Automate test cases for PriceLabs

Objective: Demonstrate your ability to understand and automate specific features of the PriceLabs application, including both UI and API testing. Focus on the Multicalendar DSO feature and one additional feature of your choice.

Description:

Multicalendar DSO Feature:

- Identify Test Cases:
  - **Functional Tests:** At least 3 key functional test cases.
  - **End-to-End Tests:** At least 2 end-to-end test cases.
  - **Negative Tests:** At least 2 negative test cases.
- **Automate UI Test Cases:** Use Cypress.io to automate the identified test cases.

Choose Another Feature:

- **Identify Test Cases:**
  - **Functional Tests:** At least 3 key functional test cases.
  - **End-to-End Tests:** At least 2 end-to-end test cases.
  - **Negative Tests:** At least 2 negative test cases.
- **Automate UI Test Cases:** Use Cypress.io to automate the identified test cases.

API Test Cases:

- **Identify API Endpoints:** Determine relevant API endpoints for the chosen features.
- **Write API Test Cases:** Create at least 3 API test cases to validate the functionality of these endpoints. This could include:
  - **Functional Tests:** Verify correct responses for valid requests.
  - **Negative Tests:** Verify appropriate error responses for invalid requests.
  - **Integration Tests:** Ensure that the API integrates correctly with the frontend (if applicable).
- **Automate API Test Cases:** Use a tool like Postman or a JavaScript testing library (e.g., Cypress's `cy.request`) to automate these tests.

Create a Repository: Set up a public repository on GitHub (or any other platform) and upload your automation scripts for both UI and API tests.

Evaluation Criteria:

- **UI Test Cases:** Depth and breadth of test case coverage, quality, readability, and maintainability of the code, effective use of Cypress.io features, and clarity of documentation.
- **API Test Cases:** Correctness of API tests, handling of various response types, coverage of different scenarios, and use of testing best practices.

Additional Instructions:

- **Environment Setup:** Ensure your test scripts include environment setup steps, such as installing dependencies and configuring the test environment.
- Test Data Management: Use appropriate test data management techniques to ensure tests are repeatable and reliable.

Test Reporting: Implement test reporting for both UI and API tests, and provide screenshots or logs as needed.
Automation strategy: Use POM (Page object modal)
UI components: Cover min 5 different types of UI components.
Validation: Proper use of assertions.

Submission:

Please submit the repository links along with any additional notes or explanations. Ensure your submission includes detailed setup and execution instructions, as well as any necessary credentials or configurations. The expected time to complete this assignment is 3 days.