



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA



Departamento de Computación  
Programación Integrativa de Componentes

Nombre: Carlos Pogo  
NRC: 16496

Tarea 1.3: Aplicaciones para web components

### Tema:

El proyecto consiste en desarrollar una aplicación web para la gestión de notas, donde los usuarios pueden crear, ver y editar notas de manera intuitiva. La aplicación utilizará arquitecturas basadas en Web Components para facilitar la reutilización de componentes y la organización modular del código.

### Estructura:

#### Tarea1.3

```
├─ index.html
├─ components/
│   └─ note-board.js
│   └─ note-item.js
│   └─ app-navigation.js
├─ styles/
│   └─ styles.css
├─ scripts/
│   └─ main.js
```

### Código:

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tablero de Notas</title>
  <link rel="stylesheet" href="styles/styles.css">
</head>
<body>
  <!-- Componente de navegación -->
  <app-navigation></app-navigation>

  <!-- Componente del tablero de notas -->
  <note-board></note-board>

  <!-- Scripts -->
  <script type="module" src="scripts/main.js"></script>
  <script type="module" src="/components/app-navigation"></script>
  <script type="module" src="/components/note-item"></script>
</body>
</html>
```

note-board.js

```
class NoteBoard extends HTMLElement {
  constructor() {
    super();
    this.attachShadow({ mode: 'open' });
    this.notes = []; // Array para almacenar las notas
    this.render(); // Renderizado inicial del tablero de notas
  }

  connectedCallback() {
    this.render();
  }

  addNote() {
    const note = prompt("Enter your note:"); // Prompt para ingresar la nota
    if (note) {
      this.notes.push(note); // Agregar la nota al array
      this.render(); // Renderizar nuevamente el tablero
    }
  }
}
```

```

    }

    removeNote(index) {
      this.notes.splice(index, 1); // Eliminar nota del array
      this.render(); // Renderizar nuevamente el tablero
    }

    render() {
      this.shadowRoot.innerHTML = `
        <style>
          .note-board {
            padding: 20px;
          }
          .note-item {
            margin: 10px 0;
          }
        </style>
        <div class="note-board">
          <button id="add-note">Add Note</button>
          ${this.notes.map((note, index) => `
            <note-item class="note-item" text="${note}"
index="${index}"></note-item>
          `).join('')}
        </div>
      `;

      // Evento para agregar nota
      this.shadowRoot.querySelector('#add-note').addEventListener('click', ()
=> this.addNote());
    }
  }

  customElements.define('note-board', NoteBoard);

```

note-item.js

```

class NoteItem extends HTMLElement {
  constructor() {
    super();
    this.attachShadow({ mode: 'open' });
  }

  connectedCallback() {
    this.render();
  }
}

```

```

    }

    render() {
      const text = this.getAttribute('text'); // Obtener el texto de la nota
      const index = this.getAttribute('index'); // Obtener el índice de la nota
      this.shadowRoot.innerHTML = `
        <style>
          .note {
            border: 1px solid #ccc;
            padding: 10px;
            border-radius: 5px;
          }
          .note button {
            margin-left: 10px;
          }
        </style>
        <div class="note">
          ${text} <!-- Mostrar el texto de la nota -->
          <button id="delete-note">Delete</button>
        </div>
      `;

      // Evento para eliminar la nota
      this.shadowRoot.querySelector('#delete-note').addEventListener('click',
    () => {
      // Disparar evento 'delete-note' con el índice de la nota como
      detalle
      this.dispatchEvent(new CustomEvent('delete-note', {
        detail: { index: parseInt(index) }, // Convertir el índice a
      entero
        bubbles: true,
        composed: true
      }));
    });
  }
}

customElements.define('note-item', NoteItem);

```

app-navigation.js

```

// Definición del componente de navegación
class AppNavigation extends HTMLElement {
  constructor() {

```

```

    super();

    // Creación del shadow DOM para el componente
    this.attachShadow({ mode: 'open' });

    // HTML del componente (puedes personalizar aquí)
    this.shadowRoot.innerHTML = `
      <nav>
        <ul>
          <li><a href="#">Inicio</a></li>
          <li><a href="#">Notas</a></li>
          <li><a href="#">Configuración</a></li>
        </ul>
      </nav>
    `;
  }
}

// Registrar el componente personalizado en el DOM
customElements.define('app-navigation', AppNavigation);

```

main.js

```

// Importar los componentes necesarios
import '../componets/app-navigation.js';
import '../componets/note-board.js';
import '../componets/note-item.js';
// Escuchar el evento 'delete-note' para eliminar una nota
document.addEventListener('delete-note', (e) => {
  // Obtener la instancia del tablero de notas
  const noteBoard = document.querySelector('note-board');
  // Llamar al método para eliminar la nota, pasando el índice desde el detalle
  // del evento
  noteBoard.removeNote(e.detail.index);
});

```

Link video:

[https://uespe-my.sharepoint.com/:v/g/personal/cepogo\\_espe\\_edu\\_ec/EdCtyesrGoNHvxLy68xM7rAB7j4ugE8HErfG8hPdaR0SBQ?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAIoiJPbmVEcml2ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXciLCJyZWZlcnJhbFZpZXciOiJNeUZpbGVzTGlua0NvcHkifX0&e=BV3UCC](https://uespe-my.sharepoint.com/:v/g/personal/cepogo_espe_edu_ec/EdCtyesrGoNHvxLy68xM7rAB7j4ugE8HErfG8hPdaR0SBQ?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAIoiJPbmVEcml2ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXciLCJyZWZlcnJhbFZpZXciOiJNeUZpbGVzTGlua0NvcHkifX0&e=BV3UCC)

Hosting

[https://14365-cepogo.github.io/Tarea1.3 Integrativa PogoCarlos/](https://14365-cepogo.github.io/Tarea1.3_Integrativa_PogoCarlos/)