CISC 327 Group 34 - CH

Assignment 1

September 30, 2024

Hayden Jenkins 20344883
Ryan Van Drunen 20331633
Madison MacNeil 20285877

UI Designs in Figma:

https://www.figma.com/design/boEAccAG4L1xxmIm9Ry1vR

Landing page UI adapted from:
Howler, Asif. “Flight Booking UI - Freebie (Community).” *Figma*, Apr. 2024, www.figma.com/design/n6gLJEoI8gONURIwXHzmoS

Seat booking UI adapted from:
Tong, Crystal, et al. “ Tripma - Flight booking web app.” *Figma,* Sep. 2023, https://www.figma.com/design/SGVIjtPzbi0apJPeucl2tG

Finalized Requirements:

Requirement 1: Users will be able to reach the registration page from the landing page.

Test Case Name: Landing page to Registration page

Objective: To verify that users can reach the registration page from the landing page.

Action: From the landing page, the user clicks the “Sign Up” button in the navbar.

Assert: Check that the user is redirected to the registration page.

Requirement 2: Users should be able to reach the registration page from the search results page.

Test Case Name: Search Results page to Registration page

Objective: To verify that users can reach the registration page from the search results page.

Action: From the search results page, the user clicks the “Sign Up” button in the navbar.

Assert: Check that the user is redirected to the registration page.

Requirement 3: Users should be able to reach the registration page from the payment page.

Test Case Name: Payment page to Registration page

Objective: To verify that users can reach the registration page from the payment page.

Action: From the payment page, the user clicks the “Sign Up” button in the navbar.

Assert: Check that the user is redirected to the registration page.

Requirement 4: Users should be able to access the slide-out login modal from the landing page.

Test Case Name: Login Modal from Landing page

Objective: To verify that users can access the login modal from the landing page.

Action: From the landing page, the user clicks the “Login” button in the navbar.

Assert: Check that the user is shown the slide-out login modal.

Requirement 5: Users should be able to access the slide-out login modal from the search results page.

Test Case Name: Login Modal from Search Results page

Objective: To verify that users can access the login modal from the search results page.

Action: From the search results page, the user clicks the “Login” button in the navbar.

Assert: Check that the user is shown the slide-out login modal.

Requirement 6: Users should be able to register for a new account.

Test Case Name: Account Registration

Objective: To verify that users can successfully register and create new a account.

Action: From the registration page, the user enters all required valid information and clicks the “Sign Up” button.

- **Input Parameters:** Full name, email, phone number, password and confirmation password.

Assert: Check that the user is successfully registered and redirected to the landing page with a success message and that the user account is created.

Requirement 7: Users should be able to sign in with an existing account.

Test Case Name: Account Sign In

Objective: To verify that users can successfully sign in with an existing account.

Action: From the slide-out login modal, the user enters a valid email and password or successfully authenticates with Google and clicks the “Sign In” button.

- **Input Parameters:** Account email and password

Assert: Check that the user is successfully logged in, a success message is displayed and the login modal is closed.

Requirement 8: A user should be able to switch the search filter from one-way to round trip flights.

Test Case Name: One Way to Round Trip Filter

Objective: To verify that users can change the search filter from one-way to round-trip flights.

Action: On the landing page, the user clicks the “Round Trip” button above the search filter.

Assert: Check that the search filter switches from one-way to round-trip flights.

Requirement 9: Users should be able to switch the search filter from round trip to one-way flights.

Test Case Name: Round Trip to One-Way Filter

Objective: To verify that users can change the search filter from round trip to one-way flights.

Action: On the landing page, the user clicks the “One Way” button above the search filter.

Assert: Check that the search filter switches from round trip to one-way flights.

Requirement 10: Users should be able to reach the registration page from the login modal.

Test Case Name: Login Modal to Registration page

Objective: To verify that users can reach the registration page from the login modal.

Action: From the slide-out login modal, the user clicks the “Sign Up” button below the input fields.

Assert: Check that the user is redirected to the registration page.

Requirement 11: Users should be able to reach the sign-in page from the registration page.

Test Case Name: Registration Page to Login Page

Objective: To verify that users can reach the login page from the registration page.

Action: From the registration page, the user clicks the “Sign In button” below the input fields.

Assert: Check that the user is redirected to the sign-in page.

Requirement 12: Users should be able to reset their password from the login page.

Test Case Name: Reset Password from Login Page

Objective: To verify that users can reset their password from the login page.

