

Sistema de Activación/Desactivación de Conductores

Resumen

Se ha implementado un **sistema de activación/desactivación** de conductores que reemplaza la funcionalidad de eliminación. Esto proporciona una solución más segura y profesional para gestionar el acceso de conductores.

¿Por qué Activación/Desactivación en lugar de Eliminación?

Ventajas sobre la eliminación:

1.  **Preserva datos históricos** - No se pierden las inspecciones pasadas
2.  **Reversible** - Se puede reactivar si fue un error
3.  **Auditoría completa** - Mantiene trazabilidad de quién hizo qué
4.  **Reutilización de emails** - Al desactivar, el email queda disponible para futuro uso
5.  **Seguridad** - El conductor inactivo no puede iniciar sesión
6.  **Integridad referencial** - No rompe relaciones con inspecciones existentes

Funcionalidades Implementadas

Página de Conductores

Filtro de Estado:

-  **Todos los estados** - Muestra todos los conductores
-  **Solo Activos** - Muestra conductores que pueden iniciar sesión
-  **Solo Inactivos** - Muestra conductores desactivados

Badges Visuales:

-  **Badge Verde** - Conductor activo
-  **Badge Gris** - Conductor inactivo
-  **Iconos de color** - Verde para activos, gris para inactivos

Botón de Activar/Desactivar:

-  **Icono UserX (naranja)** - Desactivar conductor activo
-  **Icono UserCheck (verde)** - Activar conductor inactivo

2 Diálogo de Confirmación

Al Desactivar:

Desactivar conductor?

Al desactivar a [Nombre], **no** podrá iniciar sesión en la aplicación hasta que sea reactivado. Sus datos históricos se mantendrán intactos y podrás reactivarlo en cualquier momento.

Al Activar:

Activar conductor?

Al activar a [Nombre], podrá volver a iniciar sesión en la aplicación y realizar inspecciones normalmente.

3 Bloqueo de Login para Conductores Inactivos

En el Login (components/auth/login-form.tsx):

```
// Verificar si el usuario está activo
if (userData?.estado === 'inactivo') {
  await auth.signOut(); // Cerrar sesión inmediatamente
  setError('Tu cuenta ha sido desactivada. Contacta al administrador.');
  return;
}
```

En el Auth Context (contexts/auth-context.tsx):

```
// Verificar si el usuario está inactivo
if (userData.estado === 'inactivo') {
  await signOut(auth);
  setUser(null);
  setFirebaseUser(null);
}
```

Resultado:

- ✗ Conductor inactivo **NO PUEDE** iniciar sesión
- ✗ Si estaba logueado y lo desactivas, **SE CIERRA LA SESIÓN AUTOMÁTICAMENTE**
- ✓ Mensaje claro de por qué no puede acceder

4 Actualización Automática en Firestore

Cuando se activa/desactiva un conductor:

```
// Actualizar en conductores
await updateDoc(doc(db, 'conductores', conductorId), {
  estado: nuevoEstado,
  updatedAt: new Date().toISOString()
});

// Actualizar en users (para auth)
await updateDoc(doc(db, 'users', conductorId), {
  estado: nuevoEstado,
  updatedAt: new Date().toISOString()
});
```

Se actualiza en ambas colecciones:

- **conductores** - Para el listado de conductores
 - **users** - Para el login y auth context
-

5 Creación de Nuevos Conductores

Al crear un conductor nuevo, se establece **automáticamente** como activo:

```
const conductorData = {
  ...formData,
  userId: firebaseUser.uid,
  estado: 'activo', //  Activo por defecto
  createdAt: new Date().toISOString(),
  updatedAt: new Date().toISOString(),
};
```

Flujo de Trabajo

Desactivar un Conductor:

1. Admin va a /admin/conductores
2. Filtra por “Solo Activos” si quiere
3. Hace clic en el botón naranja con icono 
4. Confirma la desactivación
5. **Conductor queda inactivo**
6. **Ya no puede iniciar sesión**
7. **Email queda disponible para usar en el futuro**

Reactivar un Conductor:

1. Admin va a /admin/conductores
2. Filtra por “Solo Inactivos”
3. Hace clic en el botón verde con icono 
4. Confirma la reactivación
5. **Conductor queda activo nuevamente**
6. **Puede volver a iniciar sesión**



Campos Añadidos

Interface User :

```
export interface User {
  id: string;
  email: string;
  name: string;
  role: 'administrador' | 'conductor';
  estado: 'activo' | 'inactivo'; // ✓ NUEVO
  createdAt: string;
  updatedAt: string;
}
```

Interface Conductor :

```
export interface Conductor {
  id: string;
  nombre: string;
  cedula: string;
  numeroLicencia: string;
  categoriaLicencia: string;
  telefono: string;
  email: string;
  userId: string;
  estado: 'activo' | 'inactivo'; // ✓ NUEVO
  fotoUrl?: string;
  licenciaVencimiento?: string;
  createdAt: string;
  updatedAt: string;
}
```



Despliegue

La aplicación está desplegada en:

<https://inspeccionpesv.abacusai.app>

Los cambios incluyen:

- ✓ Sistema de activación/desactivación completo
- ✓ Filtros de estado
- ✓ Bloqueo de login para inactivos
- ✓ Badges visuales
- ✓ Documentación completa

❓ Preguntas Frecuentes

¿Qué pasa con las inspecciones del conductor desactivado?

R: Se mantienen intactas. El historial completo sigue disponible.

¿El conductor inactivo puede ver su perfil?

R: No. Se cierra automáticamente su sesión y no puede volver a iniciar sesión.

¿Se puede reutilizar el email de un conductor desactivado?

R: Sí, pero recomendamos reactivar el conductor existente en lugar de crear uno nuevo.

¿Los administradores pueden ser desactivados?

R: No. Solo los conductores tienen esta funcionalidad.

¿Cómo saber si un conductor está inactivo?

R: Usa el filtro “Solo Inactivos” o busca el badge gris en la lista.



Archivos Modificados

1. app/admin/conductores/page.tsx - Lista y gestión de conductores
 2. app/admin/conductores/nuevo/page.tsx - Creación con estado activo
 3. components/auth/login-form.tsx - Bloqueo de login
 4. contexts/auth-context.tsx - Verificación continua de estado
 5. lib/auth-types.ts - Tipos TypeScript actualizados
-



Resumen

El sistema de activación/desactivación proporciona:

- **Seguridad** - Control total sobre quién puede acceder
- **Flexibilidad** - Reversible en cualquier momento
- **Integridad** - No se pierden datos históricos
- **Profesionalismo** - Gestión empresarial de usuarios
- **Auditoría** - Trazabilidad completa

¡Ya no tienes que preocuparte por emails bloqueados o datos perdidos! 🎉