**Guiado Formulario Responsivo con Flexbox**

 **Validación nativa HTML5**: Uso de atributos como required, pattern y type para validación.

 **Indicadores visuales CSS**: Uso de pseudo-clases CSS para mostrar estados de validación.

**1. Validación nativa de HTML5**

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

El atributo type="email" proporciona validación automática del formato de correo. El atributo required hace que el campo sea obligatorio.

Para contraseñas, añadí un patrón de validación:

<input type="password" id="password" name="password" required pattern=".{8,}"

       title="La contraseña debe tener al menos 8 caracteres">

El atributo pattern=".{8,}" valida que la contraseña tenga al menos 8 caracteres. El atributo title muestra un mensaje cuando el patrón no coincide.

**2. Estilos CSS para estados de validación**

/\* Validación nativa con CSS \*/

input:required:invalid,

input:focus:invalid {

    border-color: #e74c3c;

    background-color: rgba(231, 76, 60, 0.05);

}

input:required:valid {

    border-color: #2ecc71;

    background-color: rgba(46, 204, 113, 0.05);

}

Estos selectores CSS:

* :required:invalid - Aplica estilos cuando un campo obligatorio está incompleto o tiene formato incorrecto
* :focus:invalid - Aplica estilos cuando un campo con formato incorrecto tiene el foco
* :required:valid - Aplica estilos cuando un campo obligatorio está correctamente completado

**3. Mensajes de ayuda en lugar de errores**

<input type="text" id="first-name" name="first-name" required>

<span class="hint">Ingrese su nombre de pila</span>

.hint {

    color: #7f8c8d;

    font-size: 0.8rem;

    margin-top: 0.25rem;

    font-style: italic;

}

/\* Mostrar ayuda adicional al enfocar \*/

input:focus + .hint {

    color: #3498db;

}

Mensajes de ayuda permanentes que cambian de color cuando el campo tiene el foco.

**4. Estructura Flexbox (manteniéndose igual)**

Las propiedades Flexbox:

.form-row {

    display: flex;

    flex-wrap: wrap;

    gap: 1rem;

    margin-bottom: 1rem;

}

.form-group {

    flex: 1 1 calc(50% - 0.5rem);

    min-width: 250px;

    display: flex;

    flex-direction: column;

}

Esta estructura permite:

* Campos que se colocan uno al lado del otro en pantallas grandes
* Campos que se apilan verticalmente en pantallas pequeñas
* Distribución equilibrada del espacio disponible

**5. Botones de acción**

<div class="form-actions">

    <a href="#" class="btn btn-secondary">Cancelar</a>

    <button type="submit" class="btn btn-primary">Registrarse</button>

</div>

El botón "Cancelar" ahora es un enlace simple y el botón "Registrarse" utiliza el comportamiento de envío de formulario nativo.

**Beneficios**

1. **HTML semántico**: HTML5 para formularios.
2. **Pseudo-clases CSS**: utilizar :valid, :invalid, :focus y otras pseudo-clases para crear interfaces interactivas.
3. **Flexbox práctico**: Flexbox resuelve problemas reales de diseño en formularios.