YAKSHA HEALTH APP WITH TYPESCRIPT AND PLAYWRIGHT PL1-4

Mymedic automation using playwright

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:/opensource-demo.orangehrmlive.com

---

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	- AdminPage - LeavePage - LoginPage - MyInfoPage	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 the "Logout" functionality**	1.Login using valid credentials 2. Click on Profile icon in top right 3. Click on the Logout button from dropdown. **It must implement this login functionality at first and then implement any other test case.**	**Reference path** \src\pages\LoginPage **methods** performLogin(), performLogout() **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.**	User should get logged out
2	Verify New holidays could be created	1.Navigate to URL with Valid Credentials 2. Navigate to "Leave" tab 3. Click on configure dropdown 4. Click Holiday option from dropdown 5. Click on "Add" Button 6. Fill the required fields 7. Click on save Button 8. Verify the Holiday gets added in the Table	**Reference path** \src\ pages**LeavePage** **methods** performLogin(), createNewLeaveRequest() **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.**	New Holiday should get added to the list
3	Verify an error appears when password and confirm	1. Login using valid credentials 2. Click on "Admin" tab in sidebar 3. Click on "Add" button 4. Enter different password in Password and confirm password input bar 5. Verify the "Password do not match" Error appears dynamically	**Reference path** \src\ pages**AdminPage** **methods** performLogin(), confirmPassword()	Password do not match error should appear

HRMS automation using playwright

	passwo rd mismat ch			
			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.**	
4	Verify Admin can add record	1. Login using valid credentials 2. Click on "Admin" tab in sidebar 3. Click on "Add" button 4. Fill the form 5. Click on save button	**Reference path** \src\ pages**AdminPage** **methods** performLogin() , AdminAdd () **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 alert saying "Saved successfully" appears on clicking save button
5	Verify User should be deleted from the table.	1. Log in to the application using valid credentials. 2. Trigger the user deletion process using the `deleteUser` method. 3. Capture the success message after deletion. 4. Assert that the deletion was successful using a utility assertion.	**Reference path** \src\ pages**AdminPage** **methods** PerformLogin(), deleteuser() **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.**	Ensures that an admin can successfully delete a user from the user management table.
6	Verify Image could be Uploaded as profile Pic	1. Login using valid credentials 2. Click on "My Info" tab in sidebar 3. Click on Profile pic 4. Click on "image add" 5. Upload the desired image from the device 6. Click on save button 7.Verify the image gets uploaded successfully	Reference path \src\pages**MyinfoPage** methods : uploadProfilePicture(), performLogin()	Image should get replaced

HRMS automation using playwright

7	Verify Admin can edit record	1. Login using valid credentials 2. Click on "Admin" tab in sidebar 3. Click on "edit" button	**Reference path** \PageObjects\Pages**AdminPage** **methods:** PerformLogin(),AdminEdit()	Verify edit user form appears
8	Verify the Admin can sort the records	1. Login using valid credentials 2. Click on "Admin" tab in sidebar 3. Click on "sort" button 4. Scroll down to choose the 'Descending' option 5. Verify the Table items get sorted according to the Descending order	**Reference path** ' \src\pages**AdminPage** **Methods** performLogin(), adminSort()	Table should get sorted accordingly
9	Verify new Tab opens on clicking the "Upgrade" button	1. Login using valid credentials 2. Click on "Upgrade" button in top right 3. Verify The user gets redirected to a new Tab opens	**Reference path** \src\ pages**AdminPage** **methods** PerformLogin(), upgrade(). **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.**	Upgrade to advance tab should open
10	Verify tooltip shows up when hovered over "Help" button (Question mark	1. Login using valid credentials 2. Click on "Admin" button in sidebar 3. Hover upon "?" button 4. Verify the Tooltip text is Displayed Dynamically **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** PerformLogin(), upgrade() **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.**	"Help" should be displayed on hovering over the "?" button in top right

HRMS automation using playwright

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.

