

YAKSHA HEALTH APP WITH TYPESCRIPT AND PLAYWRIGHT PL2-4

Usecase summary

Project Name:opensource-demo.orangehrmlive.com app – HR management system.

Use Case Summary: opensource-demo.orangehrmlive.com is designed to manage Employee record. Records (HRMS). It allows users to view, search, and manage employee records. It features functionality such as adding/editing employee details, filtering data by department and role, and exporting records. The primary use case is to automate the process of employee data management, ensuring efficient and reliable operations for human resources teams in organizations

Technology Stack:

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

Key Features:

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

Expected Outcomes:

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

Overview of the application

Pages/Features that are to be focused for the application

Please use the Application URL <https://yakshahrm.makemylabs.in/orangehrm-5.7>

PROBLEM STATEMENT

Need to automate the following activities using playwright+typescript

You will be given few Json files in Data folder like Login.json and TestData.Json

Path	File	Description
src\data	login.json TestData.json	1. Contains data to read from json file.

HRMS automation using playwright

src\ pages	<ul style="list-style-type: none"> • AdminPage • BuzzPage • LoginPage • MyInfoPage • PIMPage 	<ol style="list-style-type: none"> 1. All core activities are to be performed here. 2. The comments associated with each templated method here describe the expectation. 3. Declare any variable/object you need to share data/status between different methods. 4. Do not modify the signature of methods declared here.
------------	---	---

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
1	Verify admin Search Functionality	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "Admin" button in sidebar 3. Enter the user role as admin from dropdown. 4. Click on "Search" button <p>It must implement this login functionality at first and then implement any other test case.</p>	<p>Reference path</p> <p>\src\pages\AdminPage</p> <p>methods</p> <p>adminSearchh();</p> <p>You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.</p>	List having role "admin" should be displayed
2	Verify Image could be dragged and dropped in buzz tab	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "Buzz" tab in sidebar 3. Click on share photo button 4. Click on image area to upload image 5. drag & drop image> Click on Share button 	<p>Reference path</p> <p>\src\ pages\BuzzPage</p> <p>methods</p> <p>SharePhotoPost()</p> <p>You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.</p>	Image should get uploaded
3	Verify Like button is Functional	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "Buzz" tab in sidebar 3. Click on Heart button on any post in the List 	<p>Reference path</p> <p>\src\ pages\BuzzPage</p> <p>Methods:-</p> <p>likePost()</p> <p>You can use highlightElement method present in</p>	Heart count should increase by 1 on clicking the like button

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
			CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	
4	Verify the Comment could be added to a post	1. Login using valid credentials 2. Click on search bar in sidebar 3. Enter the keyword in the sidebar 4. Verify all the entries fetched contains the keyword	Reference path \src\ pages\BuzzPage Methods:- addCommentToPost() You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	Comment should get added to the post successfully
5	Verify "Edit Post" Functionality	1. Login using valid credentials 2. Click on "Buzz" tab in sidebar 3. Click on Edit button on any post of user in the List 4. Enter a message and Click on Post button	Reference path \src\ pages\AdminPage methods AdminEdit() You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	post message should get edited
6	Verify "Delete Post" Functionality	1. Login using valid credentials 2. Click on "Buzz" tab in sidebar 3. Click on Delete button on any post in the List 4. Click on "Yes delete" button in popup	Reference path \src\pages\BuzzPage methods : delete()	Post should get Deleted
7	Verify "Get" help button is functional	1. Login using valid credentials 2. Click on "Question mark icon" button	Reference path \PageObjects\Pages\LoginPage methods: getUrl()	User should be redirected to new URL

			highlight the element before performing any action on it. It takes locator as a parameter.	
12	Verify the Subtabs css change after click on the subtab	1. Navigate to URL with Valid Credentials 2. Navigate to "admin" tab 3. Click on the subtab Note: - If you cannot find the locator for "Proceed" button, please try to keep the page at 75%-80%.	Reference path \src\pages\AdminPage Methods: hoverAdmin() You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	verify the subtab appearance changes when clicked upon subtab.
13	Verify two employees with same employee id can't be created	1. Navigate to URL with Valid Credentials 2. Navigate to "PIM" tab 3. Fetch the id from list in "Employee List" subtab 4. Click on add employee and add same employee id fetched from previous step Note: - If you cannot find the locator for "Proceed" button, please try to keep the page at 75%-80%.	Reference path \src\pages\PIMPage Methods: EmpIdError() You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	"Employee Already Exist" message should be displayed
14	Verify the "Corporate branding" "reset to default" functionality	1. Navigate to URL with Valid Credentials 2. Navigate to "Admin" tab 3. Fetch the Primary color from default setting from "Corporate Branding" 4. Reset the color to desired color & Publish 5. "Reset to default" button should reset the color back to default value Note: - If you cannot find the locator for "Proceed" button, please try to keep the page at 75%-80%.	Reference path \src\pages\AdminPage Methods: resetColor() You can use highlight Element method present in Common Methods file to highlight the element before performing any action on it. It takes locator as a parameter.	Primary color should set back to #FF7B1D

