Problem Statement 2: Form Validator Application (Client-Side Validation Project)

**Steps and screenshot**

**HTML**

1. Created a form inside a centered container.
2. Added input fields for:
   * Full Name
   * Email
   * Password
   * Confirm Password
3. Included a password rule checklist using <ul> with five <li> items.
4. Added a password strength label below the password field.
5. Included a “Show/Hide Password” toggle icon.
6. Added a submit button and a message area to display success or error feedback.

**CSS**

1. Styled the container with padding, rounded corners, and box shadow.
2. Applied green/red borders to inputs based on validation status.
3. Styled password rules with .valid-rule (green) and .invalid-rule (red).
4. Highlighted invalid rows using .form-group.error.
5. Styled the submit button with hover effects.
6. Made the layout responsive for smaller screens.

**JavaScript**

1. Selected all form elements using getElementById.
2. Defined password validation rules using regular expressions:
   * Minimum 8 characters
   * One uppercase letter
   * One lowercase letter
   * One number
   * One special character
3. Added real-time validation for each field using input event listeners.
4. Applied .valid and .invalid classes to show green/red borders.
5. Toggled password visibility using the eye icon.
6. Updated password rule checklist dynamically as the user types.
7. Displayed password strength (Weak, Medium, Strong) based on matched rules.
8. On form submission:
   * Prevented default behavior
   * Validated all fields
   * Highlighted invalid rows
   * Displayed success message if all fields are valid
   * Displayed error message if any field is invalid

**Html Code**

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Form Validator</title>

  <link rel="stylesheet" href="style.css" />

  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css" />

</head>

<body>

  <div class="container">

    <h2>Form Validator</h2>

    <form id="form">

      <div class="form-group">

        <label for="fullName">Full Name</label>

        <input type="text" id="fullName" />

      </div>

      <div class="form-group">

        <label for="email">Email</label>

        <input type="email" id="email" />

      </div>

<div class="form-group">

  <label for="password">Password</label>

  <div class="password-wrapper">

    <input type="password" id="password" />

    <span id="toggleIcon" class="eye-icon"><i class="fas fa-eye"></i></span>

  </div>

  <ul id="passwordRules">

    <li id="ruleLength">- At least 8 characters</li>

    <li id="ruleUpper">- One uppercase letter</li>

    <li id="ruleLower">- One lowercase letter</li>

    <li id="ruleNumber">- One number</li>

    <li id="ruleSpecial">- One special character</li>

  </ul>

  <div id="strengthLabel"></div>

</div>

      <div id="strengthLabel"></div>

      <div class="form-group">

        <label for="confirmPassword">Confirm Password</label>

        <input type="password" id="confirmPassword" />

      </div>

      <button type="submit">Submit</button>

      <p id="formMessage"></p>

    </form>

  </div>

  <script src="script.js"></script>

</body>

</html>

**Css Code**

body {

  font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;

  background-color: #f3f2b7;

  padding: 30px;

}

.container {

  max-width: 400px;

  width: 100%;

  margin: auto;

  background: #fff;

  padding: 20px;

  border-radius: 10px;

  box-sizing: border-box;

  box-shadow: 0 0 10px rgba(0,0,0,0.1);

}

h2 {

  text-align: center;

  margin-bottom: 20px;

}

.form-group {

  margin-bottom: 20px;

  width: 100%;

  display: flex;

  flex-direction: column;

  box-sizing: border-box;

}

input {

  width: 100%;

  padding: 10px;

  border: 2px solid #ccc;

  border-radius: 5px;

  box-sizing: border-box;

}

input.valid {

  border-color: green;

}

input.invalid {

  border-color: red;

}

.password-wrapper {

  position: relative;

  display: flex;

  align-items: center;

}

.password-wrapper input {

  flex: 1;

  padding-right: 40px;

}

.eye-icon {

  position: absolute;

  right: 10px;

  top: 50%;

  transform: translateY(-50%);

  cursor: pointer;

  color: #555;

}

#passwordRules {

  list-style: none;

  padding-left: 0;

  margin-top: 10px;

  font-size: 0.9em;

}

#passwordRules li {

  margin-bottom: 5px;

}

.valid-rule {

  color: green;

  font-weight: bold;

}

.invalid-rule {

  color: red;

  font-weight: normal;

}

#strengthLabel {

  margin-top: 8px;

  font-weight: bold;

  font-size: 0.9em;

}

#formMessage {

  text-align: center;

  font-weight: bold;

  margin-top: 15px;

