

YAKSHA HEALTH APP WITH JAVASCRIPT AND CYPRESS

Mymedic automation using 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 fixtures folder which you have to use.

Path	File	Description
\fixtures	<ul style="list-style-type: none">• DateData.json• Dispensary.json• Nursing.json• PatientName.json• ValidLogin.json	1. Contains data to read from json file.
PageObjects\Pages	<ul style="list-style-type: none">• AccountingPage• AppointmentPage• CustomPage	1. All core activities to be performed here. 2. The comments associated

Mymedic automation using cypress

	<ul style="list-style-type: none"> ● DispensaryPage ● HelpDeskPage ● LaboratoryPage ● LoginPage ● MaternityPage ● NursingPage ● OperationTheaterPage ● ProcurementPage ● SettingsPage 	<p>with each templated method here describe the expectation.</p> <p>3. Declare any 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 login.json file and fetch the user's name and password.</p> <p>2. It should call the method performLogin()</p> <p>3. Perform login method will perform authentication with the username and password.</p> <p>4. Verify admin name is visible on the home page.</p> <p>It is must to implement this login functionality at first and then implement any other test case.</p>	<p>Reference path</p> <p>\PageObjects\Pages\LoginPage</p> <p>methods</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 https://healthapp.yaksha.com/Home/Index#/Maternity/PatientList</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</p>	<p>Reference path</p> <p>\PageObjects\Pages\MaternityPage</p> <p>methods</p> <p>verifyReceiptModal()</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.

Mymedic automation using cypress

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
		Details for the first record. 5. Verify that the "Maternity Allowance Payment Receipt" modal appears.		
2	Verify Error Message When Editing an Occupied Bed	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard 2. Click on the Manage Bed tab. 3. Locate any occupied bed and click the Edit button. 4. Verify that an error message "Cannot modify occupied beds." is displayed.	Reference path \PageObjects\Pages\SettingsPage methods verifyErrorMessage()	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	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation 2. Note down the Male Ward Bed Count from the table. 3. Navigate to https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard 4. Click on the Manage Bed tab. 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 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. 7. Navigate back to https://healthapp.yaksha.com/Home/Index#/Helpdesk/BedInformation 8. Verify that the Male Ward Bed Count is updated by 1.	Reference path \PageObjects\Pages\HelpDeskPage methods verifyTotalMaleWard()	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	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Dispensary/ActivateCounter 2. Activate any counter.	Reference path \PageObjects\Pages\DispensaryPage	Add New Patient modal should open.

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
	Alt + Enter keyboard shortcut	3. Select the Sale tab and press Alt + N.	methods activateCounter()	
5	Verify Remarks text field placeholder value	1. Navigate to https://healthapp.yaksha.com/Home/Index#/OperationTheatre/OtBookingList 2. Click on new OT booking button. 3. Verify Remarks Text area is enable 4. Verify the placeholder name of Remarks text area.	Reference path \PageObjects\Pages\OperationTheaterPage methods verifyRemarks()	Verify the placeholder name of Remarks text area.
6	Capture screenshot of Inventory Requisition section	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Settings/ADTManage/ManageWard 2. Click on the Manage Bed tab. 3. Click on the Add Bed button. 4. Enter the required details to create a new Male Ward bed. 5. Unselect the "Is Active" checkbox. 6. Click Save to create the bed. 7. Use the Search Box to search for the newly created bed. 8. Verify that the bed appears in the list with inactive status.	Reference path \PageObjects\Pages\SettingsPage methods verifyBedCreation()	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	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Accounting/Reports 2. Select Daily Transactions Report from the available reports. 3. Choose 2023 as the Fiscal Year. 4. Click on the Load button.	Reference path \PageObjects\Pages\AccountingPage methods verifyDailyTransaction()	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	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Appointment/ListAppointment 2. Select counter i.e NEW-1 and	Reference path \PageObjects\Pages\AppointmentPage	Upon searching for the same patient again using 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
	Check-in and Invoice Print	<p>then navigate to link.</p> <p>2. Select "New Patient" from the dropdown.</p> <p>3. Select "All Doctors" option from the Doctor field.</p> <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 "Check-In this 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 1st January 2020 and search for the same patient and check its status.</p>	<p>methods</p> <p>verifyAppointmentStatusChangeAfterInvoicePrint()</p>	
9	Verify Adding Triage for an Outpatient	<p>1. Navigate to https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient</p> <p>2. Locate the same patient used in the previous test cases.</p>	<p>Reference path</p> <p>\PageObjects\Pages\NursingPage</p>	<p>A success confirmation popup with the message: "Chief Complaint Added Successfully" should appear.</p>

