YAKSHA HEALTH APP WITH TYPESCRIPT AND 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](https://healthapp.yaksha.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. |

|  |  |  |
| --- | --- | --- |
| src\ pages | * AdminPage * BuzzPage * LeavePage * LoginPage * MyInfoPage * PIMpage | 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 "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**  performLogin(), delete()  **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 should get Deleted |
|  |  |  | **Reference path** |  |
| 2 | Verify "Get" help button is functional | 1. Login using valid credentials 2. Click on "Question mark icon" button | \src\ pages\**LoginPage methods**  performLogin(), getUrl()  **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 be redirected to new URL |
|  |  |  | **Reference path** |  |
| 3 | Verify List of Reports | 1. Login using valid credentials 2. Click on "PIM" in the sidebar 3. Click on the Employee List tab 4. check for presence of a list | \src\ pages\**PIMpage**  **methods**  performLogin(), getEmpList () | A list of employees should appear |
|  |  |  | **You can use highlightElement method present in** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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.** |  |
|  |  | 1. Login using valid credentials 2. Click on "PIM" in the sidebar 3. Click on the Employee List tab 4. Enter the Username to search  5. Click the "Search" button | | **Reference path** |  |
|  |  | \src\ pages\**PIMpage** |  |
| 4 | Verify Employee List Search functionality | **methods**  performLogin(), verifyEmp()  **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.** | Only searched keyword should be present in the list |
|  |  |  | | **Reference path** |  |
| 5 | Verify Primary Colour of corporate branding could be changed | 1. Login using valid credentials 2. Click on "admin" in the sidebar 3. Click on the "Corporate branding" tab 4. Click the color circle after primary color label 5. Enter the hexacode of the color 6. Click on Publish button | | \src\ pages\**AdminPage**  **methods**  PerformLogin()*,* primaryColor()  **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.** | **Primary color should be set to desired one** |
| 6 | Verify Client Logo could not be uploaded above 1 mb | 1. Login using valid credentials 2. Click on "admin" in the sidebar 3. Click on the "Corporate branding" tab 4. Click the Client Logo Browse button 5. Select the desired file from the device and click open button | | Reference path  \src\pages\**AdminPage**  methods: performLogin(), companyLogo(). | **Incorrect Dimension Error should Display** |
| 7 | Verify Language change Functionality | 1. Login using valid credentials 2. Click on "admin" in the sidebar 3. Click on the "Configuration" tab 4. Click the Localization button 5. Select the desired Language from the dropdown (Chinese) 6. Click on Save button | | **Reference path**  \PageObjects\Pages\**AdminPage methods:**PerformLogin(), changeLanguage | **Successfully Updated alert should be displayed** |
| 8 | Verify Required Field Error in Leaves Tab displays when required field is empty. | 1. Login using valid credentials 2. Click on "leave" in the sidebar 3. Click on the "assign Leave" button 4. Click on the “Assign" button without filing any details in any field | | **Reference path**    \src\pages\**LeavePage** |  |
|  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case No.** | **Test Case Name** | **Test Steps to be performed** | **Path & Method Used** | **Expected Result** |
|  |  |  | **methods**  perfromLogin(), verifyLeaveField()  **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.** | **"Required" error should be displayed** |
|  |  | 1. Login using valid credentials 2. Click on "My info" in the sidebar 3. Scroll to the bottom 4. Click on "Download icon" button | **Reference path** |  |
|  |  | \src\ pages\**MyInfoPage** |  |
|  |  | **methods** |  |
| 9 | Verify My info Download Functionality | PerformLogin(), downloadInfo().  **You can use highlightElement method present in CommonMethods file to highlight the element before performing any action on it. It takes locator as** | File should get Downloaded |
|  |  | 1. Login using valid credentials 2. Click on "My info" in the sidebar 3. Enter the username, middle name, last name in the form 4. Click the save button>Refresh | **a parameter.**  **Reference path**  \src\pages\**MyInfoPage** |  |
| 10 | Verify User details could be Updated |  | **methods**  PerformLogin(), verifyUser() | User name should get Updated successfully |
|  |  | **Note :- If you cannot find the locator for “Proceed” button, please try to keep the page at 75%-80%.** | **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.** |  |
|  |  |  |  |  |