15	Verify Attachment could get Uploaded to the "My info" tab	1. Navigate to URL with Valid Credentials 2. Navigate to "My Info" tab 3. Click on "My Info" tab from sidebar 4. Click on "Contact Details" and add a attachment 5. Select desired from the device 6. Click the Save Button Note: - If you cannot find the locator for "Proceed" button, please try to keep the page at 75%-80%.	Reference path src\pages\MyInfoPage Methods: - UploadAttachmentInDependent() You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as a parameter.	Success Message should be displayed and uploaded file should be displayed in the list
----	---	---	--	---

Learners will gain experience in building strongly typed applications using React.js and managing data flow with **TypeScript**. They'll learn how to define interfaces, use types for error prevention, and improve code maintainability. URL <https://opensource-demo.orangehrmlive.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.

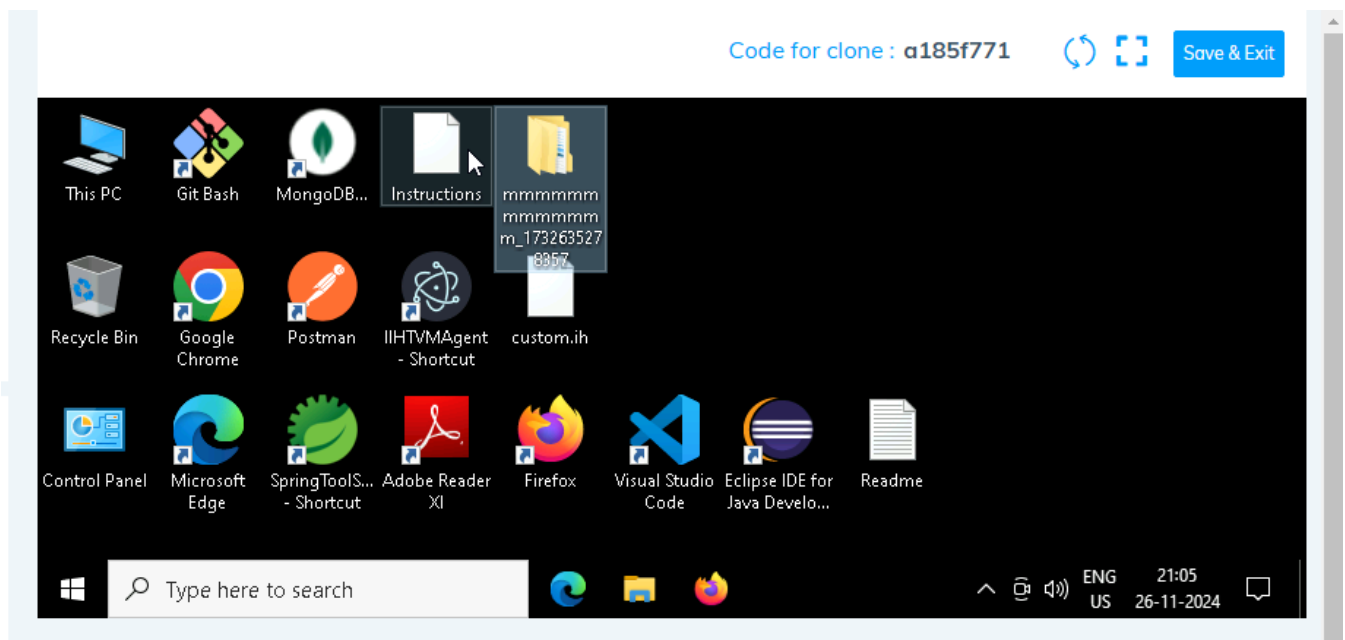
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

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:

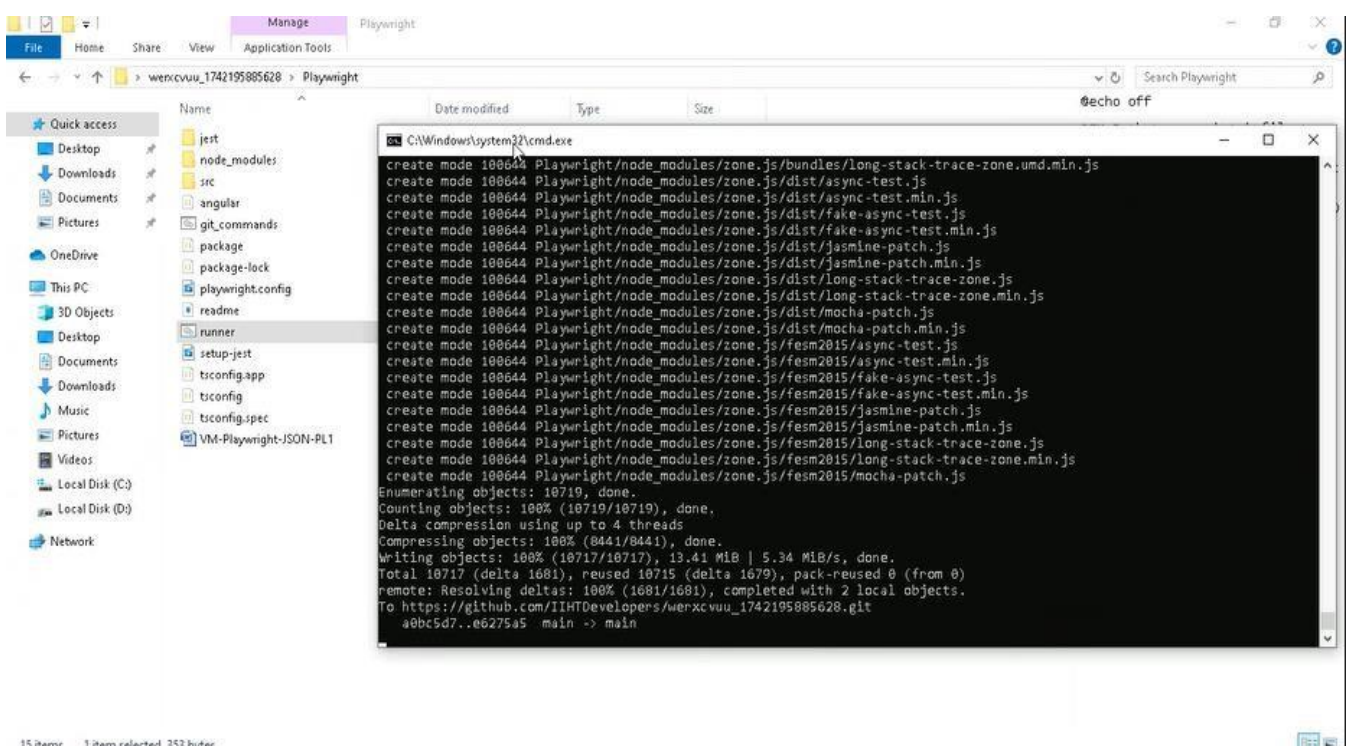
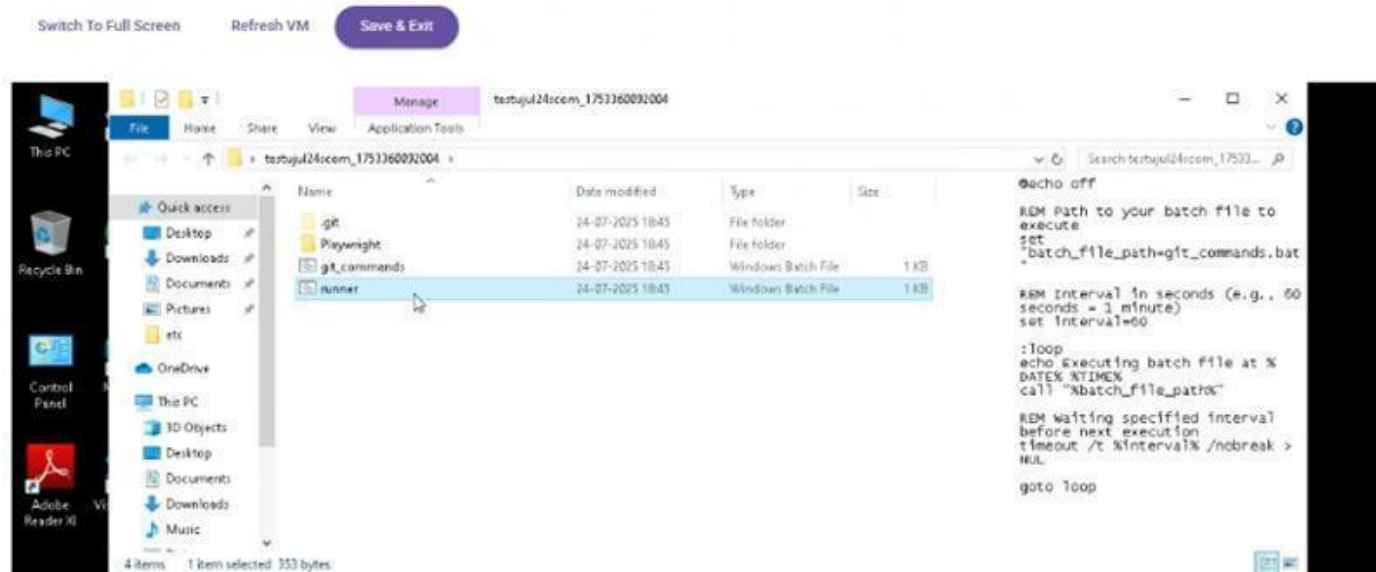
Steps for Execution:



