Airline Reservation - Registration Page

Source Code:

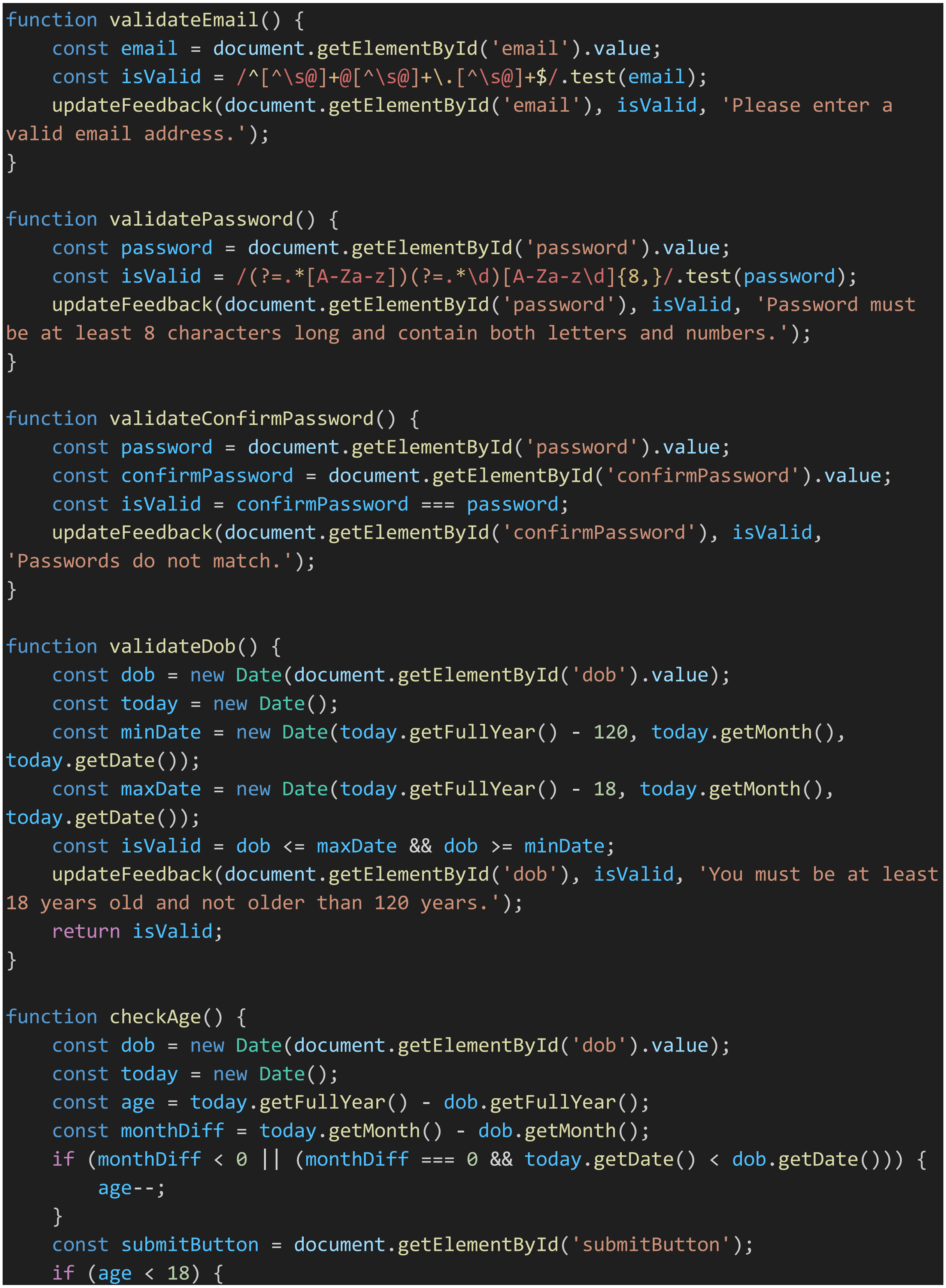
registration.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Airline Reservation - Registration</title>
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link
href="https://cdn.jsdelivr.net/npm/tailwindcss@3.3.0/dist/tailwind.min.css"
rel="stylesheet">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.5.2/css/all.min.css">
    <style>
        .wrapper {
            min-height: 100vh;
            display: flex;
            flex-direction: column;
        }
        .navbar {
            background-color: #c8102e;
        }
        .footer {
            background-color: #222;
            color: #fff;
        }
        .form-control {
            border: 1px solid #ced4da;
            border-radius: 0.375rem;
            padding: 0.5rem;
            transition: border-color 0.3s ease;
        }
        .form-control.error {
            border-color: red;
            background-color: #f8d7da;
        }
        .form-control.valid {
            border-color: green;
            background-color: #d4edda;
        }
        .form-feedback {
            font-size: 0.875em;
            margin-top: 0.25rem;
        }
        .form-error {
            color: red;
        }
        .form-valid {
            color: green;
        }
    </style>
</head>
<body class="bg-gray-100">
    <div class="wrapper">
        <nav class="navbar navbar-expand-lg navbar-dark bg-red-700">
            <div class="container">
                <a class="navbar-brand text-white" href="main.html">
                    <i class="fa-solid fa-plane-departure fa-2x"></i> Airline
Reservations
                </a>
                <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-
expanded="false" aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>
                <div class="collapse navbar-collapse" id="navbarNav">
                    <ul class="navbar-nav ms-auto">
                        <li class="nav-item">
                            <a class="nav-link active text-white"
href="main.html">Home <i class="fa-solid fa-house"></i></a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link text-white"
href="registration.html">Registration <i class="fa-solid fa-user-plus"></i></a>
                        </li>
                    </ul>
                </div>
            </div>
        </nav>