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. **Code Quality/Optimizations**

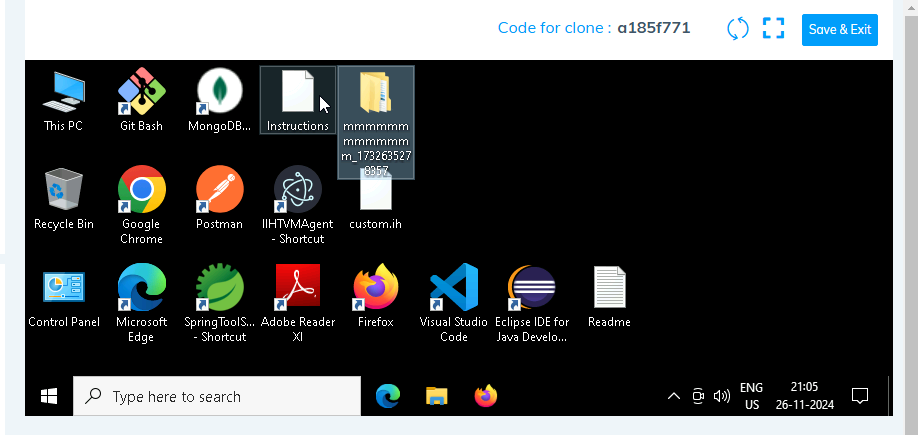
# Associates should have written clean code that is readable.

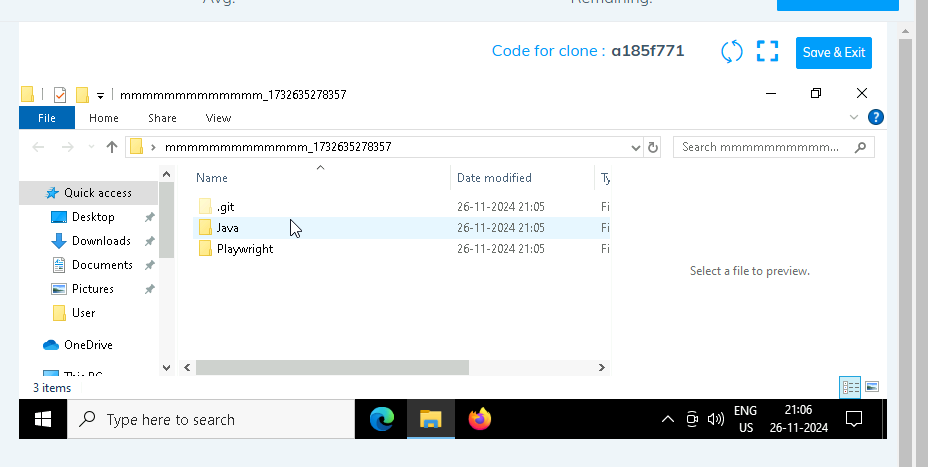
* + 1. Associates need to follow SOLID programming principles.

