Flight ticketing

Requirements

- Users may search for flights based on arrival and/or departure city
- Users may or may not add a specific seat selection to their flight.
- Users are able to pay for their flight
- Users can sort flights by price
- Users can create accounts and sign in
- Users can view their booked flights

Test Cases

User Registration, New User

- **Objective**: to verify users can register for a new account
- **Action**: In the user registration page, fill in a username that does not currently exist and password and click “Register” button
- **Assert**: Check that the new user is registered and logged in (redirected to main page)

User Registration, Existing User

- **Objective**: to verify users can register for a new account
- **Action**: In the user registration page, fill in a username that already exists and password and click “Register” button
- **Assert**: Check that the new user is not registered and a message is displayed notifying the user that an account with that username already exists.

User Login

- **Objective**: to verify users can login to an existing account
- **Action**: In the user login page, fill in a known existing username and correct password and click “Login” button
- **Assert**: Check that the user is logged in (redirected to main page)

Incorrect Password

- **Objective**: to verify users can’t log in if their password is incorrect.
- **Action**: In the user login page, fill in a known existing username and an incorrect password, then click the “Login” button.
- **Assert**: Check that the user is not logged in and a message is displayed notifying the user of an incorrect password.

Flight Search

- **Objective**: to verify users can search for flights based on departure and arrival cities
- **Action**: In the main page, enter a departure and arrival and click “Search”
- **Assert**: Check that relevant flights show up on the results page.

Flight Sort

- **Objective**: to verify users can sort flight search results based on price.
- **Action**: In the Search Result page, click the “Sort by Price” button.
- **Assert**: Check that relevant flights show up on the results page in ascending order based on price.

Seat Select

- **Objective**: to verify users can select a seat for their flight.
- **Action**: In the Seat Select page, select a seat and click “Next”.
- **Assert**: Check that the chosen seat shows up with the rest of the booking information on the Confirm and Pay page.

Random Seat Select

- **Objective**: to verify users can skip selecting a seat and still have a valid booking.
- **Action**: In the Seat Select page, click “Skip”
- **Assert**: Check that a random available seat shows up with the rest of the booking information on the Confirm and Pay page.

Confirm and Pay

- **Objective**: to verify users can finalize their bookings.
- **Action**: In the Confirm and Pay page, enter payment information and click the “Book Now” button.
- **Assert**: Check that the Thank You page is displayed

No Free Flights

- **Objective**: to verify users can’t finalize their bookings without payment information.
- **Action**: In the Confirm and Pay page, do not enter payment information and click the “Book Now” button.
- **Assert**: Check that no redirect happens and that a message is displayed that says “Please enter payment information”.

View Bookings

- **Objective**: to verify users can view their existing bookings.
- **Action**: In the main page, click the “View My Bookings” button.
- **Assert**: Check that the user's flights show up on the My Bookings page, including departure city, date, and time, arrival city, date, and time, seat code.

UI diagrams

https://www.figma.com/community/file/1422416248204971157/flight-booking-system