        <main class="container my-5">
            <h3 class="text-center mb-4 text-2xl font-semibold">Register</h3>
            <div class="max-w-md mx-auto bg-white p-8 border border-gray-300 rounded-lg shadow-sm">
                <form id="registrationForm">
                    <div class="mb-4">
                        <label for="name" class="block text-sm font-medium text-gray-700">Name</label>
                        <input type="text" class="form-control mt-1 block w-full" id="name" placeholder="Enter your name">
                        <div id="nameFeedback" class="form-feedback mt-1"></div>
                    </div>
                    <div class="mb-4">
                        <label for="email" class="block text-sm font-medium text-gray-700">Email address</label>
                        <input type="email" class="form-control mt-1 block w-full" id="email" placeholder="Enter your email">
                        <div id="emailFeedback" class="form-feedback mt-1"></div>
                    </div>
                    <div class="mb-4">
                        <label for="password" class="block text-sm font-medium text-gray-700">Password</label>
                        <input type="password" class="form-control mt-1 block w-full" id="password" placeholder="Enter your password">
                        <div id="passwordFeedback" class="form-feedback mt-1"></div>
                    </div>
                    <div class="mb-4">
                        <label for="confirmPassword" class="block text-sm font-medium text-gray-700">Confirm Password</label>
                        <input type="password" class="form-control mt-1 block w-full" id="confirmPassword" placeholder="Confirm your password">
                        <div id="confirmPasswordFeedback" class="form-feedback mt-1"></div>
                    </div>
                    <div class="mb-4">
                        <label for="dob" class="block text-sm font-medium text-gray-700">Date of Birth</label>
                        <input type="date" class="form-control mt-1 block w-full" id="dob" placeholder="YYYY-MM-DD">
                        <div id="dobFeedback" class="form-feedback mt-1"></div>
                    </div>
                    <button type="submit" id="submitButton" class="btn btn-primary bg-indigo-600 hover:bg-indigo-700 text-white py-2 px-4 rounded-md shadow-sm">Register</button>
                </form>
            </div>
```

```html
        </main>

        <footer class="footer py-4 text-center bg-gray-900 text-white">
            <div class="container">
                <p>&copy; 2024 Airline Reservations. All rights reserved.</p>
            </div>
        </footer>
    </div>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
    <script src="form-validation.js"></script>
</body>
</html>
```

form-validation.js

```javascript
function updateFeedback(input, isValid, message) {
    const feedbackElement = document.getElementById(`${input.id}Feedback`);
    if (isValid) {
        feedbackElement.innerHTML = '<i class="fa-solid fa-check form-valid"></i>';
        input.classList.remove('error');
        input.classList.add('valid');
    } else {
        feedbackElement.innerHTML = `<i class="fa-solid fa-xmark form-error"></i> ${message}`;
        input.classList.remove('valid');
        input.classList.add('error');
    }
}

function validateName() {
    const name = document.getElementById('name').value;
    const isValid = /^[A-Za-z\s]{3,}$/.test(name);
    updateFeedback(document.getElementById('name'), isValid, 'Name must be at least 3 characters long and contain only letters and spaces.');
}
```



```
function validateEmail() {
    const email = document.getElementById('email').value;
    const isValid = /^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email);
    updateFeedback(document.getElementById('email'), isValid, 'Please enter a
valid email address.');
}

function validatePassword() {
    const password = document.getElementById('password').value;
    const isValid = /(?=.*[A-Za-z])(?=.*\d)[A-Za-z\d]{8,}/.test(password);
    updateFeedback(document.getElementById('password'), isValid, 'Password must
be at least 8 characters long and contain both letters and numbers.');
}

function validateConfirmPassword() {
    const password = document.getElementById('password').value;
    const confirmPassword = document.getElementById('confirmPassword').value;
    const isValid = confirmPassword === password;
    updateFeedback(document.getElementById('confirmPassword'), isValid,
'Passwords do not match.');
}

function validateDob() {
    const dob = new Date(document.getElementById('dob').value);
    const today = new Date();
    const minDate = new Date(today.getFullYear() - 120, today.getMonth(),
today.getDate());
    const maxDate = new Date(today.getFullYear() - 18, today.getMonth(),
today.getDate());
    const isValid = dob <= maxDate && dob >= minDate;
    updateFeedback(document.getElementById('dob'), isValid, 'You must be at least
18 years old and not older than 120 years.');
    return isValid;
}