**Execution Steps:**

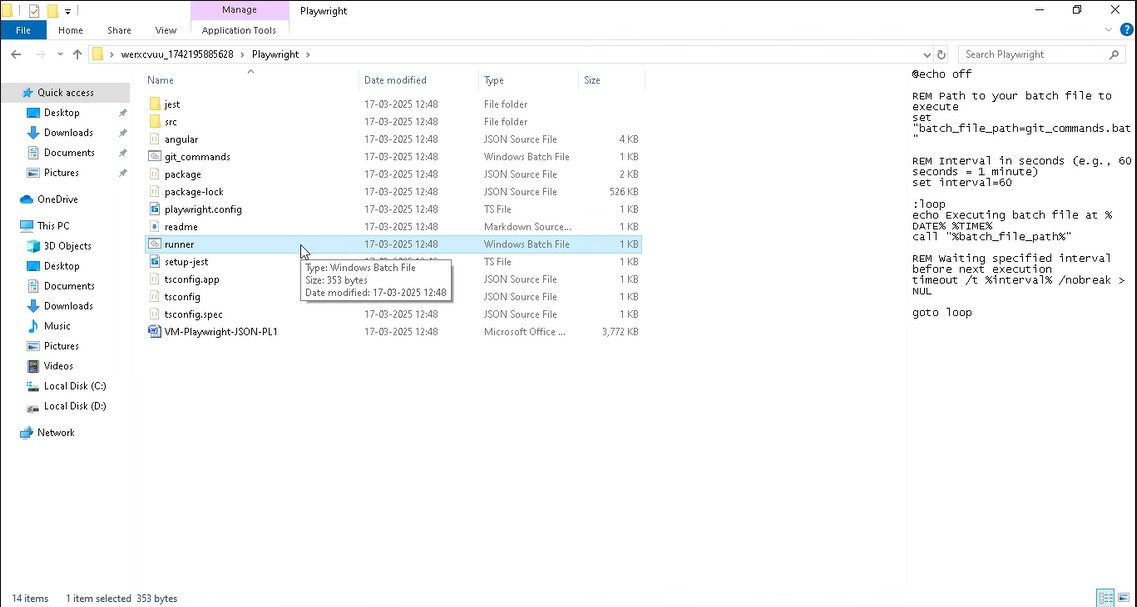
**Steps for Execution:**

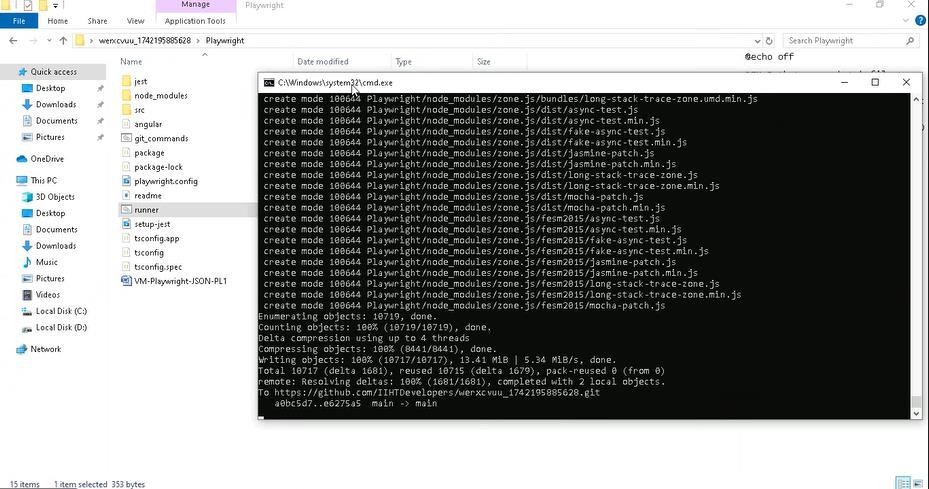
1. **Please open the folder created on desktop with the email name you used to login.**

