

Requirements Document - Group 43

Flight Ticket Booking System UI link:

<https://www.figma.com/design/Ar9JiwuYIAEIDd9dk8O8FV/flight-ticket-booking-system-UI?node-id=0-1&t=zEF7zslo7TmRvBfx-1>

Functional Requirements

Requirement 1:

Title: *User Registration*

Description:

The system shall allow the user to register for a new account by providing their name, email, and a password.

Acceptance Criteria/Test Cases:

Test Case 1: Successful registration

- **Objective:** To verify successful user registration given valid inputs
- **Preconditions:**
 - *User is on the registration page*
- **Test Steps:**
 - *Enter a valid name in the name field*
 - *Enter a valid email in the email field*
 - *Enter a valid password in the "password" field*
 - *Submit the registration form*
- **Expected Result:**
 - *The system creates a new user account and a confirmation message is displayed*

Test Case 2: Registration with already existing email

- **Objective:** To verify the system does not allow the user to make an account with an existing email
- **Preconditions:**
 - *User is on the registration page*
- **Test Steps:**
 - *Enter an email that's already registered*
 - *Submit the registration form*
- **Expected Result:**

- *An error message is displayed explaining that there is already a registered account using that email*
- *User stays on the registration page*

Test Case 3: Registration with empty fields

- **Objective:** To verify that the system prompts the user when email, name and or password fields are left empty
- **Preconditions:**
 - *User is on the registration page*
- **Test Steps:**
 - *Leave the “name”, “email” and “password” fields empty*
 - *Submit the registration form*
- **Expected Result:**
 - *An error message is displayed explaining that all fields are required*
 - *User stays on the registration page*

Requirement 2:

Title: *User Login*

Description:

The system shall allow the user to login to their account using their registered email and password

Acceptance Criteria/Test Cases:

Test Case 1: Successful login

- **Objective:** To verify successful user login given valid inputs
- **Preconditions:**
 - *User is on the login page*
- **Test Steps:**
 - *Enter a registered email in the “email” field*
 - *Enter the correct password associated with that email in the “password” field*
 - *Submit the login form*
- **Expected Result:**
 - *The system authenticates that user and redirects them to whatever page they were previously on*

Test Case 2: Unsuccessful login with incorrect password

- **Objective:** To verify the system prevents login given an incorrect password
- **Preconditions:**
 - *User is on the login page*
- **Test Steps:**
 - *Enter a registered email in the “email” field*
 - *Enter an incorrect password in the “password” field*
 - *Submit the login form*
- **Expected Result:**
 - *The system displays an error message saying that the entered email or password is incorrect*
 - *User stays on the login page*

Test Case 3: Unsuccessful login with incorrect email

- **Objective:** To verify the system prevents login given an unregistered email
- **Preconditions:**
 - *User is on the login page*
- **Test Steps:**
 - *Enter a unregistered email in the “email” field*
 - *Enter any password in the “password” field*
 - *Submit the login form*
- **Expected Result:**
 - *The system displays an error message saying that entered email or password is incorrect*
 - *User stays on the login page*

Test Case 4: login with empty fields

- **Objective:** To verify that the system prompts the user when email or password fields are left empty
- **Preconditions:**
 - *User is on the login page*
- **Test Steps:**
 - *Leave the “email” and or “password” fields empty*
 - *Submit the login form*
- **Expected Result:**
 - *An error message is displayed explaining that both fields are required*
 - *User stays on the login page*

Requirement 3:

Title: *Flight Search Functionality*

Description:

The system shall allow the users to search for flights based on origin, destination, date, number of passengers, and select between one-way and return trips.

Acceptance Criteria/Test Cases:

Test Case 1: Successful one-way flight search

- **Objective:** To verify that users can search for one-way flights successfully
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select the "One-way" trip option*
 - *Enter a valid origin airport or city in the "From" field.*
 - *Enter a valid destination airport or city in the "To" field*
 - *Select a valid departure date in the "Departure" field*
 - *Ensure the "Return" field is disabled or left empty*
 - *Enter a valid number of passengers in the "Passengers" field*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system proceeds to the search results page which displays a list of available one-way flights that match the search criteria*

Test Case 2: Successful return trip flight search

- **Objective:** To verify that users can search for return trips successfully
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select the "Return Trip" option*
 - *Enter a valid origin airport or city in the "From" field.*
 - *Enter a valid destination airport or city in the "To" field*
 - *Select a valid departure date in the "Departure" field*
 - *Select a valid return date in the "Return" field that is after the departure date*
 - *Enter a valid number of passengers in the "Passengers" field*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system displays a list of available flights for both departure and return that match the search criteria*

Test Case 3: Switching between one-way and return trip options

- **Objective:** To verify that the system updates the form fields appropriately when switching between trip types.
- **Preconditions:**
 - *User is on the flight search page*

