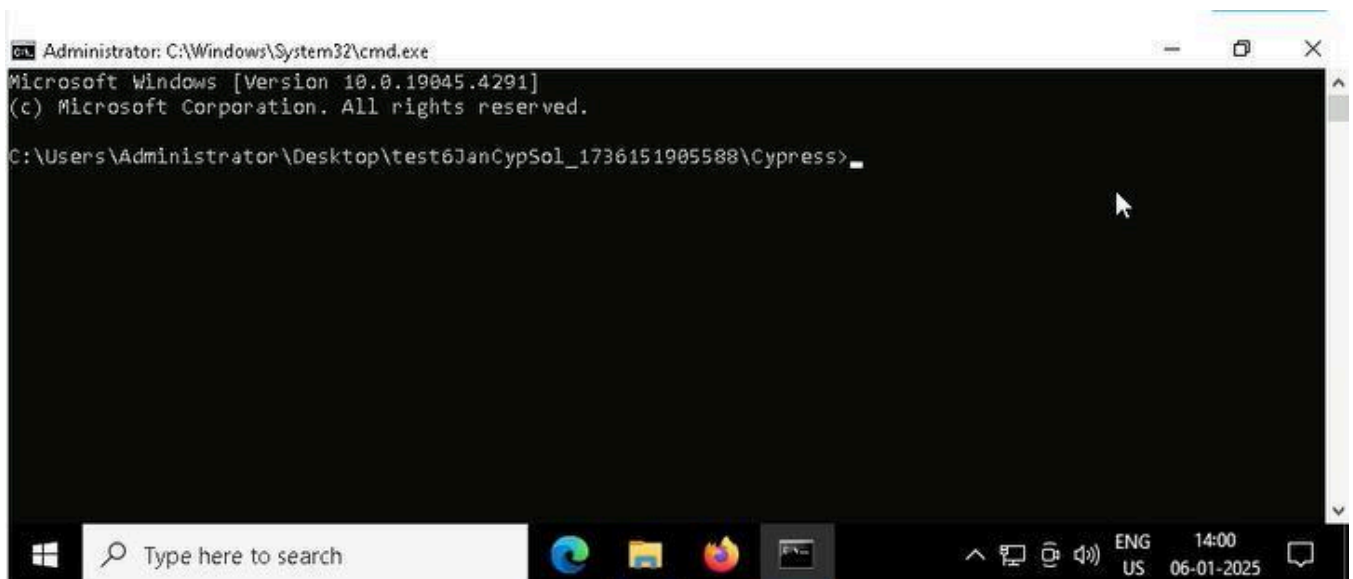
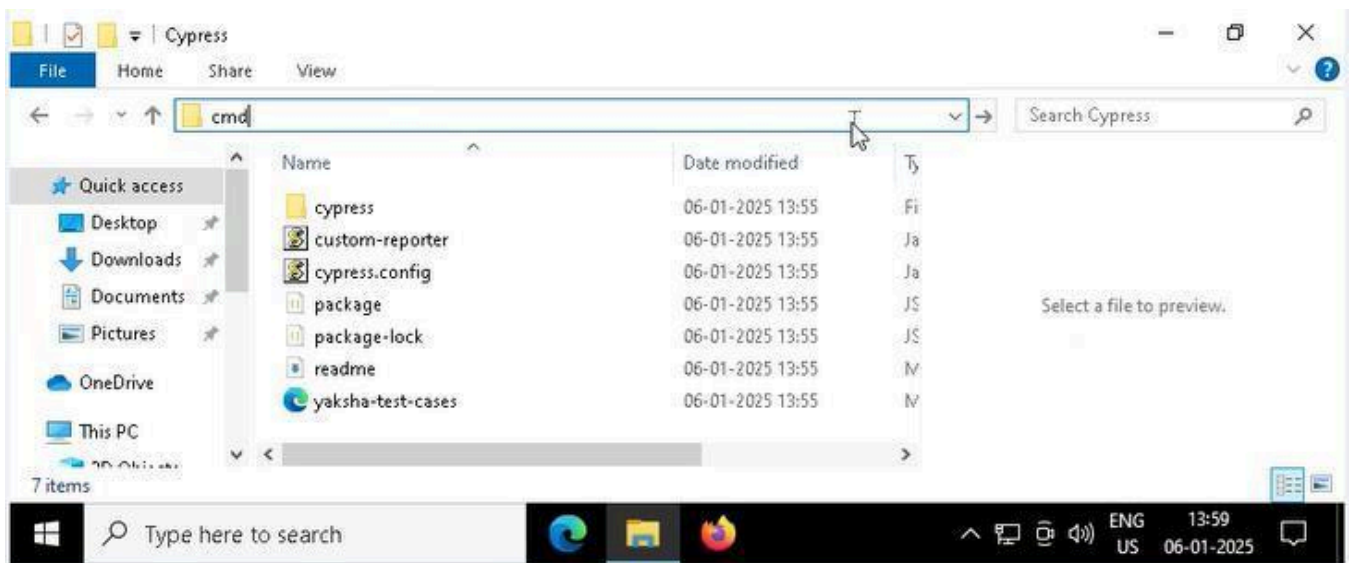
2. Execute the “runner” file to keep pushing the code at regular intervals.

## Mymedic automation using cypress



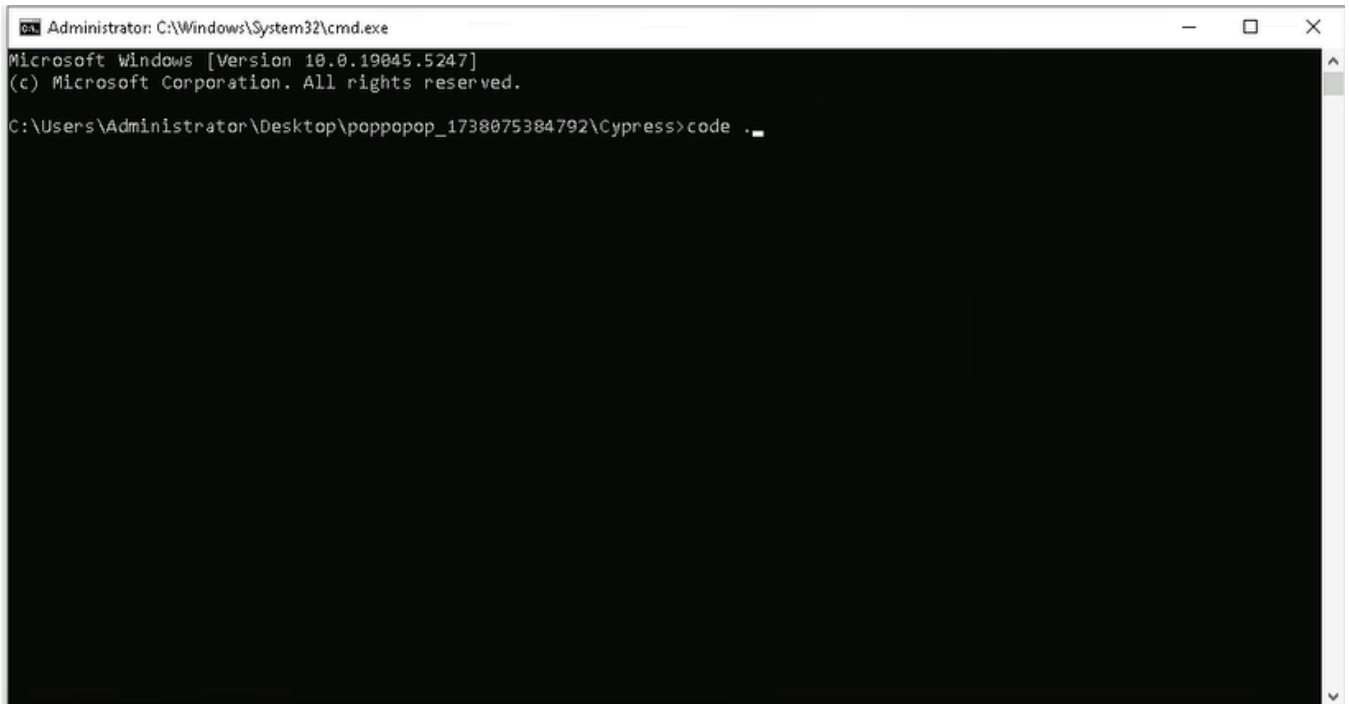
Mymedic automation using cypress

### 3. Open command prompt with its location using “cmd” in