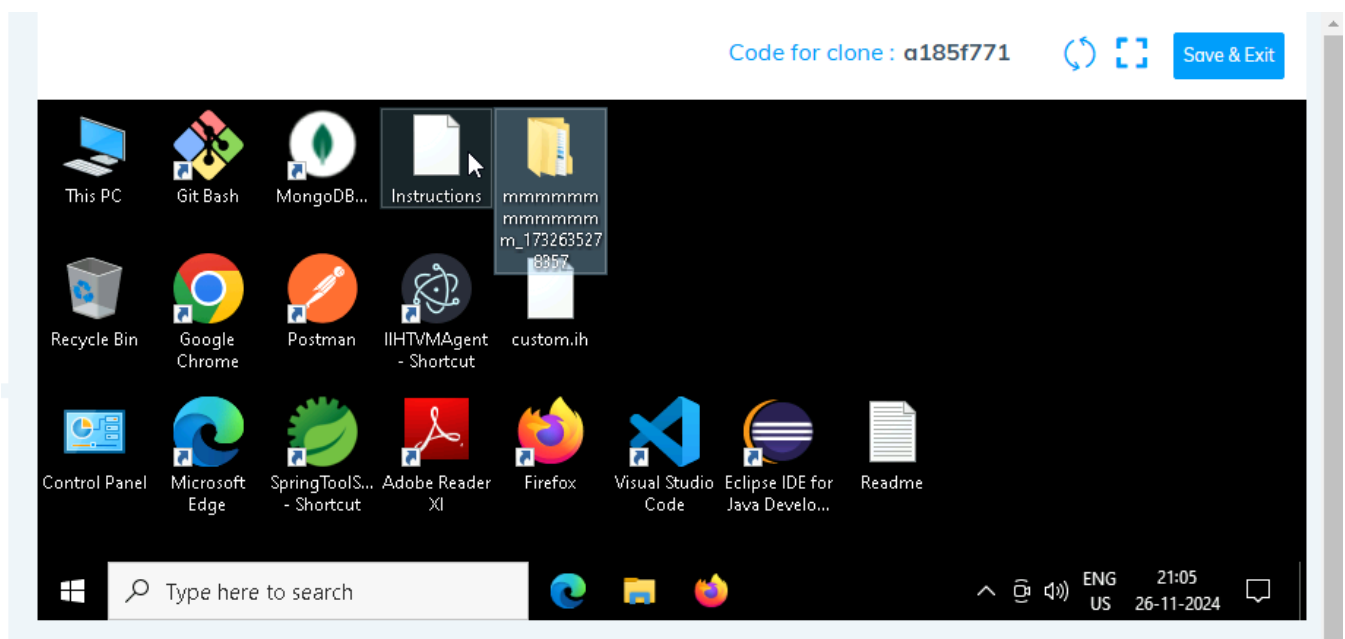## IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS
1. Associates should have written clean code that is readable.
