

User Registration Form Documentation

File Name:

`signup.html`

Purpose:

The purpose of the `signup.html` page is to collect essential user information for registration on a new website. The form includes fields for first name, last name, email, date of birth (DOB), country, password, and confirm password. This document details the validation logic implemented to ensure data integrity and user-friendly feedback.

Form Structure:

The form includes the following fields:

1. **First Name** - Input accepts only alphabetic characters and spaces, with a minimum of 3 characters.
2. **Last Name** - Input accepts only alphabetic characters and spaces, with a minimum of 3 characters.
3. **Email** - Input requires a valid email format (e.g., "user@example.com").
4. **Date of Birth (DOB)** - Input accepts the "YYYY-MM-DD" format and ensures the user is at least 18 years old.
5. **Country** - Dropdown or input field for the user's country.
6. **Password** - Input must be at least 8 characters long and contain both letters and numbers.
7. **Confirm Password** - Input must match the "Password" field.

Validation Logic:

1. First Name and Last Name Validation:

- **Function:** `validateFirstName()` and `validateLastName()`
- **Validation Criteria:**
 - Accepts only alphabetic characters and spaces.
 - Requires a minimum of 3 characters.
- **Feedback:**
 - Displays a green checkmark for valid input.
 - Displays a red "x" and an error message for invalid input.
 - Border color is dynamically updated (green for valid, red for invalid).

2. Email Validation:

- **Function:** `validateEmail()`
- **Validation Criteria:**
 - Requires a valid email format.

- **Feedback:**
 - Displays a green checkmark for valid input.
 - Displays a red "x" and an error message for invalid input.
 - Border color is dynamically updated.

3. Date of Birth (DOB) Validation:

- **Function:** `validateAge()`
- **Validation Criteria:**
 - Accepts date in "YYYY-MM-DD" format.
 - Checks if the user is at least 18 years old.
- **Feedback:**
 - Displays a green checkmark for valid input.
 - Displays a red "x" and an error message for invalid input.
 - Border color is dynamically updated.
 - Disables the submit button if the user is under 18.

4. Password and Confirm Password Validation:

- **Functions:** `validatePassword()` and `validateConfirmPassword()`
- **Validation Criteria:**
 - Password must be at least 8 characters long and contain both letters and numbers.
 - Confirm Password must match the Password field.
- **Feedback:**
 - Displays a green checkmark for valid input.
 - Displays a red "x" and an error message for invalid input.
 - Border color is dynamically updated.
 - Continuously checks if the "Password" and "Confirm Password" fields match.

Implementation Details:

JavaScript Validation Functions:

- Each input field has an associated validation function that checks the input value against the specified criteria.
- Event listeners are attached to each form element to trigger validation as the user interacts with the form (e.g., `input`, `blur`, `change` events).
- The form submission is prevented if any field fails validation, ensuring that only valid data is submitted.

Real-Time Feedback:

- As users type in each field, validation is performed in real-time.
- Icons (green checkmark or red "x") are displayed next to each input field to indicate validity.
- Border colors of the input fields are dynamically updated to red for invalid inputs and green for valid inputs.

- Error messages are displayed next to fields that fail validation, guiding the user to correct their input.

Design Decisions:

- **User-Friendly Experience:** The real-time feedback approach ensures users can correct errors as they fill out the form, reducing frustration and improving the overall user experience.
- **Visual Indicators:** The use of color-coded borders and icons provides clear visual cues, making it easy for users to understand the validation status of each field.
- **Security:** Enforcing strong password criteria and matching passwords increases the security of user accounts.

Form Submission:

- The form will only submit when all fields pass validation, ensuring that all collected data is correct and meets the specified criteria.