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
		3. Select the checkbox for that patient. 4. Click on the Triage button. 5. Enter "Temp value" in Chief Complaint field. 6. Click on Add Triage.	methods verifyAddingTriageForOutpatient()	
10	Verify Nursing Check-in for an Outpatient	1. Perform all points of TC#8 and then navigate to https://healthapp.yaksha.com/Home/Index#/Nursing/OutPatient 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.	Reference path \PageObjects\Pages\NursingPage methods verifyNursingCheckinForOutpatient()	A success confirmation popup with the message: "Nursing CheckIn Added Successfully" 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 dispensary module. 2. Click on Reports. 3. Click on User Collection report to navigate to the respective report section. 4. Read the fromDate from dtaeRange.json file and enter in From Date input field. 5. Select morning counter from dropdown. 6. Click on Show Report button to generate the report. 7. Verify the generated report correctly displays the morning counter in counter column.	Reference path \PageObjects\Pages\DispensaryPage methods generateMorningCounterReport()	Verify the generated report correctly displays the morning counter in 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 of the star icon and retrieves the tooltip text.	Reference path \PageObjects\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
			methods verifyTooltipTextIsVisible()	
13	Add and Verify New Imaging Type in Radiology	1. Navigates to the Settings module and clicks on the Radiology submodule. 2. Clicks on the "Add Imaging Type" button to open the modal for adding a new imaging type. 3. Fills the "Imaging Item Name" field with a random name (Test-{random4digitnumber}) and clicks "Add". 4. Verifies that the newly added imaging type appears in the list of imaging types.	Reference path \PageObjects\Pages\SettingsPage methods addAndVerifyImg()	Verifies that the newly added imaging type appears in the list of imaging types.
14	Web Element Handling for Dropdowns in Purchase Request	1. Navigate to Purchase Request List. 2. Fetch the fromDate and toDate from dateRange.json file. 3. Apply date range filter. 4. Retrieve dates from "Requested Date" column and validate each date within range. 5. Wait for success or failure.	Reference path \PageObjects\Pages\ProcurementPage methods verifyRequestedDateColumnDateWithinRange()	It should be able to retrieve dates are within range.
15	Verify logout functionality from Admin dropdown	1. Navigate to index page i.e https://healthapp.yaksha.com/Home/In dex#/ 2. Click on the "Admin" dropdown 3. Click on "Log Out" option. 4. Verify the user is redirected to the login page.	Reference path \PageObjects\Pages\LoginPage methods verifyLogoutFunctionality()	User is logged out successfully and the login page is displayed.

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

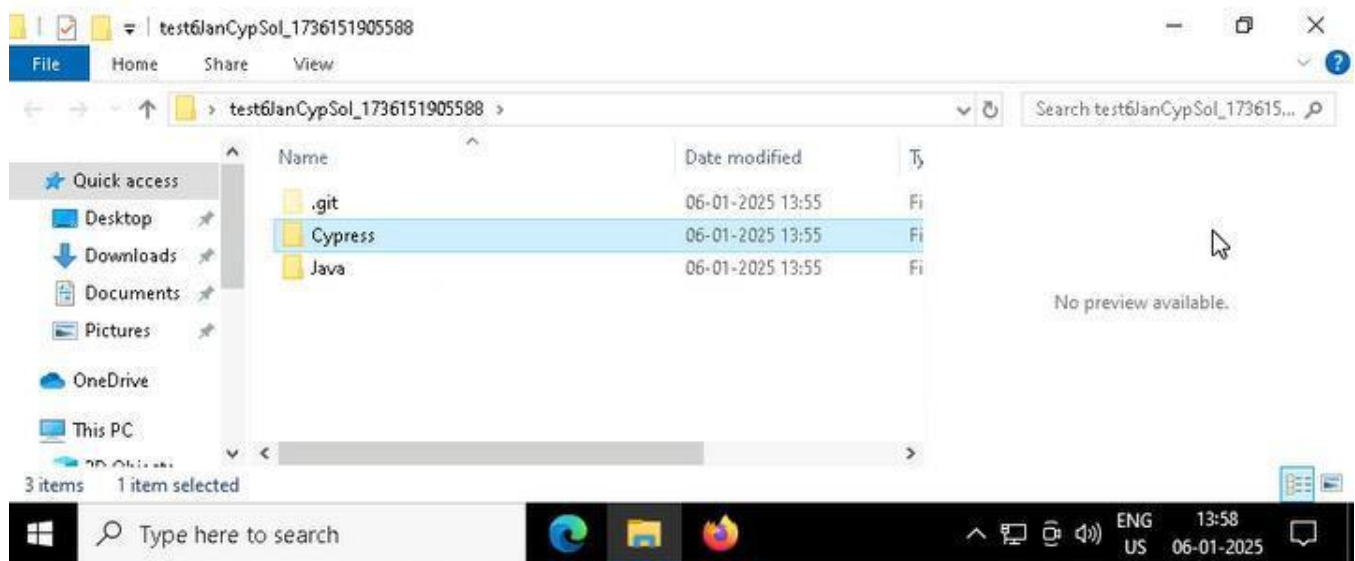
- **Browser Automation:** Interacting with web elements and testing multiple browsers.
- **Assertions & Validations:** Ensuring app behaviour meets expected results.
- **End-to-End Testing:** Automating real user interactions and validating overall app functionality.