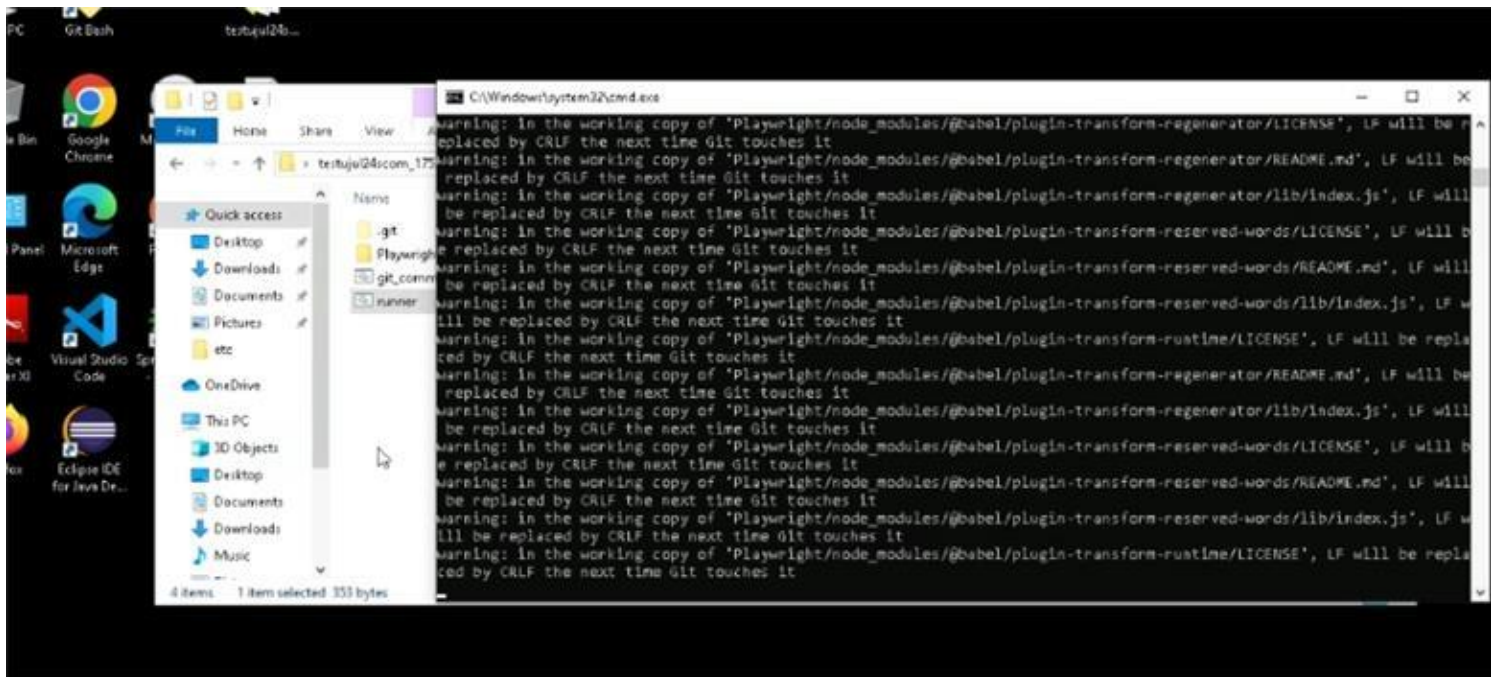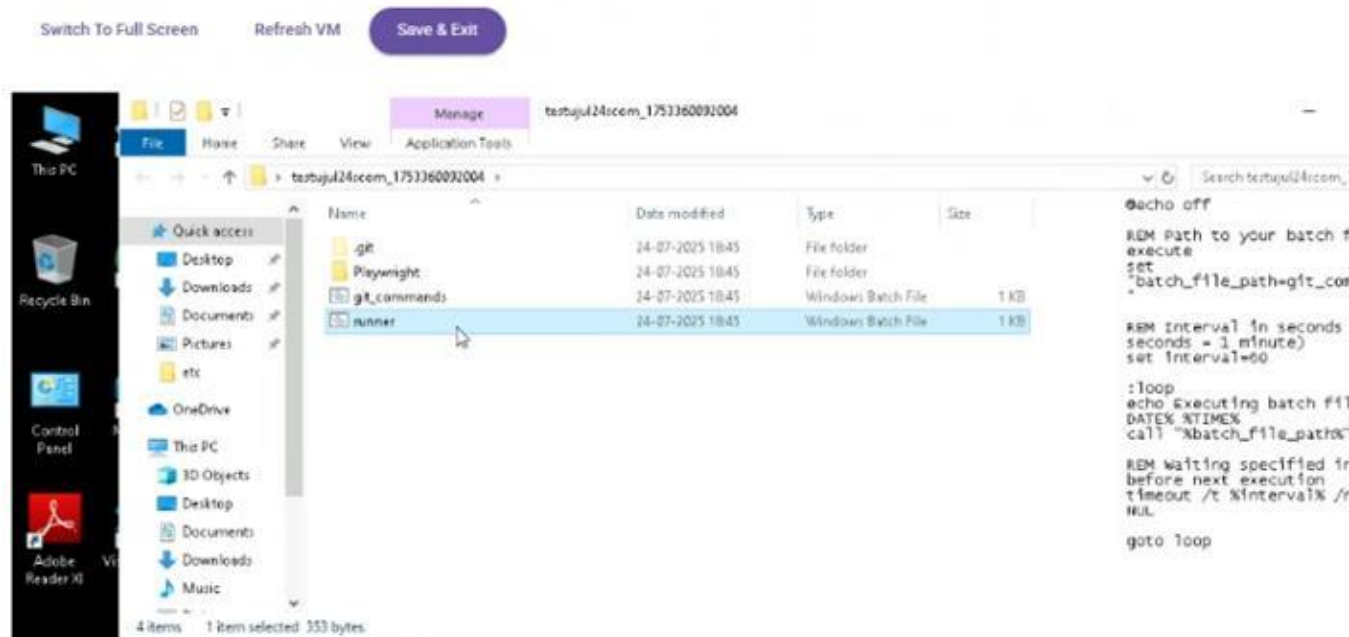2. Associates need to follow SOLID programming principles.

