

Propósito y Funcionalidad

Sistema de búsqueda y selección de dispositivos con autocompletado, integrado con el state management global.

Características Técnicas

Arquitectura de Búsqueda

```
javascript
```

```
// Debounce para performance

let searchTimeout;

searchInput.addEventListener('input', (e) => {

  clearTimeout(searchTimeout);

  searchTimeout = setTimeout(() => {

    const searchTerm = e.target.value.trim();

    if (searchTerm.length > 0) {

      filterDevices(searchTerm);

    }

  }, 300); // 300ms debounce

});

// Algoritmo de filtrado multi-campo

function filterDevices(searchTerm) {

  const filtered = currentDevices.filter(device =>
```

```

        device.nombre.toLowerCase().includes(searchTerm.toLowerCase()) ||

        device.id_dispositivo.toLowerCase().includes(searchTerm.toLowerCase()
        ) ||

        device.tipo?.toLowerCase().includes(searchTerm.toLowerCase())

    );
}

```

Sistema de Eventos Personalizados

```

javascript

deviceElement.onclick = () => {

    selectDevice(device); // Actualizar estado global

    // Disparar evento personalizado

    container.dispatchEvent(new CustomEvent('deviceSelected', {

        detail: {

            device: device,

            timestamp: Date.now(),

            source: 'deviceSelector'

        }

    }));

};

```

Visualización de Estado en Tiempo Real

```

javascript

```

```
// Mapeo de estados a colores
```

```
const statusColor = device.estado === 'en_linea' ? '#4CAF50' :  
  
    device.estado === 'fuera_linea' ? '#FF9800' :  
'#F44336';
```

```
// HTML dinámico con template strings
```

```
deviceElement.innerHTML = `  
  
    <div style="display: flex; justify-content: space-between;  
    align-items: center;">  
  
        <div>  
  
            <strong>${device.nombre}</strong>  
  
            <div style="font-size: 0.9em; color: #666;">  
  
                ID: ${device.id_dispositivo} | Tipo: ${device.tipo || 'N/A'}  
  
            </div>  
  
            ${device.ubicacion ?  
  
                `<div style="font-size: 0.8em; color: #888;">📍  
                ${device.ubicacion}</div>` :  
  
                ''}  
  
        </div>  
  
        <div style="display: flex; align-items: center; gap: 5px;">  
  
            <span style="width: 8px; height: 8px; border-radius: 50%;  
  
                background-color: ${statusColor};"></span>  
  
            <span style="font-size: 0.8em; color:  
            #666;">${device.estado}</span>
```

```
</div>
```

```
</div>
```

```
`;
```

Componente de Información del Dispositivo

javascript

```
export function selectedDeviceInfoWidget() {

  const { selectedDevice } = getState();

  if (!selectedDevice) {

    return el("div", {

      class: "card",

      style: "text-align: center; color: #666; padding: 20px;"

    }, "No hay dispositivo seleccionado");

  }

  // Grid layout responsive

  return el("div", { class: "card" },

    el("h3", {}, "Dispositivo Seleccionado"),

    el("div", {

      style: "display: grid; grid-template-columns: 1fr 1fr; gap: 15px;"

    },
```

```

// Columna información

el("div", {},

    el("h4", {}, "Información General"),

    el("ul", { style: "list-style: none; padding: 0;" },

        el("li", { style: "margin-bottom: 8px;" },

            el("strong", {}, "Nombre: "), selectedDevice.nombre

        ),

        // ... más campos

    )

),

// Columna estado

el("div", {},

    el("h4", {}, "Estado"),

    // Indicador visual de estado

    el("div", {

        style: "display: flex; align-items: center; gap: 10px;
margin-bottom: 10px;"

    },

        el("span", {

            style: `width: 12px; height: 12px; border-radius: 50%;

                background-color: ${statusColor};`

        },

```

```
                el("span", { style: "text-transform: capitalize;" },
selectedDevice.estado)

            )

        )

    )

);

}
```

Métodos Públicos

- `deviceSelectorWidget()`: Componente principal de búsqueda
- `selectedDeviceInfoWidget()`: Componente de información
- Evento `deviceSelected`: Notifica selecciones