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

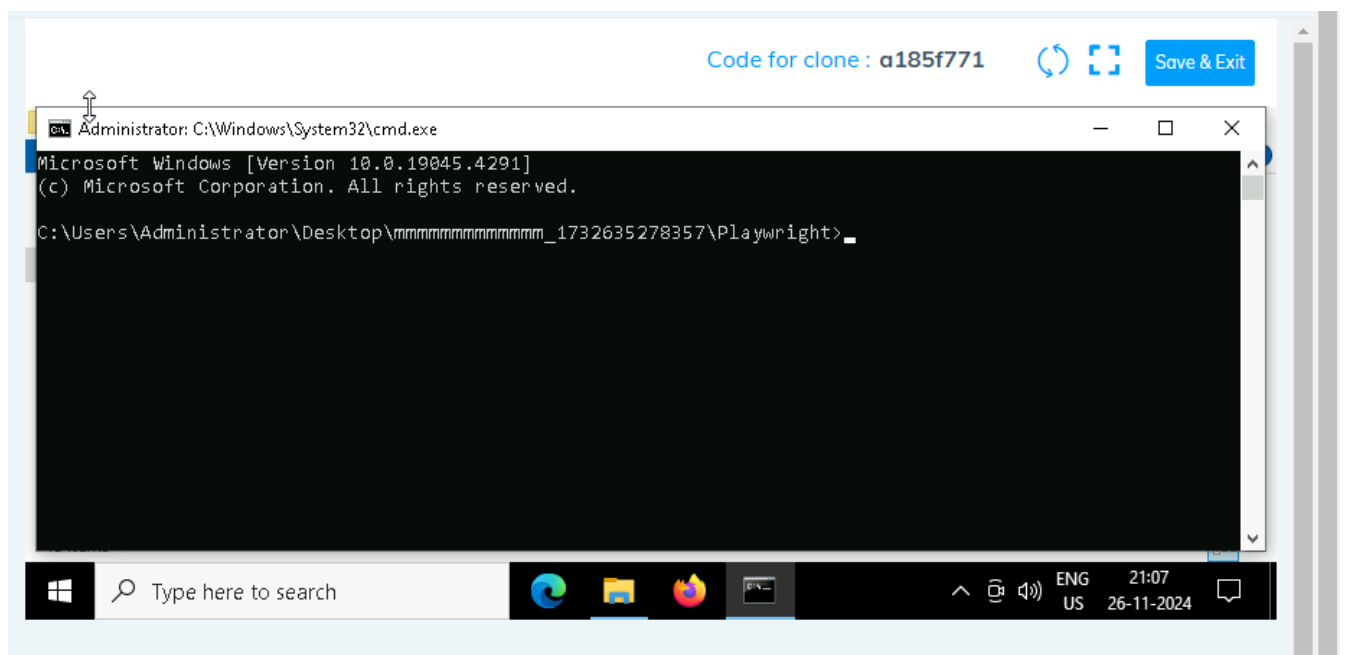
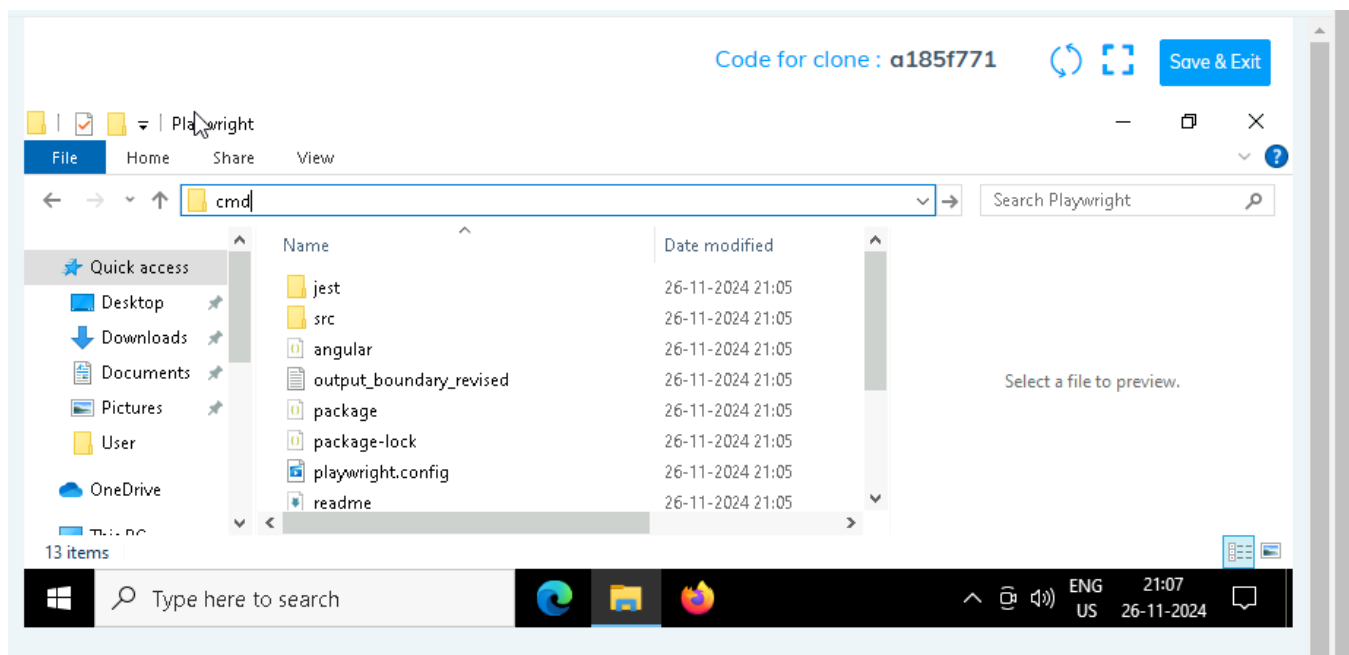
HRMS automation using playwright

