

**UNIVERSIDAD LAICA ELOY ALFARO DE MANABI**

**NOMBRES Y APELLIDOS**

**JORK DAVIS LUCAS GARCIA**

**FECHA**

**05/09/2025**

**MATERIA**

**APLICACIONES PARA EL CLIENTE WEB**

**PARALELO**

**“A”**

**ACTIVIDAD**

**INGRESO DE DATOS DE CLIENTES CON HTML, CSS Y  
JAVASCRIPT**

## INGRESO DE DATOS DE CLIENTE CON HTML, CSS Y JAVASCRIPT

### 1. Index.html

Define la estructura de la pagina web y el formulario. Permite al usuario ingresar los datos del cliente (nombre, correo, teléfono) y muestra la tabla con los clientes registrados.

I. Sirve para mostrar el título de la página:

```
<h1>Registro de Clientes</h1>
```

II. Sirve para crear el formulario donde se ingresan los datos del cliente:

```
<form id="clienteForm">
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre" required>

  <label for="email">Correo electrónico:</label>
  <input type="email" id="email" name="email" required>

  <label for="telefono">Teléfono:</label>
  <input type="tel" id="telefono" name="telefono" required>

  <button type="submit">Registrar</button>
</form>
```

III. Mostrar mensajes de error o confirmación:

```
<div id="resultado"></div>
```

IV. Mostrar la lista de clientes registrados en formato tabla:

```
<h2>Clientes Registrados</h2>
<table id="tablaClientes">
  <thead>
    <tr>
      <th>Nombre</th>
      <th>Correo electrónico</th>
      <th>Teléfono</th>
    </tr>
  </thead>
  <tbody>
    <!-- Aquí van los datos -->
  </tbody>
</table>
```

## V. Cargar la funcionalidad del formulario y la tabla:

```
<script src="script.js"></script>
```

## 2. Styles.css

Da estilos visuales modernos y atractivos al formulario y a la tabla. Hace que la pagina se vea profesional y ordenada

### I. Dar estilo general y separar el contenido del borde de la página:

```
body {  
  font-family: Arial, sans-serif;  
  background: #f2f2f2;  
  padding: 40px;  
}
```

### II. Centrar los títulos principales y secundarios:

```
h1, h2 {  
  text-align: center;  
}
```

### III. Dar apariencia moderna y centra al formulario:

```
form {  
  background: #fff;  
  padding: 20px;  
  margin: 0 auto 30px auto;  
  width: 320px;  
  border-radius: 8px;  
  box-shadow: 0 2px 8px rgba(0,0,0,0.1);  
}
```

### IV. Estilizar los campos de entrada y las etiquetas:

```
label {  
  margin-top: 10px;  
  font-weight: bold;  
}  
input {  
  width: 95%;  
  padding: 8px;  
  margin-bottom: 12px;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
}
```

V. Estilizar el botón y su efecto al pasar el mouse:

```
button {  
  background: #007bff;  
  color: #fff;  
  border: none;  
  padding: 10px;  
  border-radius: 4px;  
  width: 100%;  
  cursor: pointer;  
  margin-top: 10px;  
  font-size: 16px;  
}  
button:hover {  
  background: #0056b3;  
}
```

VI. Dar estilo a los mensajes de validación y éxito:

```
#resultado {  
  margin-top: 20px;  
  text-align: center;  
  font-weight: bold;  
}
```

VII. Dar formato y color a la tabla de clientes registrados:

```
#tablaClientes {  
  margin: 0 auto;  
  margin-top: 20px;  
  width: 80%;  
  border-collapse: collapse;  
  background: #fff;  
  box-shadow: 0 2px 8px rgba(0,0,0,0.1);  
}  
#tablaClientes th, #tablaClientes td {  
  border: 1px solid #ccc;  
  padding: 10px;  
  text-align: left;  
}  
#tablaClientes th {  
  background-color: #007bff;  
  color: #fff;  
}
```

## 3. Script.js

Gestiona la funcionalidad:

- Valida los datos ingresados (que no estén vacíos y que el correo tenga “ @ ”)
- Registra el cliente en una lista
- Actualiza la tabla para mostrarlos clientes registrados
- Muestra mensajes de error o éxito