- **Test Steps:**
 - *Select the "Return Trip" option*
 - *Observe that the "Return" field is enabled*
 - *Switch to the "One-way" trip option*
 - *Observe that the "Return" field is disabled or cleared*

- **Expected Result:**
 - *The "Return" field is enabled when "Return Trip" is selected and disabled or cleared when "One-way" is selected*

Test Case 4: Flight search with invalid dates

- **Objective:** To verify that the system handles invalid date inputs appropriately
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select either "Return Trip" or "One-way" options*
 - *Enter a valid origin airport or city in the "From" field.*
 - *Enter a valid destination airport or city in the "To" field*
 - *Select a departure date that is in the past*
 - *Enter a valid number of passengers.*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system displays an error message indicating that the departure date cannot be in the past*

Test Case 5: Return date before departure date for "Return Trip" flights

- **Objective:** To verify that the system handles cases where the return date is before the departure date.
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select the "Return Trip" option*
 - *Enter a valid origin airport or city in the "From" field.*
 - *Enter a valid destination airport or city in the "To" field*
 - *Select a departure date in the "Departure" field*
 - *Select a return date that is before the departure date in the "Return" field*
 - *Enter a valid number of passengers*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system displays an error message indicating that the return date cannot be before the departure date*

Test Case 6: Flight search with invalid number of passengers

- **Objective:** To verify that the system handles invalid passenger inputs
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select either "Return Trip" or "One-way" options*
 - *Enter a valid origin airport or city in the "From" field.*
 - *Enter a valid destination airport or city in the "To" field*
 - *Select a departure date in the "Departure" field*
 - *If "Return Trip" is selected, select a valid return date from the "Return" field*
 - *Enter "0" or a negative number in the "Passengers" field*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system displays a validation message indicating that the number of passengers must be at least one*

Test Case 7: Flight search with non-existent route

- **Objective:** To verify that the system handles searches for routes that don't exist
- **Preconditions:**
 - *User is on the flight search page*
- **Test Steps:**
 - *Select either "Return Trip" or "One-way" options*
 - *Enter a valid origin airport or city in the "From" field*
 - *Enter a destination that is not serviced in the "To" field*
 - *Select a departure date in the "Departure" field*
 - *If "Return Trip" is selected, select a valid return date from the "Return" field*
 - *Enter a valid number of passengers*
 - *Click the "Search" button*
- **Expected Result:**
 - *The system displays a message indicating that no flights are available for the selected route*

Requirement 4:

Title: *Flight Selection*

Description:

After performing a flight search, the system shall allow users to select a flight (or flights) from the list of available options for booking

Acceptance Criteria/Test Cases:

Test Case 1: Successful flight selection

- **Objective:** To verify that a user can successfully select a flight from the search results
- **Preconditions:**
 - *User has performed a flight search and is on the search results page*
 - *A list of available flights is displayed*
- **Test Steps:**
 - *Review the list of available flights*
 - *Click the "Check" button on the flight you want to view*
- **Expected Result:**
 - *The system proceeds to the flight details page for that specific flight*

Test Case 2: Selecting multiple flights for return trip

- **Objective:** To verify that a user can select outbound and return flights for a return trip
- **Preconditions:**
 - *User has searched for a return trip*
 - *A list of outbound flights are displayed first*
- **Test Steps:**
 - *Click the "Check" button on the flight you want to view, this should take you to the flight details page for that flight*
 - *Click the "Choose seats" button, this will take you to a seat selection page*
 - *Click "Confirm", This will take you to a page that displays a list of return flights*
 - *Repeat the process for the return flight*
- **Expected Result:**
 - *Once both the outbound and return flights are confirmed the system will redirect the user to the booking summary page*

Requirement 5:

Title: Seat Selection

Description:

After selecting flights, the system shall allow users to select their preferred seats from an interactive seat map before proceeding to payment

Acceptance Criteria/Test Cases:

Test Case 1: Successful seat selection during booking

- **Objective:** To verify that users can select seats for their flights before making payment
- **Preconditions:**
 - *User has selected a flight*
 - *User is on the seat selection page*
- **Test Steps:**
 - *View the interactive seat map displayed for the selected flight*
 - *Click on available seats to select them for each passenger*

- Click the "Confirm" button
- **Expected Result:**
 - The selected seats are reserved and added to the booking summary
 - The system proceeds to the booking summary page with selected flights' information displayed

Test Case 2: Attempting to select unavailable or occupied seats

- **Objective:** To verify that the system prevents users from selecting seats that are already occupied or unavailable by greying them out and making them non-clickable
- **Preconditions:**
 - User has selected a flight or flights (outbound and return if applicable)
 - User is on the seat selection page
 - Some seats on this flight have already been reserved.
- **Test Steps:**
 - Identify seats that are greyed out
 - Note that available seats are displayed normally and are selectable
 - Try to select a seat that is displayed as greyed out
- **Expected Result:**
 - Greyed-out seats are non-clickable and do not respond to any user interaction