Action: From the login page, the user clicks the “Forgot Password?” button.

Assert: Check that the user is redirected to the Forgot Password page.

Requirement 13: Users should be able to reset their password from the login modal.

Test Case Name: Reset Password from Modal

Objective: To verify that users can reset their password from the login modal.

Action: From the login modal, the user clicks the “Forgot Password?” button.

Assert: Check that the user is redirected to the Forgot Password page.

Requirement 14: From the landing page, users should be able to search for flights with filters that they select and apply.

Test Case Name: Landing Page Filtered Search

Objective: To verify that users can successfully search for flights from the landing page with applied filters.

Action: From the landing page, the user edits the search filters (departure city, arrival city, dates, one way or round trip), and then clicks the “Search Flights” button.

Assert: Check that the user is redirected to the search results page, and is only shown flights that match the selected filters.

Requirement 15: On the search results page, users will be able to swap the departure and arrival cities.

Test Case Name: Search City Swap

Objective: To verify that users can successfully swap the departure and arrival cities on the search results page.

Action: From the search results page, the user clicks the button with 2 orange arrows.

Assert: Check that the departure and arrival cities are swapped in the search filters and that a new search is automatically performed with the updated filters.

Requirement 16: On the search results page, users will be able to edit the search filters (departure city, arrival city, dates, one way or round trip), and refresh the search results.

Test Case Name: Search Refresh

Objective: To verify that users can successfully edit the search filters and refresh the search results.

Action: From the search results page, the user edits the search filters, then clicks the “Search” button.

Assert: Check that the search results page displays new flights that match the new filters.

Requirement 17: On the search results page, users will be able to click on a flight to see more information on the right side of the screen.

Test Case Name: Expanded Flight Information

Objective: To verify that users can successfully see more information of a selected flight.

Action: From the search results page, the user clicks on a flight option.

Assert: Check that the correct information corresponding to the selected flight is displayed on the right side of the screen in a pop-out modal.

Requirement 18: After clicking a flight on the search results page, users will be able to continue to flight booking.

Test Case Name: Flight Booking Redirection

Objective: To verify that users are successfully redirected to the flight booking page.

Action: From the search results page, after clicking on a specific flight, the user clicks the “Book My Ticket Now” button.

Assert: Check that the user is redirected to the flight booking page for the correct flight.

Requirement 19: From the flight booking page, users will be able to scroll up and down on the plane to select their seat.

Test Case Name: Flight Booking Plane Scroll

Objective: To verify that users can successfully scroll up and down on the plane.

Action: On the flight booking page, the user scrolls up and down.

Assert: Check that the plane is scrolling correctly, and that all seats are being shown to the user.

Requirement 20: In the flight booking page, users will be able to click on their desired seat, as long as it is not taken.

Test Case Name: Flight Booking Seat Selection

Objective: To verify that users can successfully select desired seats.

Action: On the flight booking page, the user clicks on a seat that is not taken.

Assert: Check that the selected seat changes colours and is marked as selected with the seat number change being reflected in the UI.

Requirement 21: From the flight booking page, users should be able to move to the seat selection for another passenger, if more than one passenger was selected, if they have filled in all the required information for passenger 1.

Test Case Name: Booking Next Passenger

Objective: To verify that users must input passenger information before continuing to the next page.

Action: On the flight booking page, the user clicks the “Next Passenger” button if they originally selected more than one passenger, after they have input all valid fields.

- **Input Parameters:** First name, Middle name, Last name, suffix, DOB, Passport number, email address, phone number, street address, apt number, province, zip code. Emergency contact information (Known traveler number, first name, last name, email address, phone number).

Assert: Check that the user is redirected to a new flight booking page, and the previous passenger information is saved.

Requirement 22: From the flight booking page, users should not be able to proceed to seat selection for another passenger once seats have been booked for all selected passengers. Additionally, users should be able to move on to the payment page or the next flight when booking round trips.

Test Case Name: Booking Passenger Completed

Objective: To verify that users are not prompted to book additional flight seats after seats have been booked for all selected passengers, and are then prompted to book their next flight, if they have selected a round-trip or pay at the payment page.

Action: On the flight booking page, after booking seats for all selected passengers, the user clicks the “Next Flight” button if they are to book a return flight or the “Continue” button if they selected a one-way flight.

Assert: Check that the user is redirected to a new flight booking page if booking a round-trip or to the payment page if booking a one-way flight, while saving the previous passenger information and ensuring that the user is not asked to select seats for any additional passengers.