Execution Steps:
Steps for Execution:
1. **Please open the folder created on desktop with the email name you used to login.**
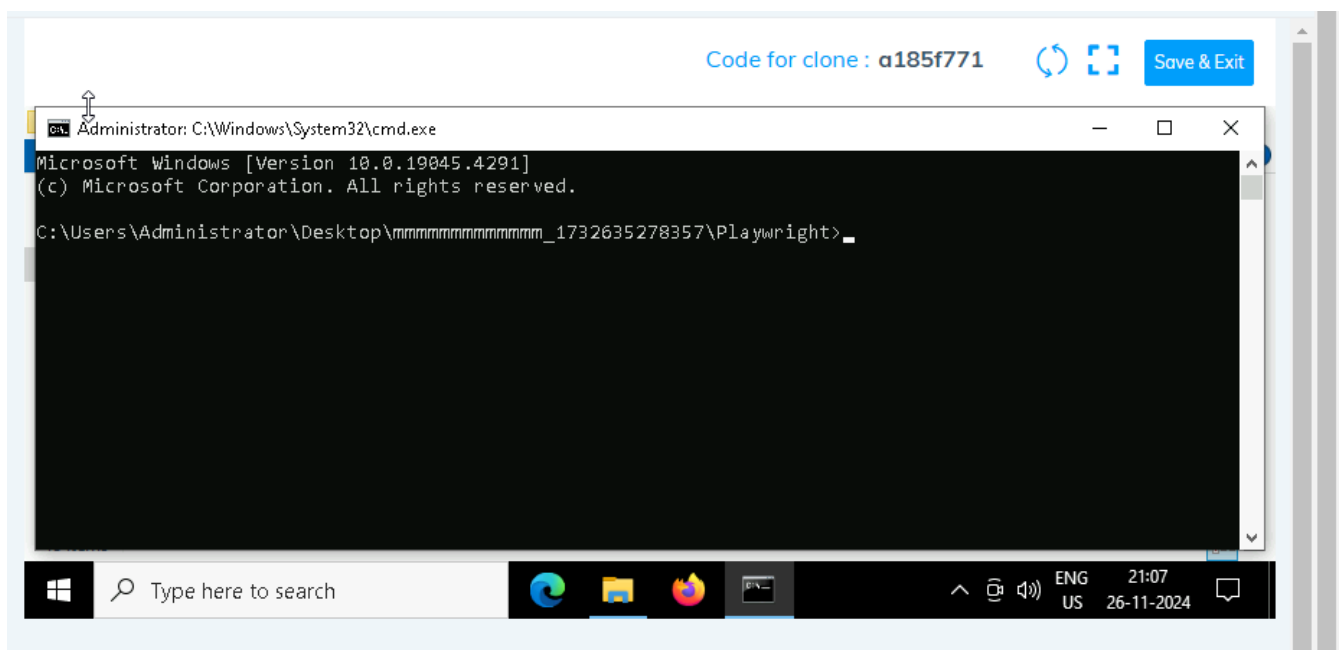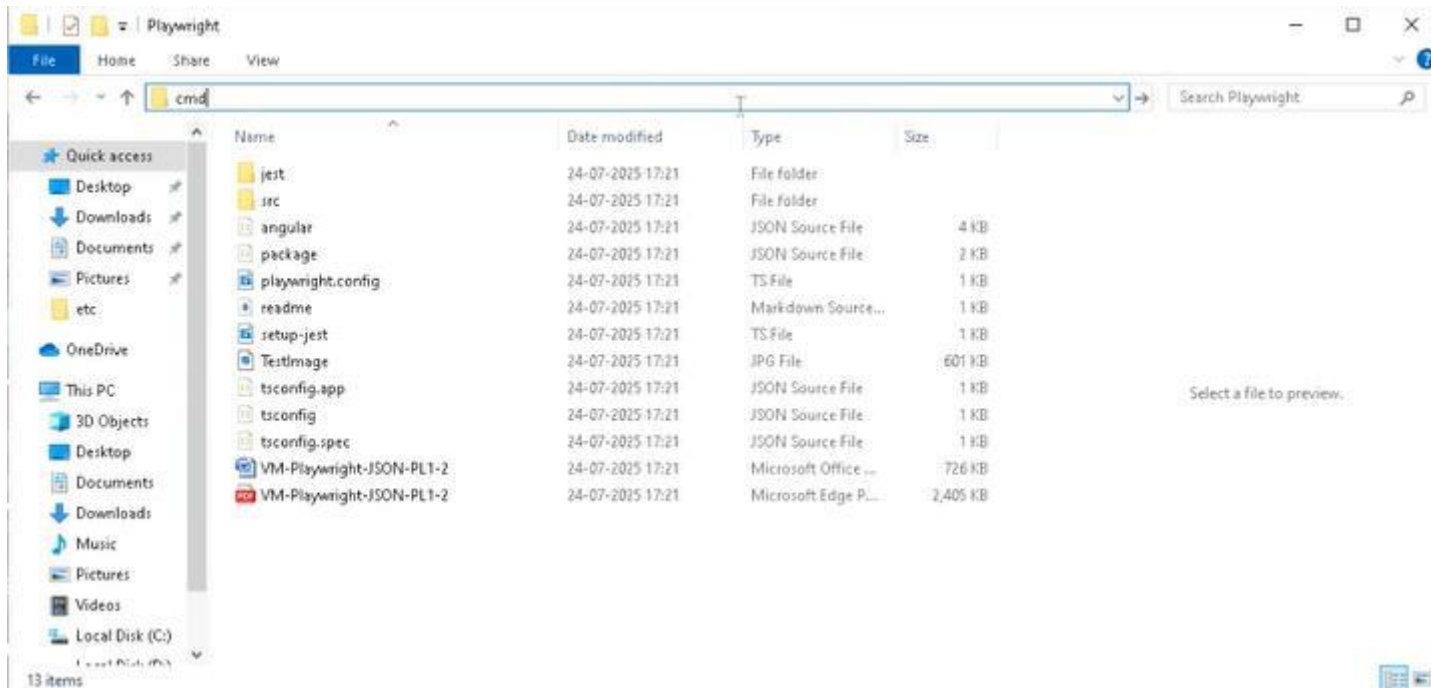

