1. Overview of the Tests

The Cypress tests are designed to validate the functionality and UI elements of the homepage. They ensure that key components, such as the hero section, dropdowns, date pickers, passenger selector, and flight search form, work as expected. These tests simulate real-world user interactions to verify the homepage's behavior.

---

2. Test Coverage

The tests are implemented in the `homepage.cy.js` file located in the `cypress/e2e` directory. Below is a summary of the tests:

1. **Hero Section:**
   - Verifies the presence of the hero section, background image, title, subtitle, and airplane icon.
2. **Origin Dropdown:**
   - Opens the origin dropdown.
   - Searches for and selects "Toronto Pearson International Airport."
   - Verifies the selected origin is displayed.
3. **Destination Dropdown:**
   - Opens the destination dropdown.
   - Searches for and selects "Vancouver International Airport."
   - Verifies the selected destination is displayed.
4. **Date Pickers:**
   - Selects a departure date and return date.
   - Verifies the selected dates are displayed correctly.
5. **Passenger Selector:**
   - Increases the passenger count.
   - Verifies the updated passenger count is displayed.
6. **Search Button:**
   - Simulates a flight search by clicking the "Search Flights" button.
   - Verifies the application navigates to the `/search-results` page with the correct query parameters.

---

3. Prerequisites

Project Setup: Ensure the application is set up and all dependencies are installed:
bash
Copy code

```
npm install
```

1.

Start the Development Server: Run the server in development mode:
bash
Copy code
```
npm run dev
```

2.

Cypress Installation: Ensure Cypress is installed in the project:
bash
Copy code
```
npm install cypress --save-dev
```

3.

---

4. Running the Tests

Open Cypress Test Runner: Use the following command to open the Cypress GUI:
bash
Copy code
```
npx cypress open
```

1. In the GUI, navigate to `homepage.cy.js` and click to run the tests.

Run Tests in Headless Mode: To run the tests in headless mode (e.g., for CI/CD):
bash
Copy code
```
npx cypress run --spec "cypress/e2e/homepage.cy.js"
```

2.

---

5. Description of Tests

Hero Section Test

- **Purpose:** Ensures the hero section is displayed correctly with all expected elements.
- **Steps:**
  1. Verify the presence of the section element.
  2. Check for the title "Your Airline Name."
  3. Validate the subtitle "Find your next dream destination."
  4. Ensure the airplane icon is present.

Origin Dropdown Test

- **Purpose:** Validates the functionality of the origin dropdown.
- **Steps:**
    1. Click the "Origin" dropdown to open it.
    2. Search for "Toronto" in the input field.
    3. Select "Toronto Pearson International Airport."
    4. Verify that "Toronto (YYZ)" is displayed.

Destination Dropdown Test

- **Purpose:** Validates the functionality of the destination dropdown.
- **Steps:**
    1. Click the "Destination" dropdown to open it.
    2. Search for "Vancouver" in the input field.
    3. Select "Vancouver International Airport."
    4. Verify that "Vancouver (YVR)" is displayed.

Date Pickers Test

- **Purpose:** Ensures the departure and return date pickers work correctly.
- **Steps:**
    1. Set the departure date to `2024-10-21`.
    2. Verify that the departure date input reflects the correct date.
    3. Set the return date to `2024-10-25`.
    4. Verify that the return date input reflects the correct date.

Passenger Selector Test

- **Purpose:** Tests the passenger selector functionality.
- **Steps:**
    1. Open the passenger selector.
    2. Increase the number of adult passengers.
    3. Verify that the passenger count updates to "2 Passengers."
    4. Close the passenger selector.

Search Button Test

- **Purpose:** Simulates a flight search and verifies navigation.
- **Steps:**
    1. Click the "Search Flights" button.
    2. Verify that the application navigates to `/search-results`.
    3. Check that the query parameters in the URL reflect the selected inputs (origin, destination, dates, and passengers).

6. Notes

- **Environment Variables:** Ensure any required environment variables (e.g., `BASE_URL`) are properly configured before running the tests.
- **Headless Mode:** Use the `--spec` option to run specific tests if necessary.
- **Debugging:** If a test fails, Cypress provides screenshots and detailed logs in the Test Runner for debugging.

7. Example Output

Upon successful execution of the tests, you should see:

- All tests pass with green checkmarks.
- Cypress Test Runner displays detailed steps and assertions for each test.