**USER INTERFACE AND DESIGN**

**EXPERIMENT – 9 NAME: P.SANDHIYA (230701282)**

**1. Aim**

To design input forms that validate data (email and phone number) and display error messages using HTML/CSS and JavaScript with Validator.js.

**2. Requirements**

* Visual Studio Code (VS Code)
* Internet connection (for loading Validator.js)
* Browser (Chrome, Edge, Firefox, etc.)

**3. Tools and Technologies Used**

* **HTML5** – Structure of the form
* **CSS3** – Styling the form
* **JavaScript** – Handling validation
* **Validator.js** – External library for email and phone number validation

**4. Procedure**

**Step 1: Creating HTML Form**

* Designed a form with email and phone number fields.
* Added placeholders and "required" attribute for basic validation.

**Step 2: Styling with CSS**

* Applied styles for form alignment, input fields, button designs, and error messages.

**Step 3: Adding JavaScript Validation**

* Used Validator.js to:
  + Check if email is valid.
  + Check if phone number is valid.
* If invalid input, show error messages below each field.

**Step 4: Testing**

* Tested the form with correct and incorrect data inputs to ensure proper validation and error display.

**5. Code Snippets**

**index.html**

!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Form Validation</title>

</head>

<body>

<!-- Form starts here -->

<form>

  <label for="email">Email:</label>

  <input type="email" id="email" name="email" required>

  <br><br>

  <label for="phone">Phone Number:</label>

  <input type="text" id="phone" name="phone" required>

  <br><br>

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

</form>

<!-- Form ends here -->

</body>

</html>

**Style.css**

body {

    font-family: Arial, sans-serif;

    background-color: #f4f4f4;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    margin: 0;

  }

  .container {

    background-color: white;

    padding: 20px;

    border-radius: 5px;

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

  }

  form {

    display: flex;

    flex-direction: column;

  }

  label {

    margin-bottom: 5px;

  }

  input {

    margin-bottom: 10px;

    padding: 10px;

    border: 1px solid #ccc;

    border-radius: 3px;

  }

  button {

    padding: 10px;

    background-color: #28a745;

    color: white;

    border: none;

    border-radius: 3px;

    cursor: pointer;

  }

  button:hover {

    background-color: #218838;

  }

  .error {

    color: red;

    font-size: 0.875em;

  }

**Script.js**

document.getElementById('myForm').addEventListener('submit', function (e) {

    e.preventDefault(); // normal submit ah stop pannuvom

    // Values eduthu vaanguradhu

    let email = document.getElementById('email').value;

    let phone = document.getElementById('phone').value;

    let emailError = document.getElementById('emailError');

    let phoneError = document.getElementById('phoneError');

    // Munadi errors clear pannuvom

    emailError.textContent = '';

    phoneError.textContent = '';

    // Email check

    if (!validator.isEmail(email)) {

      emailError.textContent = 'Please enter a valid email address.';

    }

    // Phone check

    if (!validator.isMobilePhone(phone, 'any')) {

      phoneError.textContent = 'Please enter a valid phone number.';

    }

    // Both sari na, console-la print

    if (validator.isEmail(email) && validator.isMobilePhone(phone, 'any')) {

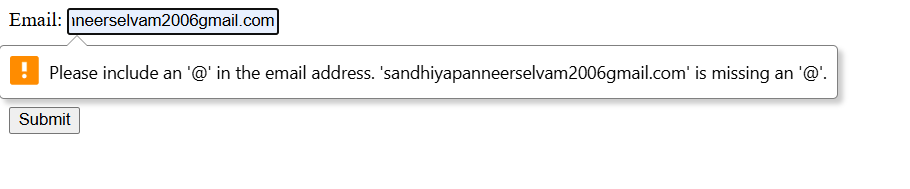
      console.log('Email:', email);

      console.log('Phone:', phone);

    }

  });

**6. Output**

* Form displayed with fields for Email and Phone Number.
* Proper error messages shown for invalid inputs.
* Successful logging of valid inputs in console.
* The below is the displayed error message.

**7. Result**

Thus, an input form with email and phone number validation was successfully designed and error messages were displayed using HTML, CSS, JavaScript, and Validator.js