1.1 CODE QUALITY/OPTIMIZATIONS

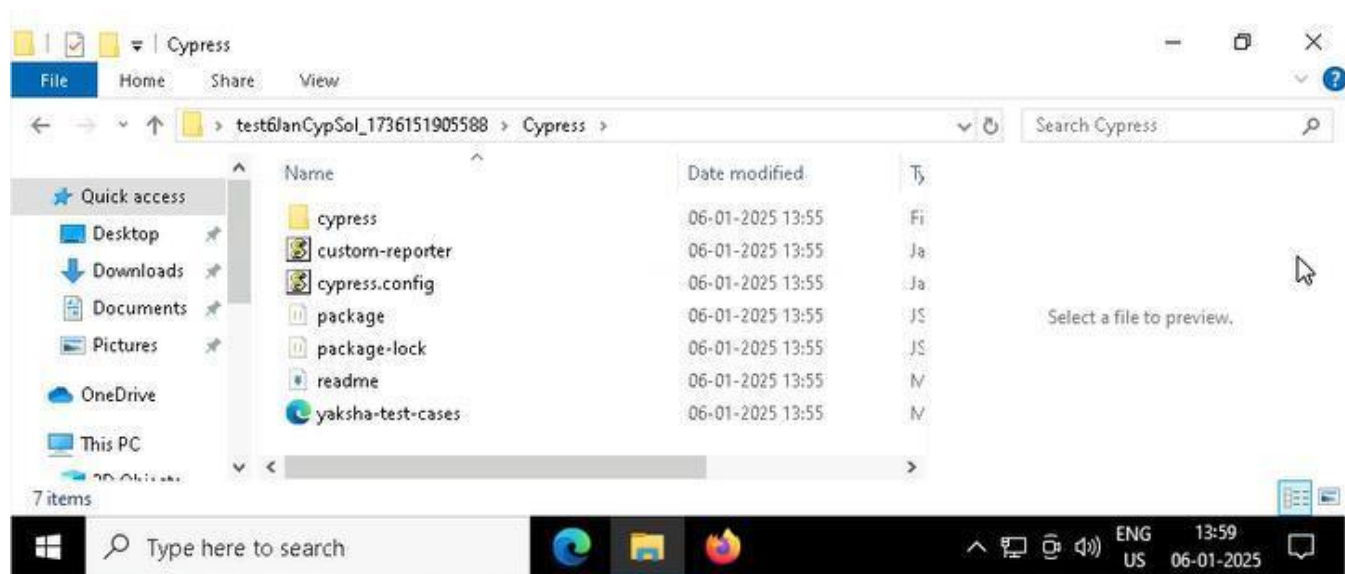
- ### Steps for Execution:

-

Mymedic automation using cypress

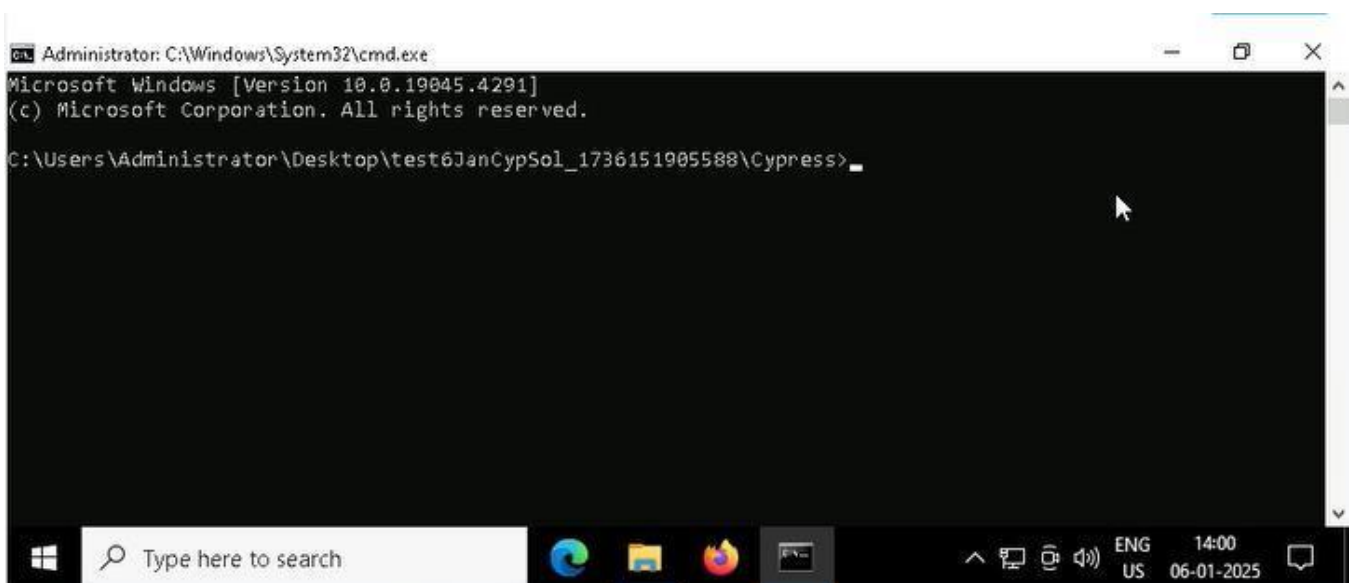
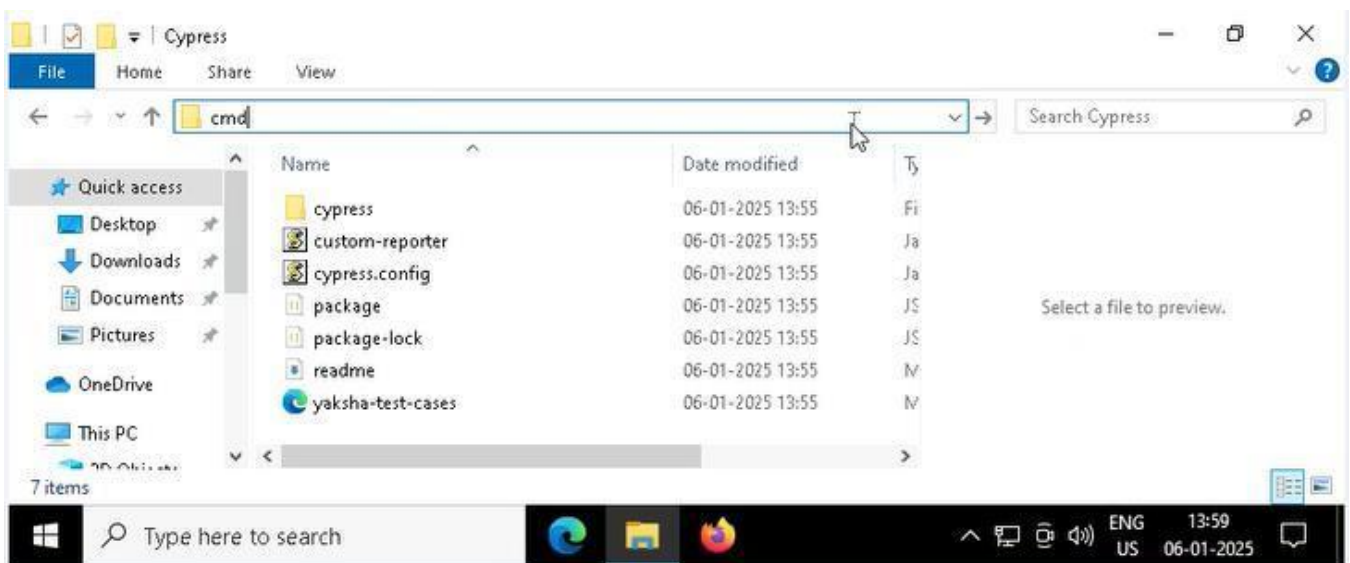