Requirement 6:

Title: Booking and Payment Processing

Description:

The system shall allow users to book selected flights and make payments using a credit card

Acceptance Criteria/Test Cases:

Test Case 1: Successful booking and payment using a credit card

- **Objective:** To verify that a user can successfully book a flight and make a payment using a credit card
- **Preconditions:**
 - User has selected a flight and their seats and is on the booking summary page
 - User is logged into their account
- **Test Steps:**
 - Review the booking summary details to ensure the correct flight is selected
 - Click on the "Payment" button
 - Enter valid credit card details
 - Click the "Pay Now" button
- **Expected Result:**
 - The system processes the payment successfully
 - User is redirected to the booking confirmation page

Test Case 2: Unsuccessful payment due to invalid credit card details

- **Objective:** To verify that the system handles payment failures when invalid credit card details are provided
- **Preconditions:**
 - *User is on the payment page after selecting a flight and clicking the "Payment" button on the booking summary details page*
- **Test Steps:**
 - *Enter invalid credit card details*
 - *Enter the billing address*
 - *Click the "Pay Now" button*
- **Expected Result:**
 - *The system displays an error message indicating that the payment was unsuccessful due to invalid card details*
 - *The user remains on the payment page*

Requirement 7:

Title: *Booking confirmation*

Description:

Upon successful booking and payment, the system shall provide booking confirmation details and reference number to the user. The system shall also send a confirmation email to the user's registered email address

Acceptance Criteria/Test Cases:

Test Case 1: Successful booking confirmation display

- **Objective:** To verify that the system displays a booking confirmation page after successful payment and sends the user a confirmation email.
- **Preconditions:**
 - *User has successfully completed the payment process*
- **Test Steps:**
 - *Complete the booking and payment process successfully*
 - *Observe the page displayed after payment completion*
 - *Check registered email address*
- **Expected Result:**
 - *The system displays a booking confirmation page that includes:*
 - *Booking reference number*
 - *Payment summary*
 - *A message confirming that a confirmation email has been sent*

Requirement 8:

Title: *Manage Bookings*

Description:

The system shall allow users to view, modify, or cancel existing bookings

Acceptance Criteria/Test Cases:

Test Case 1: Viewing existing bookings

- **Objective:** To verify that a user can view their existing bookings
- **Preconditions:**
 - *User is logged into their account*
 - *User has at least one existing booking*
- **Test Steps:**
 - *Navigate to the "My Bookings" section.*
 - *Observe the list of bookings displayed*
- **Expected Result:**
 - *The system displays a list of the user's existing bookings*
 - *The bookings should display detailed information such as flight details and dates*

Test Case 2: Modifying a booking

- **Objective:** To verify that a user can modify an existing booking
- **Preconditions:**
 - *User is logged into their account*
 - *User has at least one existing booking*
- **Test Steps:**
 - *Navigate to the "My Bookings" section.*
 - *Select the booking to modify*
 - *Click on the "modify" option*
 - *Update the desired details*
 - *Proceed to the payment page if additional charges apply*
 - *Complete the payment process (if applicable)*
 - *Confirm the changes*
- **Expected Result:**
 - *The system updates the booking with the new details*
 - *A confirmation message is displayed, and a confirmation email is sent with the updated booking information*
 - *Any additional charges are processed successfully*

Test Case 3: Cancelling a booking

- **Objective:** To verify that a user can cancel an existing booking
- **Preconditions:**
 - *User is logged into their account*

- *User has at least one existing booking*
- **Test Steps:**
 - *Navigate to the "My Bookings" section.*
 - *Select the booking to cancel*
 - *Click on the "Modify" option*
 - *Click the "Cancel" option*
 - *Confirm the cancelation*
- **Expected Result:**
 - *A cancellation confirmation message is displayed*
 - *The booking status is updated to "Cancelled" in the user's account*
 - *A confirmation email is sent to the user*

Requirement 9:

Title: *Boarding Pass Access*

Description:

The system shall allow users to view their boarding passes from the "Manage Bookings" section.

Acceptance Criteria/Test Cases:

Test Case 1: Viewing the Boarding Pass Online

- **Objective:** To verify that a user can view their boarding pass
- **Preconditions:**
 - *User is logged into their account*
 - *User has an upcoming flight booking*
- **Test Steps:**
 - *Navigate to the "My Bookings" section*
 - *Click on the "View Boarding Pass" option*
- **Expected Result:**
 - *The system displays the boarding pass on-screen, including:*
 - *Passenger name*
 - *Flight details (flight number, departure date and time, gate, seat)*
 - *Barcode for scanning at the airport*