****

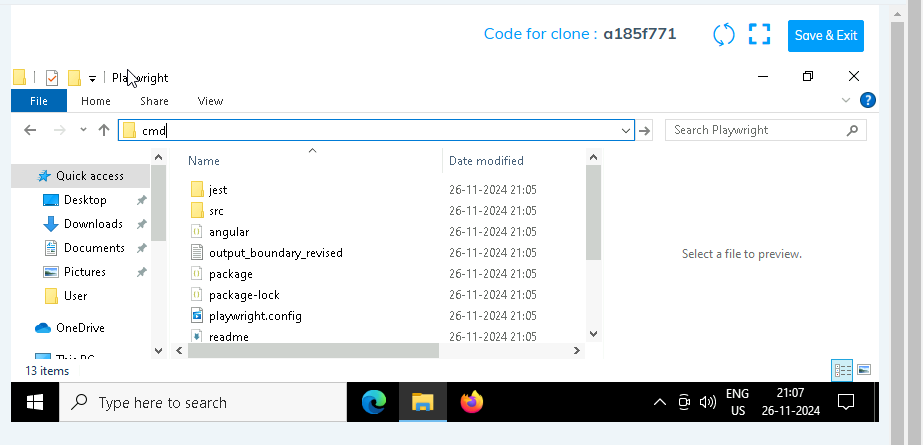


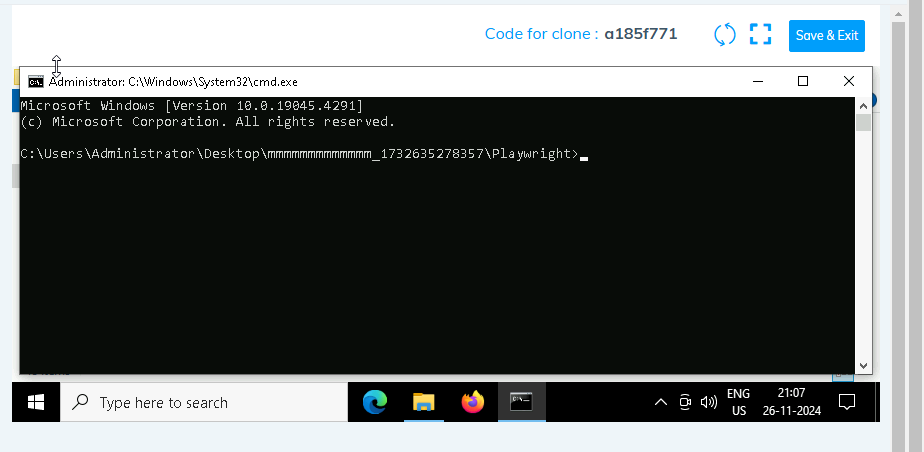
1. **Go into the Playwright folder and execute this “runner” file. This will keep pushing the code at regular intervals**

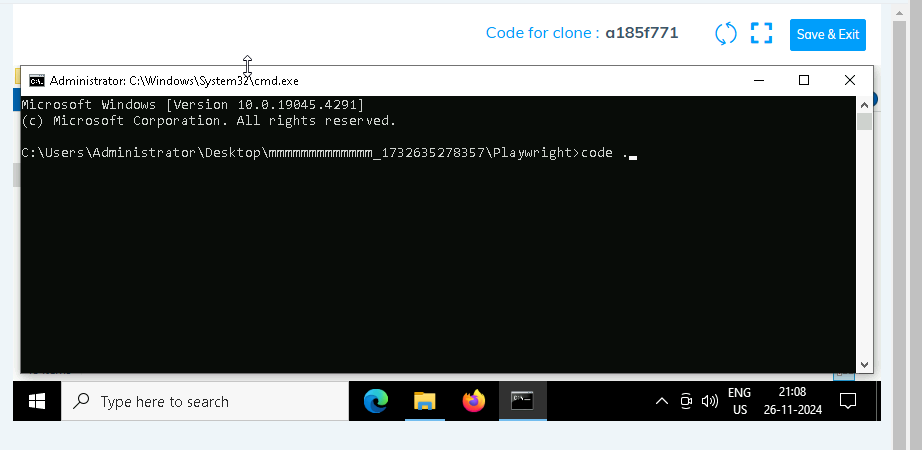


****

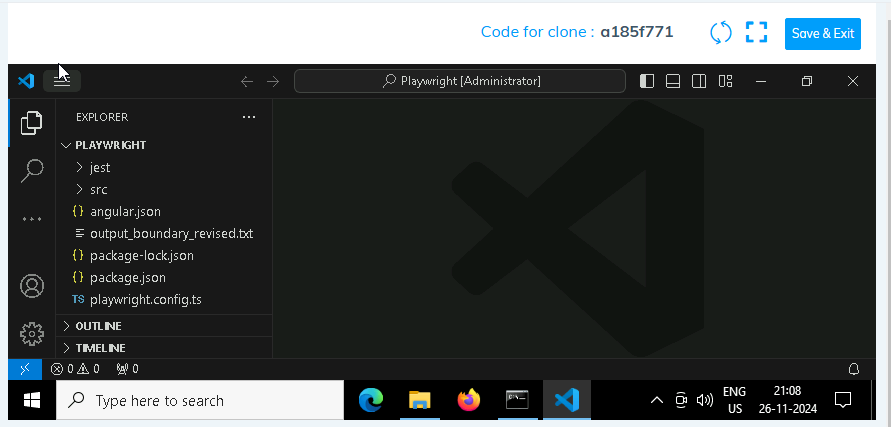
1. **Open command prompt with its location and use below command: code .**