2. Go into the Cypress folder



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

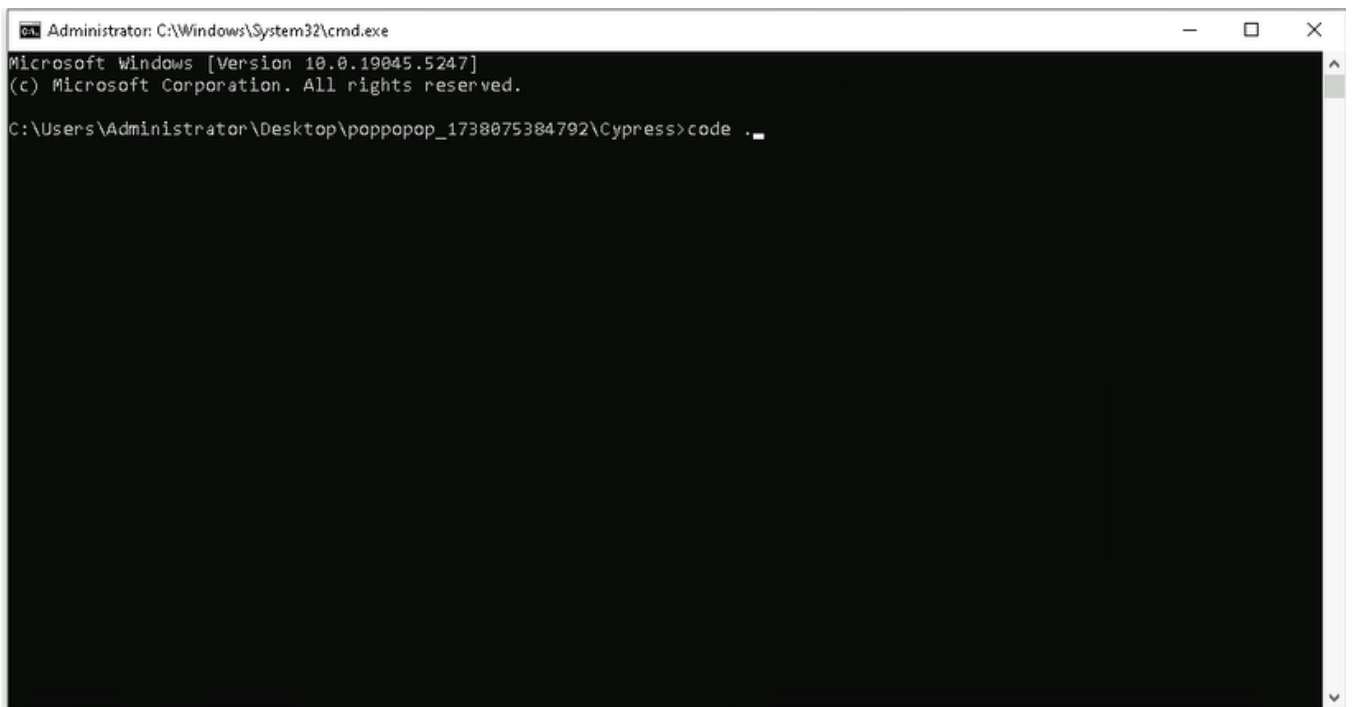
Mymedic automation using cypress



5. Open VS Code through cmd using below command:

`code .`

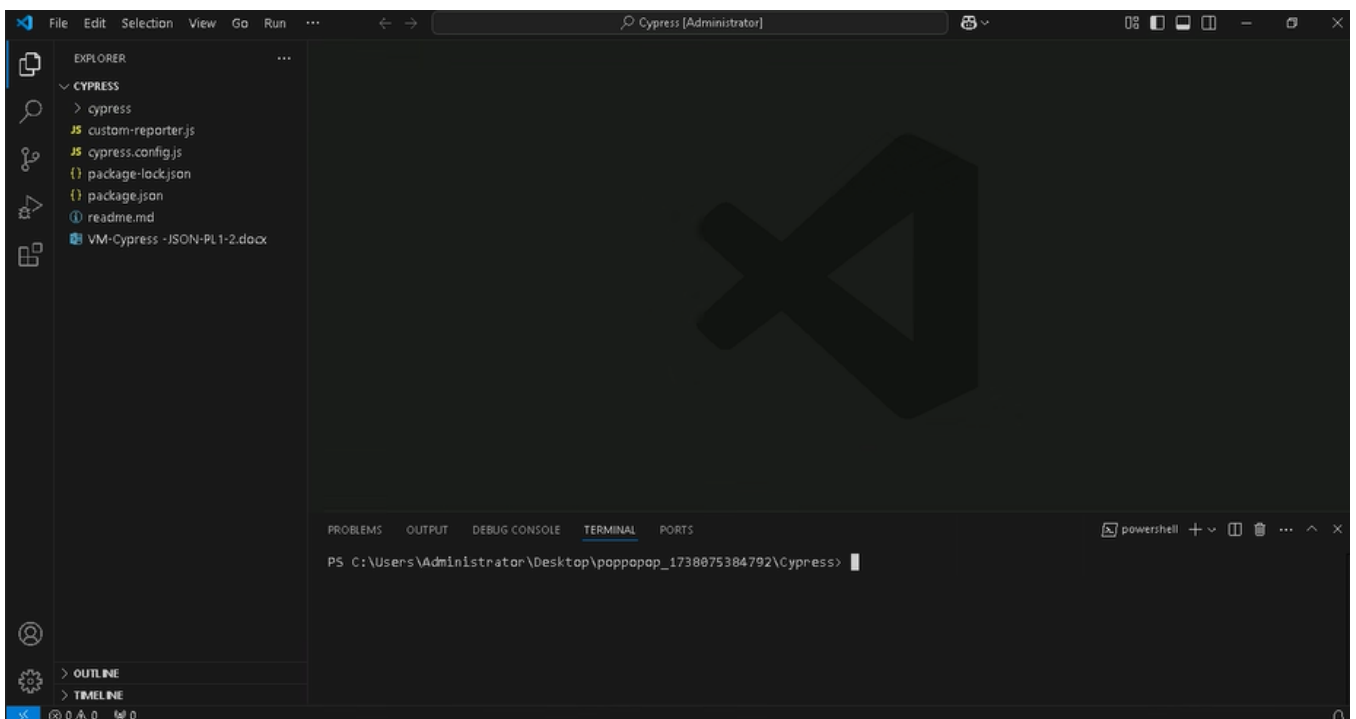
Mymedic automation using cypress



```
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 .
```