search bar:



4. Open VS Code through cmd using below command:

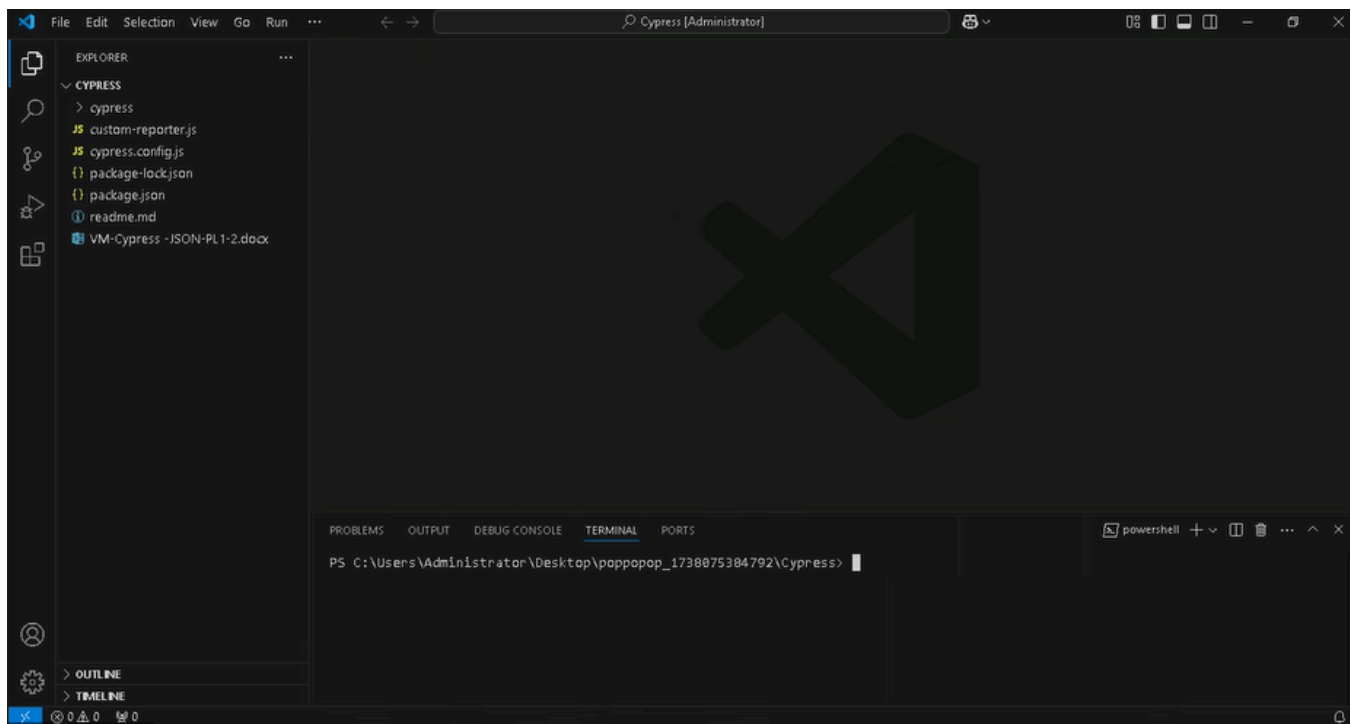
`code .`



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5247]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop\poppopop_1738075384792\Cypress>code .
```

1. Once VsCode is open. Please open the terminal and execute below given next commands:



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

**npm install**

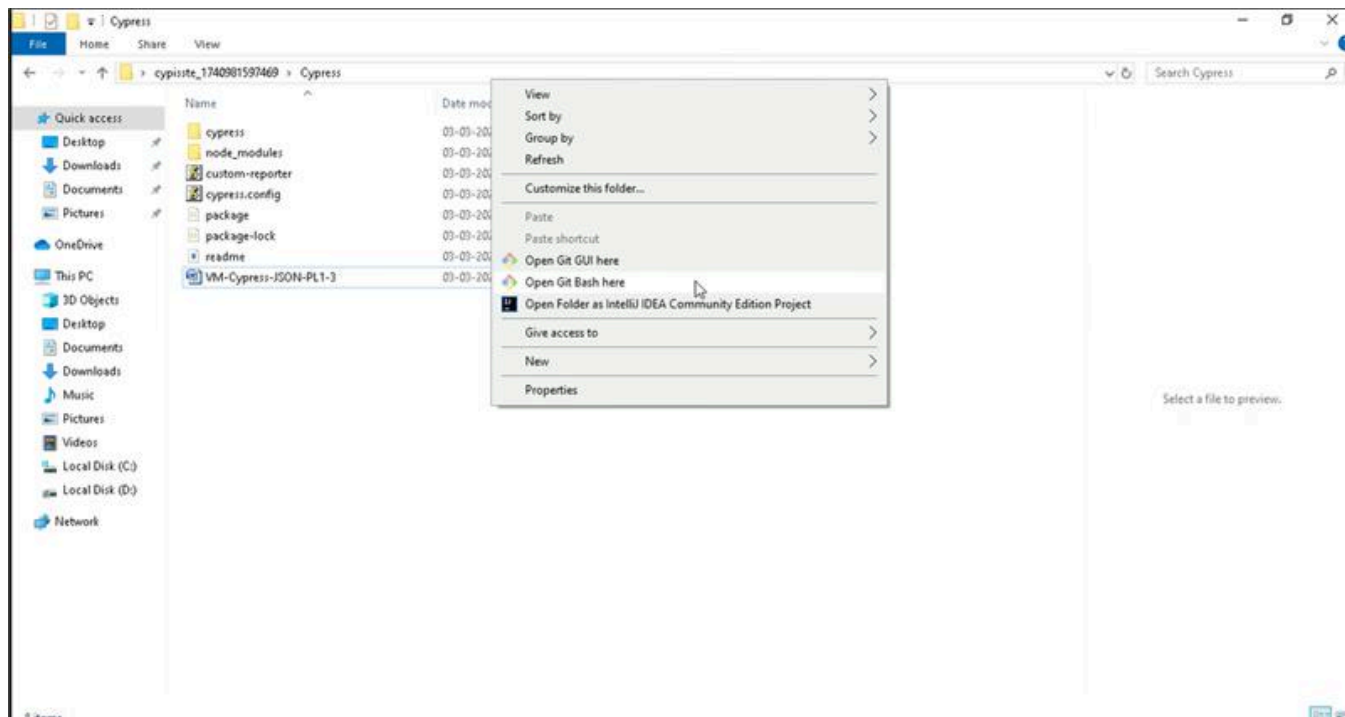
2. Run the following command to open the interactive Cypress Test Runner in the Cypress folder path:

**npx cypress open**

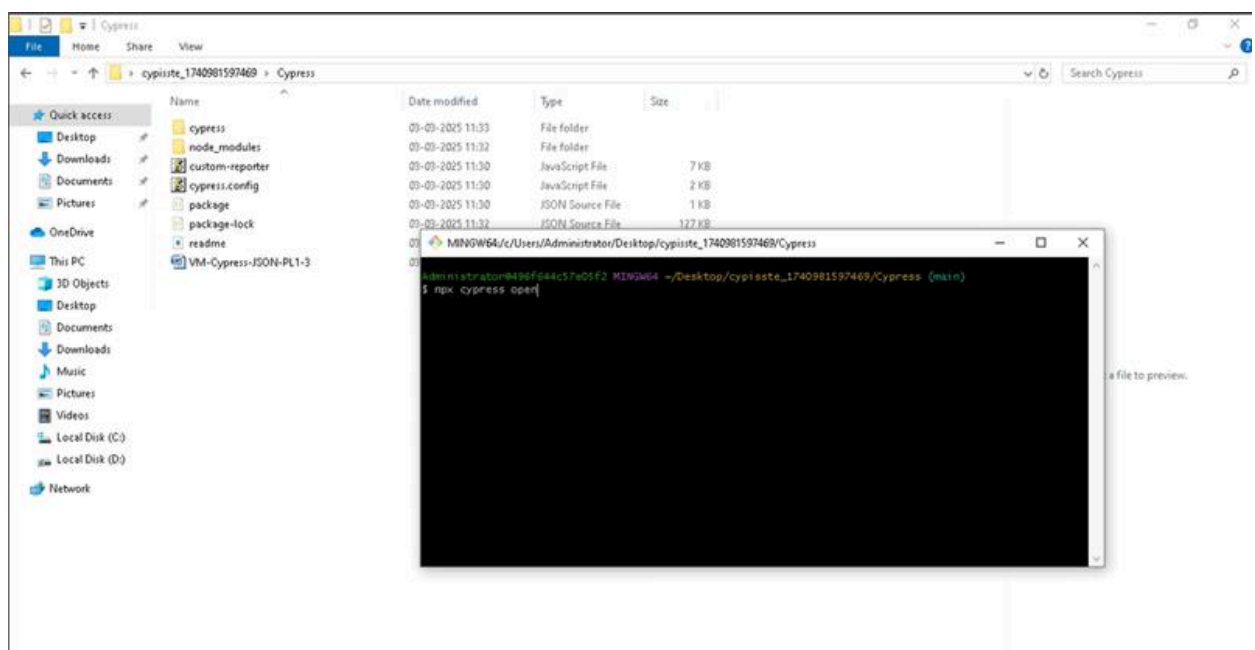
3. Run the following command to run test cases in headless mode in the Cypress folder path:

**npx cypress run**

If you face any issue in executing above commands i.e npx cypress open and npx cypress run in cmd. Please try to run these in git bash from cypress folder as:



and then execute commands like:



Mymedic automation using cypress