



Feature: Selección de Impresora Bluetooth



Descripción General

Se implementó funcionalidad completa para que los cobradores de campo puedan **escanear, seleccionar y guardar** su impresora Bluetooth desde una lista de dispositivos disponibles, utilizando **Web Bluetooth API**.



Problema Resuelto

Antes:

- ✗ El cobrador tenía que ingresar manualmente el nombre de su impresora
- ✗ Alto riesgo de errores de escritura
- ✗ No había forma de verificar si el nombre era correcto
- ✗ Dificultad para usuarios no técnicos

Después:

- ✓ El sistema escanea automáticamente dispositivos Bluetooth cercanos
- ✓ Lista interactiva de impresoras disponibles
- ✓ Selección con un solo clic
- ✓ El nombre se guarda automáticamente
- ✓ Persistencia de configuración en base de datos



Implementación Técnica

1. Web Bluetooth API

Se utiliza la API nativa del navegador para acceder a dispositivos Bluetooth:

```
const device = await navigator.bluetooth.requestDevice({
  acceptAllDevices: true,
  optionalServices: ['000018f0-0000-1000-8000-00805f9b34fb'] // Servicio común de impresoras
});
```

2. Detección de Compatibilidad

```
useEffect(() => {
  if (typeof navigator !== 'undefined' && 'bluetooth' in navigator) {
    setBluetoothAvailable(true);
  }
}, []);
```

3. Flujo de Selección



Interfaz de Usuario

Componentes Agregados

1. Botón de Búsqueda Bluetooth

```
<Button
  type="button"
  variant="outline"
  onClick={scanBluetoothDevices}
  disabled={scanning}
>
  <Bluetooth className="h-4 w-4 mr-2" />
  Buscar
</Button>
```

2. Badge de Estado

```
{bluetoothAvailable && (
  <Badge variant="outline" className="gap-1">
    <Radio className="h-3 w-3" />
    Bluetooth disponible
  </Badge>
)}
```

3. Alerta de Compatibilidad

Si el navegador no soporta Web Bluetooth, se muestra:

-  Mensaje de advertencia
-  Instrucciones para habilitar Bluetooth
-  Lista de navegadores compatibles

Persistencia de Datos

Base de Datos (Ya existente)

La configuración se guarda en la tabla `users` :

- `impresoraNombre` (String): Nombre del dispositivo seleccionado
- `impresoraAnchoPapel` (Int): 58mm o 80mm
- `impresoraTamanoFuente` (String): pequeña, mediana, grande
- `impresoraAutoImprimir` (Boolean): Auto-imprimir tickets

API Endpoint

```
POST /api/users/printer-config
Body: {
    impresoraNombre: "Epson TM-T20",
    impresoraAnchoPapel: 80,
    impresoraTamanoFuente: "mediana",
    impresoraAutoImprimir: true
}
```

Casos de Uso

Escenario 1: Usuario con Bluetooth Disponible

1. Cobrador abre “Mi Impresora”
2. Ve badge “Bluetooth disponible” 
3. Click en “Buscar”
4. El navegador muestra lista de dispositivos Bluetooth
5. Selecciona su impresora “Epson TM-T20”
6. El nombre se auto-completa en el input
7. Click “Guardar Configuración”
8.  Configuración guardada

Escenario 2: Usuario sin Bluetooth Web

1. Cobrador abre “Mi Impresora”
2. Ve alerta  “Bluetooth Web no disponible”
3. Se le muestra:
 - Navegadores compatibles (Chrome, Edge, Samsung Internet)
 - Pasos para habilitar Bluetooth
 - Opción de ingresar nombre manualmente

Escenario 3: Error de Escaneo

1. Usuario cancela la selección → Toast: “No se seleccionó ningún dispositivo”
2. Bluetooth bloqueado → Toast: “Bluetooth bloqueado. Verifica permisos”
3. Otro error → Toast: “Error al escanear dispositivos Bluetooth”

Compatibilidad de Navegadores

Navegador	Versión Mínima	Soporte
Chrome	56+	 Completo
Edge	79+	 Completo
Samsung Internet	6.4+	 Completo
Opera	43+	 Completo
Firefox		No soportado
Safari		No soportado

Requisitos Adicionales

-  **HTTPS obligatorio** (o localhost para desarrollo)
-  Android 6.0+ o Windows 10+ o ChromeOS
-  Bluetooth activado en el dispositivo