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



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

Mymedic automation using cypress

npm install

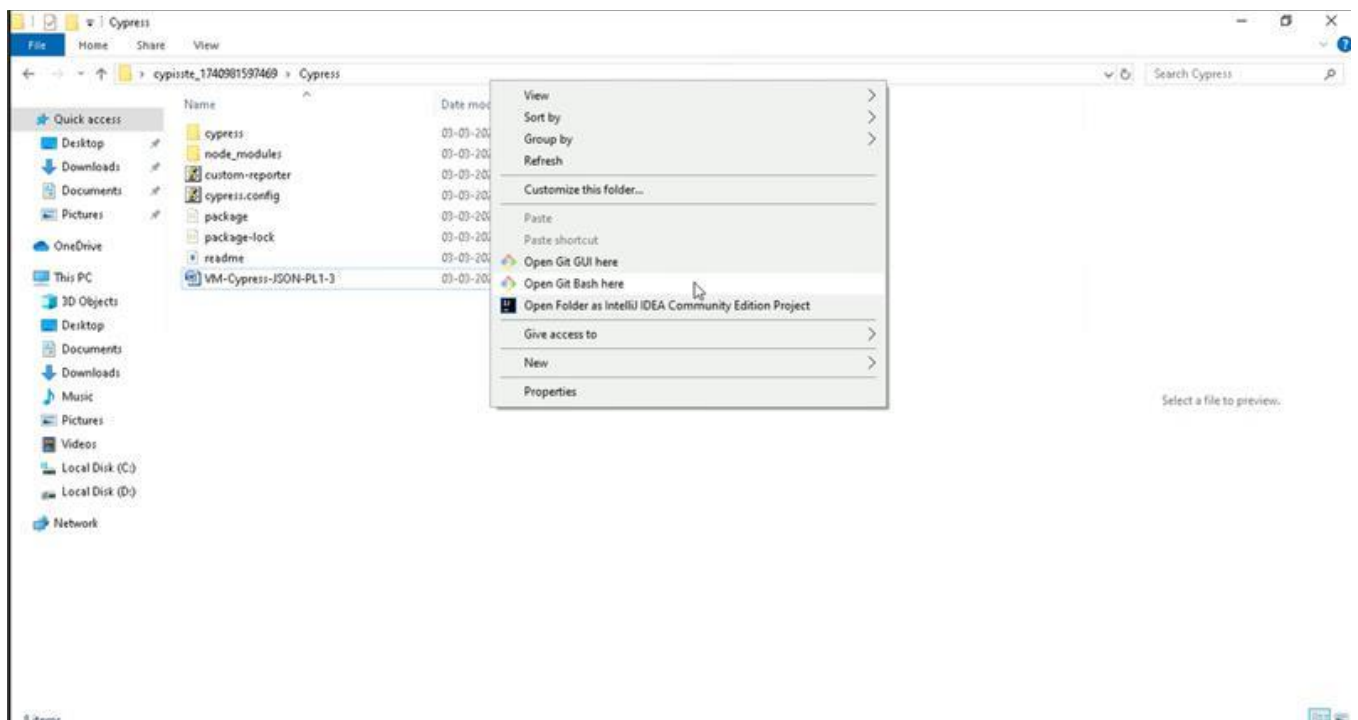
2. Run the following command to open the interactive Cypress Test Runner in the Cypress folder path:

npx cypress open

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

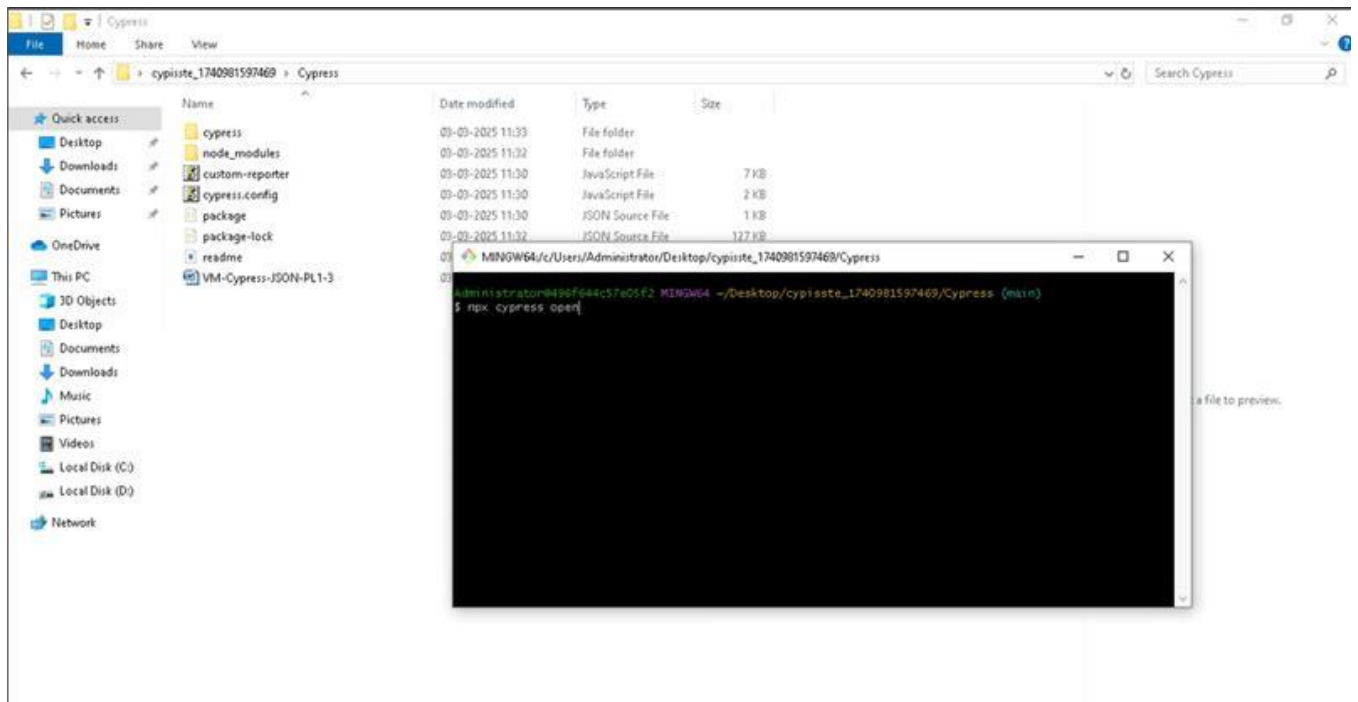
npx cypress run

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:



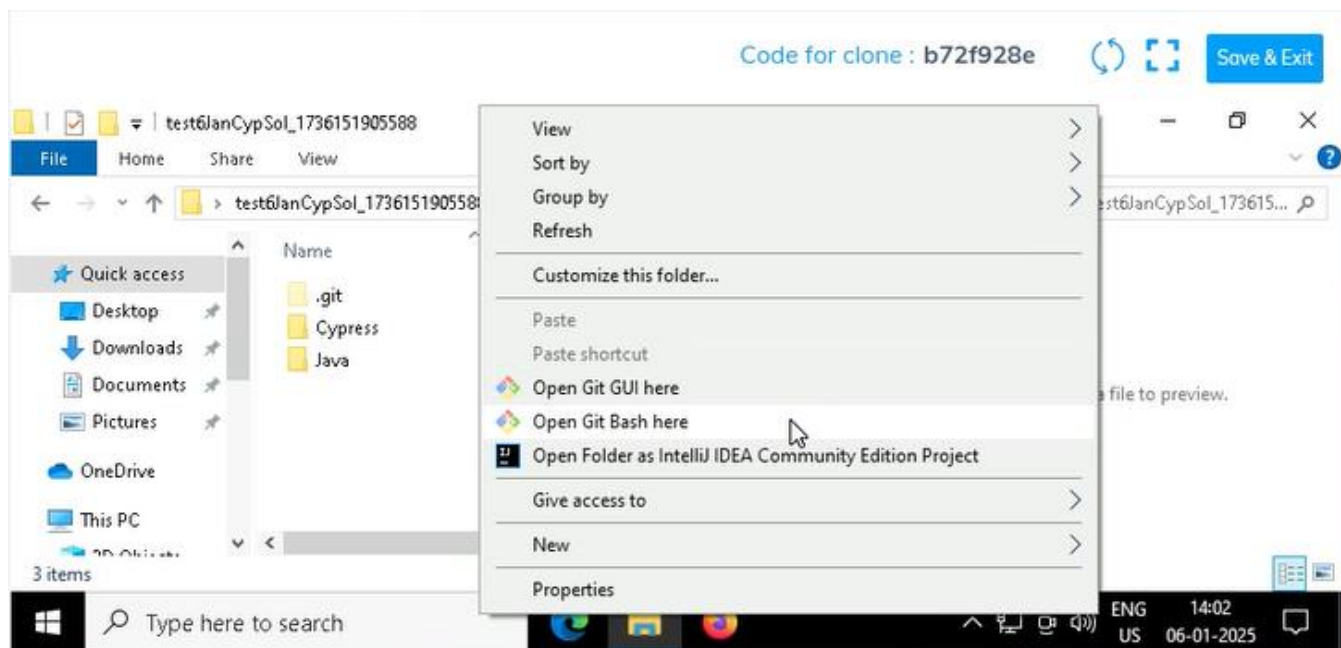
Mymedic automation using cypress

and then execute commands like:



4. Once you have executed the test cases. Now it is necessary to push your code to git. For this, please go inside the folder created on desktop with the email id you have used to login and then:

1. Open gitbash



Mymedic automation using cypress

2. Add all files

The screenshot shows a web-based IDE interface. At the top, there's a status bar with 'Speed Test Avg: 4.36Mbps', 'Live: >= 10Mbps', 'Time Remaining: 02:54:11', and a 'Final Submission' button. The main area is divided into three panels. The left panel contains a 'Question' section titled 'Q1: VM-Playwright-Excel-Solution' and an 'Instructions' section with an 'Introduction' and a 'Launch' button. The right panel shows a file explorer with a search bar and a 'Select a file to preview.' message. The bottom panel is a terminal window with the following content:

```
MINGW64/c/Users/Administrator/Desktop/zxczxczxc_1732782665185
Administrator@4f6d8dad9dbf5e2 MINGW64 ~/Desktop/zxczxczxc_1732782665185 (main)
$ git add .
```

3. Commit the changes

The screenshot shows a terminal window with the following content:

```
MINGW64/c/Users/Administrator/Desktop/zxczxczxc_1732782665185
warning: in the working copy of 'Playwright/node_modules/zone.js/plugins/zone-patch-rxjs/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/plugins/zone-patch-socket-io.min/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/plugins/zone-patch-socket-io/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/plugins/zone-patch-user-media.min/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/plugins/zone-patch-user-media/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/testing/package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/zone.d.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Playwright/node_modules/zone.js/zone_externs.js', LF will be replaced by CRLF the next time Git touches it
Administrator@4f6d8dad9dbf5e2 MINGW64 ~/Desktop/zxczxczxc_1732782665185 (main)
$ git commit -m "committing the updates"
```

4. Push the changes

Mymedic automation using cypress

Code for clone : be5b9203



Save & Exit

MINGW64/c:/Users/Administrator/Desktop/zxczxczc_1732782665185

```
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-message-port/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-promise-test.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-promise-test/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-resize-observer.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-resize-observer/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-rxjs-fake-async.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-rxjs-fake-async/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-rxjs.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-rxjs/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-socket-io.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-socket-io/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-user-media.min/package.json
create mode 100644 Playwright/node_modules/zone.js/plugins/zone-patch-user-media/package.json
create mode 100644 Playwright/node_modules/zone.js/testing/package.json
create mode 100644 Playwright/node_modules/zone.js/zone.d.ts
create mode 100644 Playwright/node_modules/zone.js/zone_externs.js
```

Administrator@4f6d8dad9dbf5e2 MINGW64 ~/Desktop/zxczxczc_1732782665185 (main)

\$ git push



Type here to search



ENG
US

14:12
28-11-2024



Mymedic automation using cypress