HRMS automation using playwright

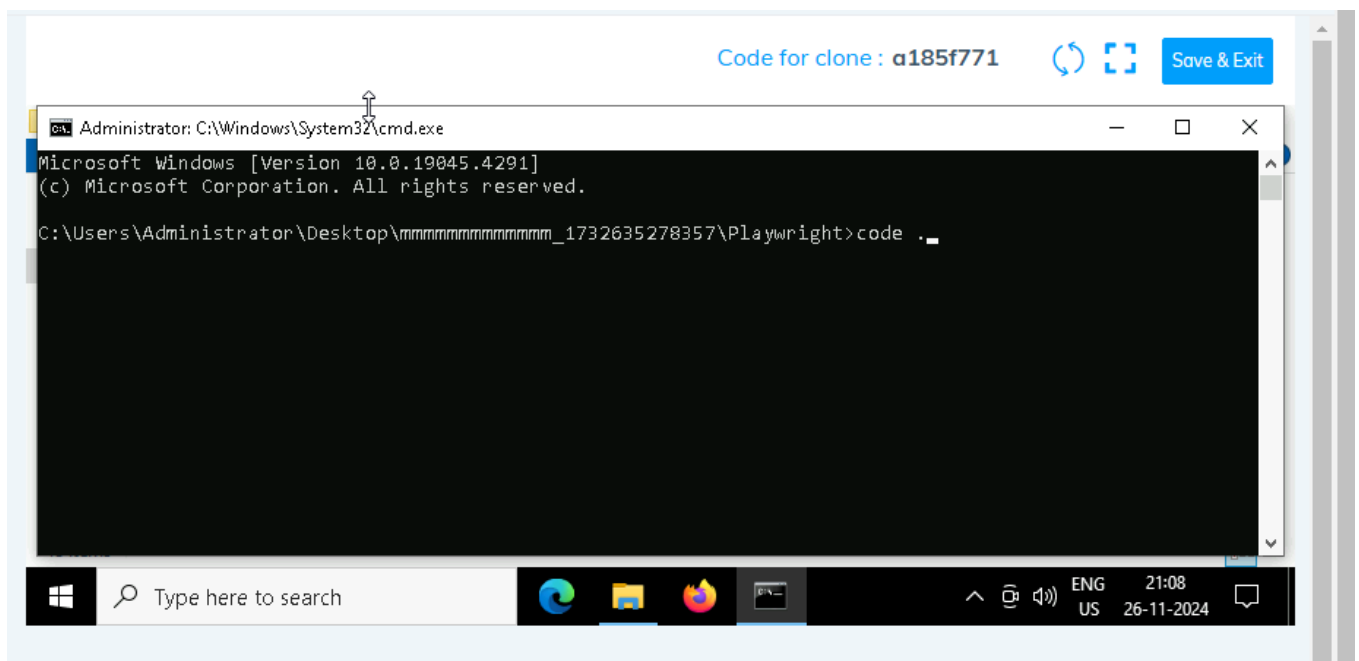**2. Execute this "runner" file. This will keep pushing the code at regular intervals**
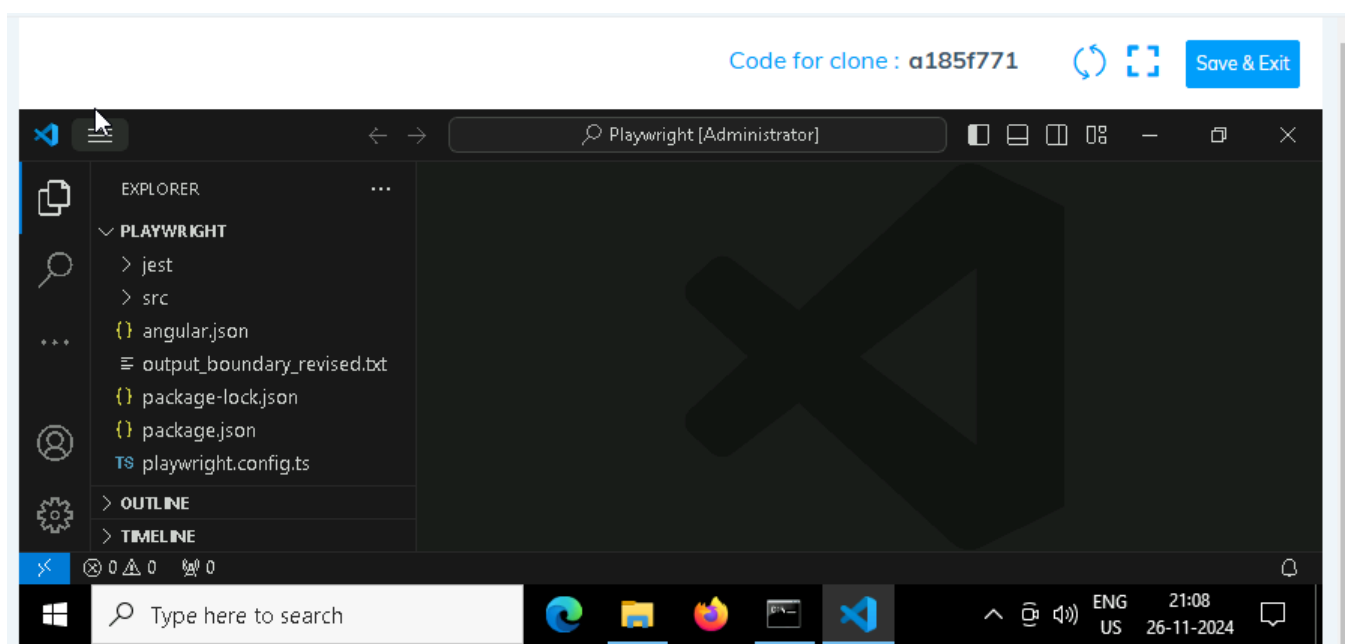

Mymedic automation using playwright

3. **Go into the Playwright folder and Open command prompt with its**

    location and use below command: code .

HRMS automation using playwright

Administrator: C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop\mmmmmmmmmmmmmm_1732635278357\Playwright>code .
```

Type here to search

ENG
US   21:08
26-11-2024

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

Playwright [Administrator]

EXPLORER

∨ PLAYWRIGHT
  > jest
  > src
  {} angular.json
  ≡ output_boundary_revised.txt
  {} package-lock.json
  {} package.json
  TS playwright.config.ts

> OUTLINE
> TIMELINE

Type here to search

ENG
US   21:08
26-11-2024

HRMS automation using playwright

5. **Install all dependencies in the Playwright folder path using:**

     npm install

6. **Install playwright in the Playwright folder path:**

     npx playwright install

7. **Run the Tests in the Playwright folder path**:

          npx playwright test ./src/tests/PL1_testcases/yaksha.spec.ts