Archivos Modificados

/app/dashboard/mi-impresora/page.tsx

```
+ import { Bluetooth, Radio } from 'lucide-react';
+ import { Badge } from '@/components/ui/badge';

+ interface BluetoothPrinter {
+   id: string;
+   name: string;
+ }

+ const [scanning, setScanning] = useState(false);
+ const [bluetoothAvailable, setBluetoothAvailable] = useState(false);

+ const scanBluetoothDevices = async () => {
+   const device = await navigator.bluetooth.requestDevice({
+     acceptAllDevices: true,
+     optionalServices: ['000018f0-0000-1000-8000-00805f9b34fb']
+   });
+   setConfig({ ...config, impresoraNombre: device.name });
+ };
```

Logging y Debugging

Console Logs Implementados

```

 Iniciando escaneo de dispositivos Bluetooth...
 Dispositivo seleccionado: { id: "...", name: "Epson TM-T20" }
 Error al escanear Bluetooth: NotNotFoundError
  
```

Toast Notifications

-  "Impresora 'Epson TM-T20' seleccionada correctamente"
-  "No se seleccionó ningún dispositivo"
-  "Bluetooth bloqueado. Verifica permisos"

Elementos Visuales

Estados del Botón

```

// Estado Normal
<Bluetooth className="h-4 w-4 mr-2" />
Buscar

// Estado Escaneando
<Loader2 className="h-4 w-4 animate-spin mr-2" />
Buscando...
  
```

Colores y Badges

-  Bluetooth disponible: Badge azul con ícono Radio
-  Advertencia: Fondo amarillo con bordes amber
-  Consejo: Fondo azul con texto informativo

Métricas de Éxito

KPIs Esperados

-  **Tiempo de configuración:** Reducción de 2 min → 30 seg
-  **Errores de escritura:** Reducción del 95%
-  **Satisfacción de usuario:** Mejora esperada del 80%
-  **Re-configuraciones:** Reducción del 70%

Próximas Mejoras (Futuro)

Fase 2 - Impresión Real

- [] Conexión directa a impresora Bluetooth
- [] Envío de comandos ESC/POS

- [] Generación de tickets en formato térmico
- [] Callback de confirmación de impresión

Fase 3 - Gestión Avanzada

- [] Guardar múltiples impresoras
 - [] Detección automática de impresora cercana
 - [] Historial de impresiones
 - [] Estado de batería/papel
-

Checklist de Implementación

- [x] Agregar detección de Web Bluetooth API
 - [x] Implementar función `scanBluetoothDevices()`
 - [x] Crear UI con botón “Buscar”
 - [x] Agregar badge de estado Bluetooth
 - [x] Implementar manejo de errores
 - [x] Agregar alertas de compatibilidad
 - [x] Actualizar información de impresoras compatibles
 - [x] Testing de selección de dispositivos
 - [x] Verificar persistencia en base de datos
 - [x] Documentación completa
-

Notas para Desarrolladores

Testing Local

1. Usar Chrome 56+ o Edge 79+
2. Servidor debe correr en HTTPS o localhost
3. Tener impresora Bluetooth encendida y en modo pairing
4. Habilitar permisos de Bluetooth en el navegador

Debugging Web Bluetooth

```
// En Chrome DevTools Console
navigator.bluetooth.getAvailability()
  .then(available => console.log('Bluetooth available:', available));
```

Permisos de Bluetooth

En Chrome, verificar:

1. `chrome://settings/content/bluetooth`
 2. Asegurar que el sitio tiene permisos
 3. Reiniciar navegador si es necesario
-

Impacto en Bundle Size

- **Peso agregado:** ~2KB (solo lógica, sin librerías externas)
 - **API nativa:** Web Bluetooth no requiere dependencias
 - **Iconos:** Lucide React (ya incluido)
-

Referencias Técnicas

- [Web Bluetooth API Specification](https://webbluetoothcg.github.io/web-bluetooth/) (<https://webbluetoothcg.github.io/web-bluetooth/>)
 - [MDN Web Bluetooth](https://developer.mozilla.org/en-US/docs/Web/API/Web_Bluetooth_API) (https://developer.mozilla.org/en-US/docs/Web/API/Web_Bluetooth_API)
 - [Chrome Bluetooth Samples](https://googlechrome.github.io/samples/web-bluetooth/) (<https://googlechrome.github.io/samples/web-bluetooth/>)
 - [ESC/POS Protocol](https://download4.epson.biz/sec_pubs/pos/reference_en/escpos/) (https://download4.epson.biz/sec_pubs/pos/reference_en/escpos/)
-

Fecha de Implementación: 17 de Noviembre, 2025

Versión: 1.5.0

Autor: DeepAgent - Abacus.AI

Estado:  Completado y Funcional