Requirement 23: From the return flight booking page, users should be redirected to the payment page.
Test Case Name: Return Flight Booking Completed

Objective: To verify that users are redirected to the payment page after booking their return flight.
Action: On the return flight booking page, after booking seats for all selected passengers, the user clicks the “Continue” button.

Assert: Check that the user is redirected to the payment page.

Requirement 24: From the payment page, users will be able to pay for their flights using credit card, google pay, paypal, etc.

Test Case Name: Flight Payment Test

Objective: To verify that users can successfully pay for their flights.

Action: On the payment page, the user enters the required valid information such as personal details, card details, or opts for a different payment option, then presses the “Confirm and pay” button.

- **Input Parameters:** Personal details (Address line, City, Province, Postal code) and card details (Cardholder's name, Card Number, expiry date, CVV)

Assert: Check that the user is shown the Payment Confirmation modal with the correct payment confirmation information.

Requirement 25: From the payment confirmation modal, the user will be able to confirm their flights and complete their payment.

Test Case Name: Payment Confirmation

Objective: To verify that users can successfully confirm their flights and complete their payment.

Action: From the payment confirmation modal, the user checks the “I accept the cancellation policy”, then clicks the “Confirm and pay” button.

Assert: Check that the user’s payment is processed and they are shown the Email Notifications modal.

Requirement 26: From the Email Notifications modal, the user will be able to subscribe to flight alerts via email.

Test Case Name: Email Notifications

Objective: To verify that users can successfully subscribe to flight alerts.

Action: From the email notifications modal, the user checks the “I would like to subscribe to flight alerts” box, then clicks the “Next” button.

Assert: Check that the user is added to the email alert list for their flight.

Requirement 27: From the Email Notifications modal, the user will be able to download the itinerary.

Test Case Name: Itinerary Download

Objective: To verify that users can successfully download the itinerary.

Action: From the email notifications modal, the user clicks the “Download Itinerary” button.

Assert: Check that a PDF of the itinerary is prepared and is downloaded to the user’s system.

Requirement 28: From any page, if the user is logged in, they will be able to access their profile dashboard.

Test Case Name: Profile Dashboard

Objective: To verify that users can successfully access their profile dashboard from any page.

Action: From any page, the user clicks their profile picture in the navbar.

Assert: Check that the user is redirected to the profile dashboard.

Requirement 29: From the profile dashboard, the user will be able to access the edit profile page.

Test Case Name: Dashboard to Edit Profile

Objective: To verify that users can successfully reach the edit profile page.

Action: From the profile dashboard, the user clicks the “Edit Profile” button.

Assert: Check that the user is redirected to the edit profile page.

Requirement 30: From the edit profile page, the user will be able to edit their profile information.

Test Case Name: Edit Profile

Objective: To verify that users can successfully edit their account information.

Action: From the edit profile page, the user enters the required valid information, then presses the “Save” button.

- **Input Parameters(not all at once):** First name, Middle Name, Last name, Suffix, Date of birth, email address, phone number, street address, apt number, province, zip code.

Assert: Check that the account information is updated and a success message is displayed.

Requirement 31: From the profile dashboard, the user will be able to go back to the landing page.

Test Case Name: Dashboard to Landing

Objective: To verify that users can successfully navigate back to the landing page from the profile dashboard.

Action: From the profile dashboard, the user clicks the “Home” button.

Assert: Check that the user is redirected to the landing page.

Requirement 32: From the profile dashboard, the user will be able to cancel a flight.

Test Case Name: Flight Cancellation

Objective: To verify that users can successfully cancel a flight.

Action: From the profile dashboard, the user clicks the cancel icon on the desired flight card.

Assert: Check that the user is shown the cancellation confirmation modal.

Requirement 33: From the flight cancellation confirmation modal, the user will be able to confirm their cancellation.

Test Case Name: Flight Cancellation Confirmation

Objective: To verify that users can successfully confirm their flight cancellation.

Action: From the flight cancellation confirmation modal, the user clicks the “Confirm” button.

Assert: Check that the user’s flight is successfully canceled, and that they are issued a refund.

Requirement 34: From the profile dashboard, the user will be able to log out of their account.

Test Case Name: User Log Out

Objective: To verify that users can successfully log out of their account.

Action: From the profile dashboard, the user clicks the “Log Out” button.

Assert: Check that the user is successfully logged out and redirected to the landing page.