}

.form-group.error {

  border: 2px solid red;

  border-radius: 5px;

  padding: 10px;

  box-sizing: border-box;

  width: 100%;

}

button[type="submit"] {

  padding: 10px 15px;

  background-color: #007bff;

  color: white;

  border: none;

  border-radius: 5px;

  cursor: pointer;

  width: 100%;

}

button[type="submit"]:hover {

  background-color: #0056b3;

  color: white;

  box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

}

@media (max-width: 480px) {

  body {

    padding: 10px;

  }

  .container {

    padding: 15px;

  }

  input,

  button[type="submit"] {

    font-size: 1em;

  }

  #passwordRules {

    font-size: 0.85em;

  }

}

**Javascript Code**

// Get form elements

const form = document.getElementById("form");

const fullName = document.getElementById("fullName");

const email = document.getElementById("email");

const password = document.getElementById("password");

const confirmPassword = document.getElementById("confirmPassword");

const formMessage = document.getElementById("formMessage");

const toggleIcon = document.getElementById("toggleIcon");

const strengthLabel = document.getElementById("strengthLabel");

// Password rule elements

const rules = {

  ruleLength: val => val.length >= 8,

  ruleUpper: val => /[A-Z]/.test(val),

  ruleLower: val => /[a-z]/.test(val),

  ruleNumber: val => /\d/.test(val),

  ruleSpecial: val => /[@$!%\*?&]/.test(val)

};

// Toggle password visibility

toggleIcon.addEventListener("click", () => {

  const isHidden = password.type === "password";

  password.type = isHidden ? "text" : "password";

  toggleIcon.innerHTML = isHidden

    ? '<i class="fas fa-eye-slash"></i>'

    : '<i class="fas fa-eye"></i>';

});

// Utility to add validation classes

function addValidation(input, isValid) {

  input.classList.remove("valid", "invalid");

  input.classList.add(isValid ? "valid" : "invalid");

}

// Full Name validation

fullName.addEventListener("input", () => {

  const isValid = /^[A-Za-z]{3,}$/.test(fullName.value);

  addValidation(fullName, isValid);

});

// Email validation

email.addEventListener("input", () => {

  const isValid = /^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email.value);

  addValidation(email, isValid);

});

// Password validation and rule highlighting

password.addEventListener("input", () => {

  const val = password.value;

  let matchedRules = 0;

  for (let rule in rules) {

    const ruleElement = document.getElementById(rule);

    if (rules[rule](val)) {

      ruleElement.classList.add("valid-rule");

      ruleElement.classList.remove("invalid-rule");

      matchedRules++;

    } else {

      ruleElement.classList.add("invalid-rule");

      ruleElement.classList.remove("valid-rule");

    }

  }

  // Password strength indicator

  let strength = "Weak";

  if (val.length >= 8) strength = "Medium";

  if (matchedRules === 5) strength = "Strong";

  strengthLabel.innerText = `Strength: ${strength}`;

  strengthLabel.style.color =

    strength === "Strong" ? "green" : strength === "Medium" ? "orange" : "red";

  addValidation(password, matchedRules === 5);

});

// Confirm Password validation

confirmPassword.addEventListener("input", () => {

  const isValid = confirmPassword.value === password.value;

  addValidation(confirmPassword, isValid);

});

// Final form submission

form.addEventListener("submit", (e) => {

  e.preventDefault();

  // Clear previous error highlights

  document.querySelectorAll(".form-group").forEach(group => {

    group.classList.remove("error");

  });

  // Validation checks

  const isNameValid = /^[A-Za-z]{3,}$/.test(fullName.value);

  const isEmailValid = /^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email.value);

  const isPasswordValid = Object.values(rules).every(fn => fn(password.value));

  const isConfirmValid = confirmPassword.value === password.value;

  // Highlight errors

  if (!isNameValid) fullName.parentElement.classList.add("error");

  if (!isEmailValid) email.parentElement.classList.add("error");

  if (!isPasswordValid) password.parentElement.classList.add("error");

  if (!isConfirmValid) confirmPassword.parentElement.classList.add("error");

  // Final message

  if (isNameValid && isEmailValid && isPasswordValid && isConfirmValid) {

    formMessage.style.color = "green";

    formMessage.innerText = "Form Submitted Successfully!";

  } else {

    formMessage.style.color = "red";

    formMessage.innerText = "Please fix the highlighted fields.";

  }

});

**Website Screenshots**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Password strength indicator**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

**Shows error when any field doesn’t match the requirements**

**A screenshot of a computer

AI-generated content may be incorrect.**