2. Go into the Playwright folder and execute this “runner” file. This will keep pushing the code at regular intervals

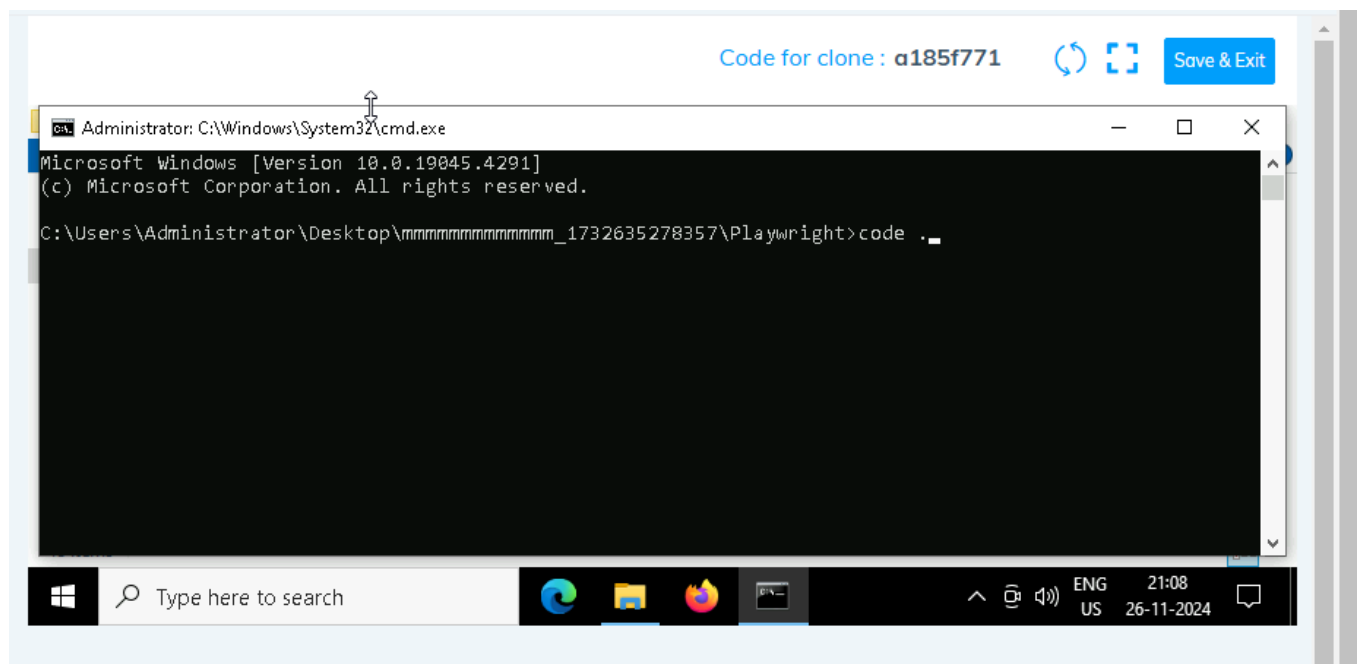


Mymedic automation using playwright

Open command prompt with its location and use below command: code .

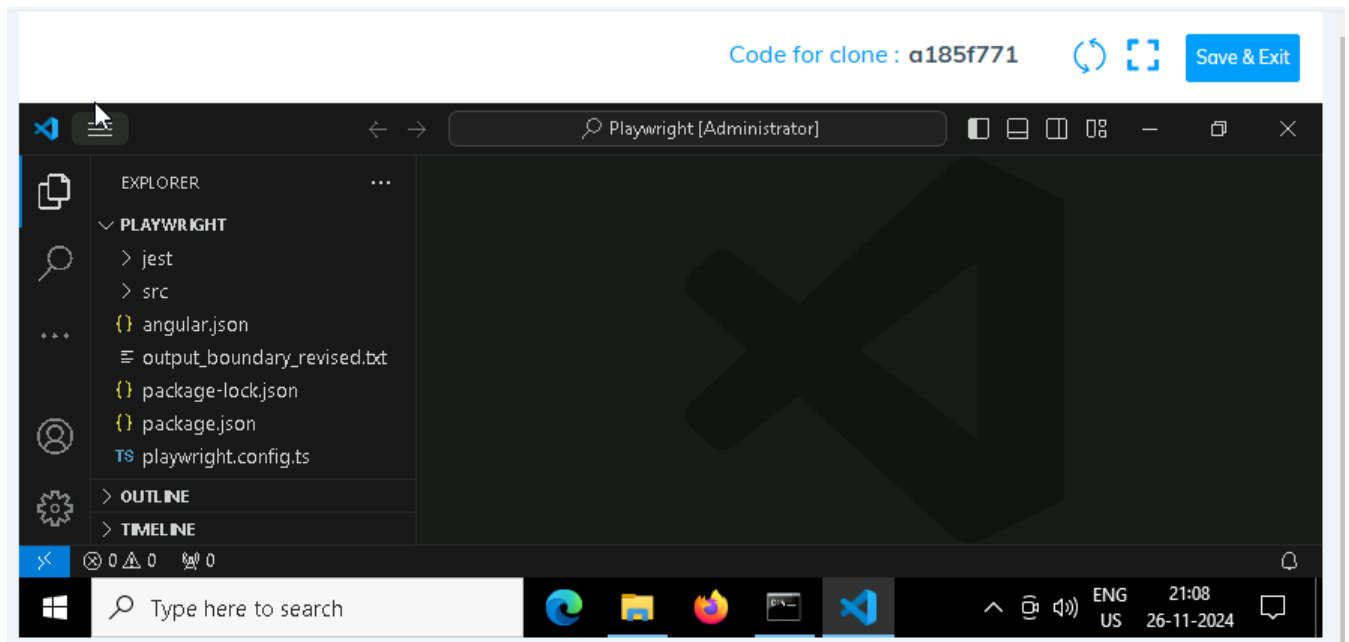


My medic automation using playwright



HRMS automation using playwright

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



HRMS automation using playwright

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

```
npm install
```

5. Install playwright in the Playwright folder path:

```
npx playwright install
```

6. Run the Tests in the Playwright folder path:

```
npx playwright test ./src/tests/PL2_testcases/yaksha.spec.ts
```