****



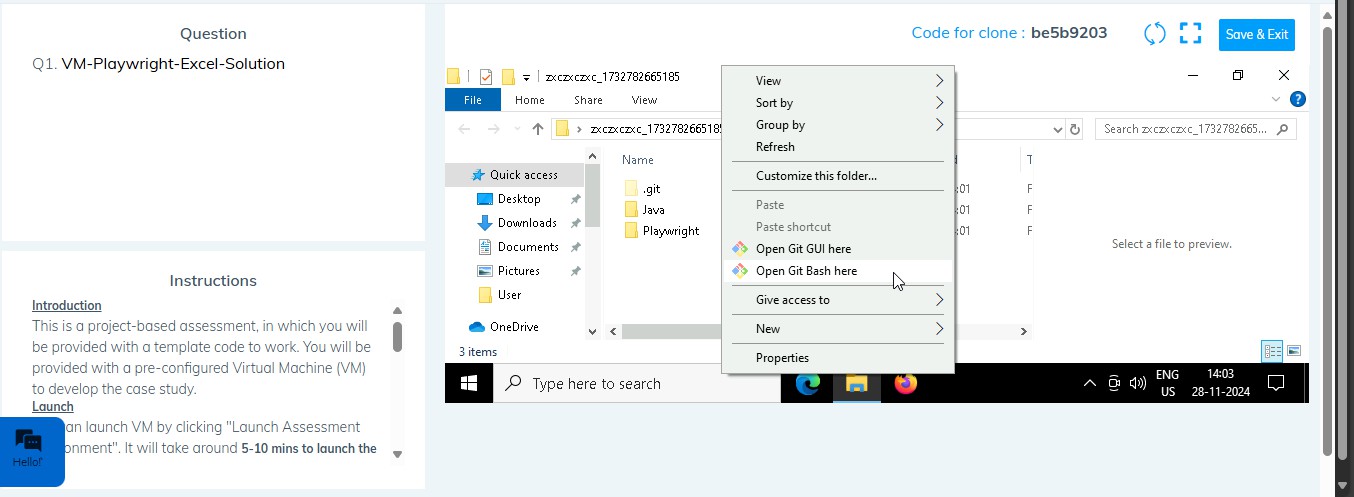
1. **Once VsCode is open. Please open the terminal in Playwright folder:**

****

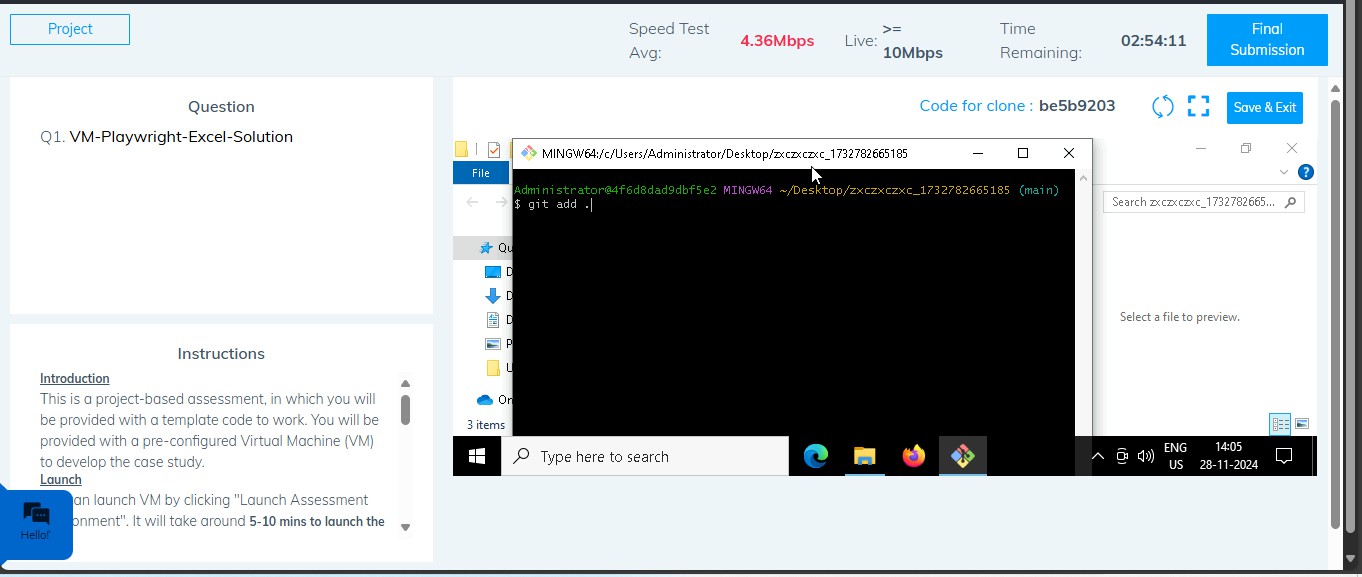
1. **Install all dependencies in the Playwright folder path using: npm install**
2. **Install playwright in the Playwright folder path: npx playwright install**
3. **Run the Tests in the Playwright folder path**:

**npx playwright test ./src/tests/PL1\_testcases/yaksha.spec.ts**

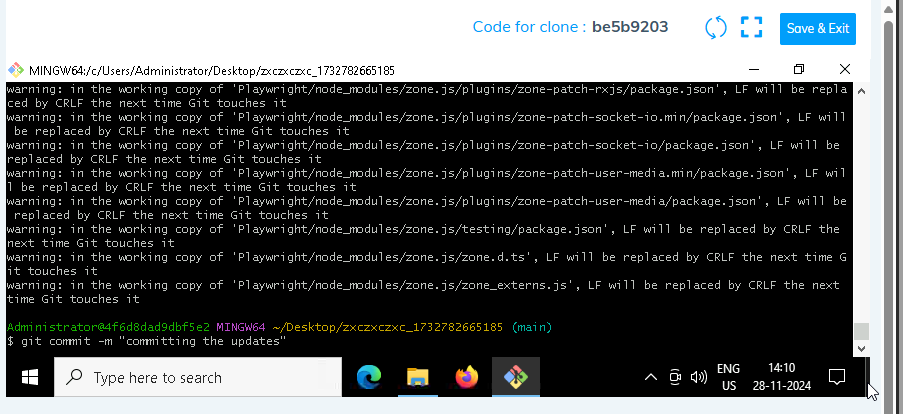
1. **Once you have executed the test cases. Now it is necessary to push your code to git. For this, please go inside the folder created on desktop with the email id you have used to login and then:**
   1. **Open gitbash**

****

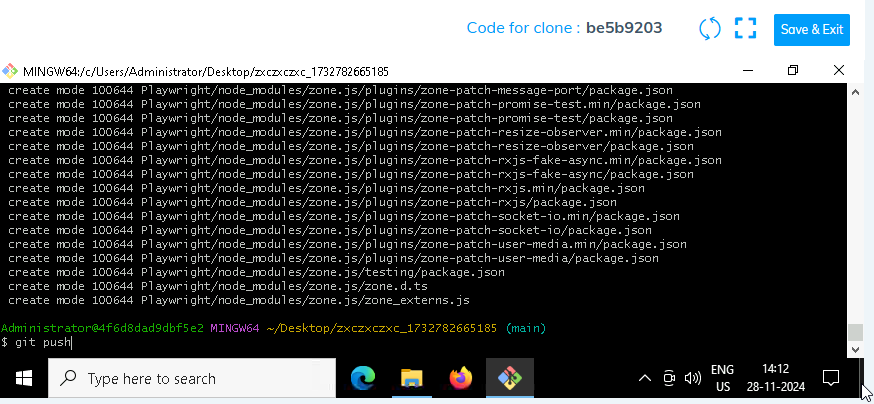
* 1. **Add all files**



* 1. **Commit the changes**



* 1. **Push the changes**

****