I. Crear el arreglo donde se guardarán los clientes registrados:

```
const clientes = [];
```

II. Captura el envío del formulario – Validar que todos los campos estén llenos y el correo contenga:

```
document.getElementById("clienteForm").addEventListener("submit", function(event) {
    event.preventDefault();

    // Tomar los datos
    const nombre = document.getElementById("nombre").value.trim();
    const email = document.getElementById("email").value.trim();
    const telefono = document.getElementById("telefono").value.trim();

    // Validaciones básicas
    if (nombre === "" || email === "" || telefono === "") {
        mostrarMensaje("Completa todos los campos.", "red");
        return;
    }
    if (!email.includes("@")) {
        mostrarMensaje("Correo inválido.", "red");
        return;
    }

    // Registrar cliente
    clientes.push({ nombre, email, telefono });
    mostrarMensaje("¡Cliente registrado!", "green");
    actualizarTabla();
    document.getElementById("clienteForm").reset();
});
```

III. Mostrar mensajes de error o éxito en pantalla:

```
// Mostrar mensaje
function mostrarMensaje(mensaje, color) {
    const div = document.getElementById("resultado");
    div.textContent = mensaje;
    div.style.color = color;
}
```

## IV. Limpiar la tabla y mostrar todos los clientes registrados:

```
// Actualizar tabla
function actualizarTabla() {
  const tbody = document.querySelector("#tablaClientes tbody");
  tbody.innerHTML = "";
  clientes.forEach(cliente => {
    const fila = document.createElement("tr");
    fila.innerHTML = `
      <td>${cliente.nombre}</td>
      <td>${cliente.email}</td>
      <td>${cliente.telefono}</td>
    `;
    tbody.appendChild(fila);
  });
}
```

## 4. Código:

### I. HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Registro de Clientes</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Registro de Clientes</h1>
  <form id="clienteForm">
    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" name="nombre" required>

    <label for="email">Correo electrónico:</label>
    <input type="email" id="email" name="email" required>

    <label for="telefono">Teléfono:</label>
    <input type="tel" id="telefono" name="telefono" required>

    <button type="submit">Registrar</button>
  </form>
  <div id="resultado"></div>

  <h2>Clientes Registrados</h2>
  <table id="tablaClientes">
    <thead>
      <tr>
        <th>Nombre</th>
        <th>Correo electrónico</th>
        <th>Teléfono</th>
      </tr>
    </thead>
    <tbody>
      <!-- Aquí van los datos -->
    </tbody>
  </table>

  <script src="script.js"></script>
</body>
</html>
```

## II. CSS:

```
body {
  font-family: Arial, sans-serif;
  background-color: #f0f0f0;
  padding: 10px;
}
h1, h2 {
  text-align: center;
}
form {
  background-color: #fff;
  padding: 20px;
  margin: 10px auto;
  width: 500px;
  border: 1px solid #ccc;
  border-radius: 10px;
  box-shadow: 0 2px 4px rgba(0,0,0,0.1);
}
label {
  margin-top: 10px;
  font-weight: bold;
}
input {
  width: 90%;
  padding: 5px;
  margin-bottom: 10px;
  border: 1px solid #ccc;
  border-radius: 4px;
}
button {
  background-color: #007bff;
  color: #fff;
  border: none;
  padding: 10px;
  border-radius: 4px;
  width: 100%;
  cursor: pointer;
  margin-top: 10px;
  font-size: 16px;
}
button:hover {
  background-color: #0056b3;
}
#resultado {
  margin-top: 20px;
  text-align: center;
  font-weight: bold;
}
#tablaClientes {
  margin: 10px auto;
  margin-top: 20px;
  width: 80%;
  border-collapse: collapse;
  background-color: #fff;
  border: 1px solid #ccc;
  box-shadow: 0 2px 4px rgba(0,0,0,0.1);
}
#tablaClientes th, #tablaClientes td {
  border: 1px solid #ccc;
  padding: 10px;
  text-align: left;
}
#tablaClientes th {
  background-color: #007bff;
  color: #fff;
}
```

## III. JAVASCRIPT:

```
const clientes = [];

document.getElementById("clienteForm").addEventListener("submit", function(event) {
  event.preventDefault();