function checkAge() {
    const dob = new Date(document.getElementById('dob').value);
    const today = new Date();
    const age = today.getFullYear() - dob.getFullYear();
    const monthDiff = today.getMonth() - dob.getMonth();
    if (monthDiff < 0 || (monthDiff === 0 && today.getDate() < dob.getDate())) {
        age--;
    }
    const submitButton = document.getElementById('submitButton');
    if (age < 18) {
        submitButton.disabled = true;
        updateFeedback(document.getElementById('dob'), false, 'You must be at least 18 years old.');
    } else {
        submitButton.disabled = false;
        updateFeedback(document.getElementById('dob'), true, '');
    }
}

document.getElementById('name').addEventListener('input', validateName);
document.getElementById('email').addEventListener('input', validateEmail);
document.getElementById('password').addEventListener('input', validatePassword);
document.getElementById('confirmPassword').addEventListener('input', validateConfirmPassword);
document.getElementById('dob').addEventListener('input', function() {
    validateDob();
    checkAge();
});

document.getElementByI
```

Overview

The page includes user registration functionality with built-in client-side validation to ensure the correctness of user inputs. This documentation will cover the HTML structure, CSS styles, JavaScript functionality, and design rationale.

---

2. HTML Structure

The HTML document is structured to create a user registration page with a responsive design. It includes:

- **Header Section**:
  - **Metadata**: Defines the character encoding, viewport settings, and page title.
  - **CSS Links**: Imports Bootstrap for layout and styling, TailwindCSS for utility classes, and Font Awesome for icons.
  - **Inline Styles**: Contains custom CSS for form controls and feedback.
- **Navigation Bar**:

    - A responsive navigation bar using Bootstrap classes. It includes links to the homepage and registration page, and is styled with a background color of red (`#c8102e`).
- **Main Content**:
    - **Registration Form**:
        - Fields for `Name`, `Email`, `Password`, `Confirm Password`, and `Date of Birth`.
        - Each field includes dynamic feedback elements that provide validation messages.
    - **Submit Button**: A button to submit the form, initially enabled but may be disabled based on age validation.
- **Footer**:
    - A footer with copyright information and simple styling.

---

3. CSS Styles

Custom CSS is used to enhance the appearance and functionality of the form:

- **`.wrapper`**:
    - Ensures the page content spans the full viewport height and uses a flexbox layout to vertically center the content.
- **`.navbar`**:
    - Sets the background color for the navigation bar to red (`#c8102e`).
- **`.footer`**:
    - Applies dark background color and white text to the footer for contrast.
- **`.form-control`**:
    - Defines base styles for input fields, including border, padding, and transition effects.
    - Additional styles are applied for error and valid states to provide visual feedback.
- **`.form-feedback`**:
    - Adjusts font size and margin for feedback messages below form inputs.

---

4. JavaScript Functionality

JavaScript is used to validate user inputs and update the user interface based on validation results. The key functions are:

- **`updateFeedback(input, isValid, message)`**:
    - Updates the feedback message and input border color based on whether the input is valid.
    - **Parameters**:
        - `input`: The HTML input element.

        - `isValid`: Boolean indicating if the input is valid.
        - `message`: The feedback message to display.
- **`validateName()`**:
    - Validates that the name is at least 3 characters long and contains only letters and spaces.
- **`validateEmail()`**:
    - Ensures the email format is correct using a regular expression.
- **`validatePassword()`**:
    - Validates that the password is at least 8 characters long and contains both letters and numbers.
- **`validateConfirmPassword()`**:
    - Confirms that the password and confirm password fields match.
- **`validateDob()`**:
    - Checks if the date of birth is within a valid range (18-120 years old).
- **`checkAge()`**:
    - Calculates the user's age and disables the submit button if the user is under 18. Provides appropriate feedback.

Event Listeners:

- **Input Events**:
    - Attach validation functions to input fields to provide real-time feedback.
    - Fields include `name`, `email`, `password`, `confirmPassword`, and `dob`.

---

5. Design

1. User Experience:

- Real-time validation improves user interaction by providing instant feedback, allowing users to correct errors immediately.

2. Form Validation:

- Client-side validation reduces server load and ensures that users input valid data before form submission. This improves data integrity and user satisfaction.

3. Responsiveness:

- The page design is responsive, adapting to different screen sizes using Bootstrap's grid system and TailwindCSS utilities.

4. Styling:

- The use of Bootstrap and TailwindCSS provides a flexible styling framework that ensures a consistent look and feel while allowing for easy customizations.

5. Accessibility:

- Clear visual feedback for errors and successful validations helps users with visual impairments understand form status.

6. Maintainability:

- Modular JavaScript functions for validation and separate CSS files for styling ensure that the code is easy to maintain and update.

---

6. Conclusion

The registration page for the Airline Reservation system is designed to be user-friendly, secure, and responsive. It ensures data validity through client-side validation while maintaining an accessible and aesthetically pleasing interface. The combination of Bootstrap, TailwindCSS, and custom JavaScript provides a comprehensive solution for user registration.