  // Tomar los datos
  const nombre = document.getElementById("nombre").value.trim();
  const email = document.getElementById("email").value.trim();
  const telefono = document.getElementById("telefono").value.trim();

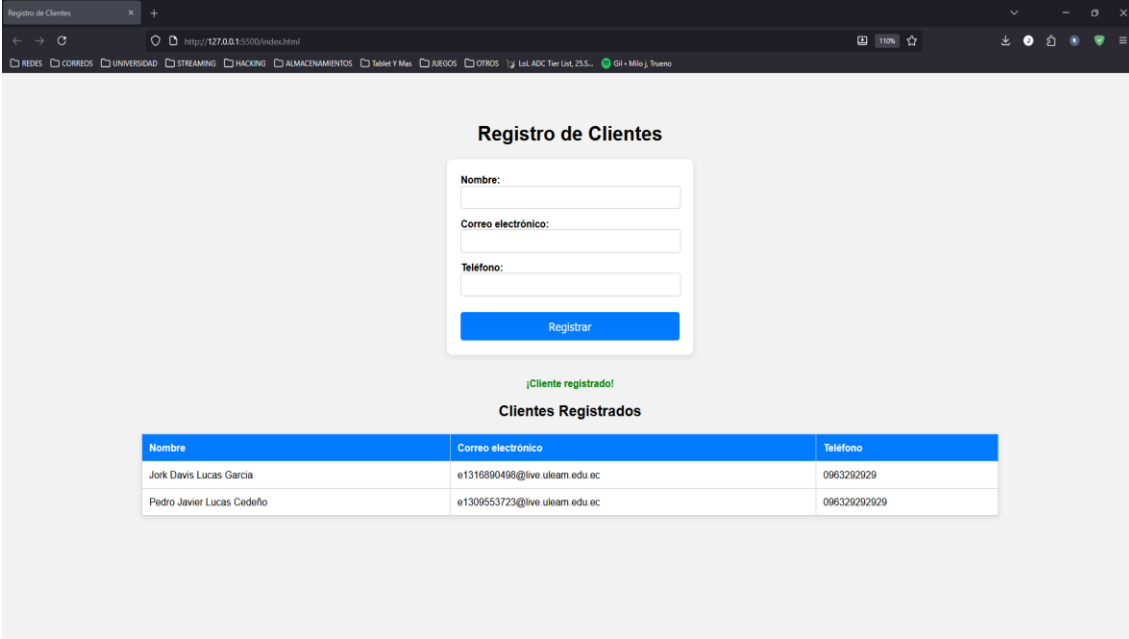
  // Validaciones básicas
  if (nombre === "" || email === "" || telefono === "") {
    mostrarMensaje("Completa todos los campos.", "red");
    return;
  }
  if (!email.includes("@")) {
    mostrarMensaje("Correo inválido.", "red");
    return;
  }

  // Registrar cliente
  clientes.push({ nombre, email, telefono });
  mostrarMensaje("Cliente registrado!", "green");
  actualizarTabla();
  document.getElementById("clienteForm").reset();
});

// Mostrar mensaje
function mostrarMensaje(mensaje, color) {
  const div = document.getElementById("resultado");
  div.textContent = mensaje;
  div.style.color = color;
}

// Actualizar tabla
function actualizarTabla() {
  const tbody = document.querySelector("#tablaClientes tbody");
  tbody.innerHTML = "";
  clientes.forEach(cliente => {
    const fila = document.createElement("tr");
    fila.innerHTML = `
      <td>${cliente.nombre}</td>
      <td>${cliente.email}</td>
      <td>${cliente.telefono}</td>
    `;
    tbody.appendChild(fila);
  });
}
```

## 5. EJECUCIÓN DEL CODIGO:



**Registro de Clientes**

Nombre:

Correo electrónico:

Teléfono:

¡Cliente registrado!

**Clientes Registrados**

Nombre	Correo electrónico	Teléfono
Jork Davis Lucas Garcia	e1316890498@live.uleam.edu.ec	0963292929
Pedro Javier Lucas Cedeño	e1309553723@live.uleam.edu.ec	0963292929

Este sistema permite registrar clientes de manera sencilla, aplicando validaciones básicas para asegurar que los datos sean los correctos al momento de ingresarlos. También muestra los datos organizados en una tabla facilitando la consulta.