

## INFORME TÉCNICO: Recuperación, Optimización y Despliegue de "Smart Boleta"

**Proyecto:** Sistema de Gestión de Boletas (Smart Boleta) **Fecha:** 24 de Diciembre de 2025

**Responsable:** Patrick Fernando Lopez Meza

### 1. Resumen Ejecutivo

Durante las pruebas de rutina del sistema web "Smart Boleta", se detectó la imposibilidad de acceder al aplicativo. Mediante el uso de las herramientas de desarrollador (Consola del Navegador), se identificó que la raíz del problema residía en la inactividad del servicio de base de datos (Supabase), el cual se había pausado automáticamente.

Tras reactivar el servicio ("Resume"), el sistema presentó nuevos errores críticos de sintaxis (SyntaxError) derivados de conflictos en la declaración de variables tras la actualización. Se procedió a corregir el código fuente archivo por archivo. Adicionalmente, para evitar futuros problemas de despliegue manual, se implementó una integración continua conectando el repositorio de **GitHub** con **Netlify**, garantizando así un flujo de trabajo profesional y automatizado.

### 2. Diagnóstico Inicial

Al intentar ingresar al sistema, la interfaz no cargaba. El análisis técnico arrojó lo siguiente:

- **Estado del Backend:** El proyecto en Supabase se encontraba en estado INACTIVE debido a falta de uso reciente.
- **Estado del Frontend:** Tras reactivar la base de datos, la consola del navegador mostró el error: Uncaught SyntaxError: Identifier 'supabase' has already been declared. Esto impedía la ejecución de cualquier script de lógica de negocio (Login, Dashboard, etc.).

**(index):48 Uncaught SyntaxError: Identifier 'supabase' has already been declared**

**at (index):48:133. Acciones Correctivas (Refactorización)**

Para solucionar el conflicto de sintaxis que bloqueaba la página, se realizó una intervención en el código JavaScript de todos los módulos HTML.

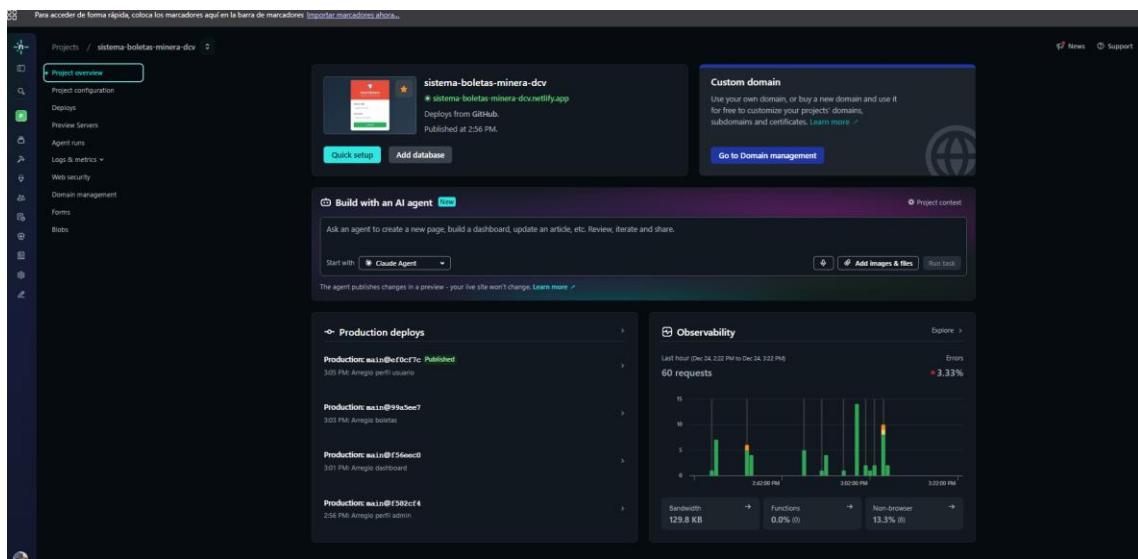
- **Aislamiento de Instancias:** Se detectó que la variable de conexión supabase generaba conflictos. Se refactorizó el código renombrando las instancias de conexión para hacerlas únicas en cada módulo:
  - index.html → Se cambió a clienteLogin.
  - admin-panel.html → Se cambió a clienteAdmin.
  - subir-boletas.html → Se cambió a clienteSubida.
  - dashboard.html y perfil.html → Se aplicaron cambios similares.

Esta acción eliminó los errores de consola y restableció la comunicación fluida con la base de datos.

### 4. Optimización del Despliegue (DevOps)

Como mejora sustancial al proyecto, se abandonó el método de subida manual de archivos para adoptar un estándar profesional de desarrollo:

- Control de Versiones (Git):** Se inicializó un repositorio local para llevar un historial detallado de cada cambio y corrección realizada.
- Repositorio Remoto (GitHub):** Se migró el código fuente a GitHub, asegurando un respaldo en la nube.
- Despliegue Automatizado (Netlify):** Se vinculó el repositorio de GitHub con Netlify.
  - Beneficio:* Ahora, cada corrección enviada desde Visual Studio Code se refleja automáticamente en la página web en producción, eliminando errores humanos por manipulación de archivos.



## 5. Conclusiones y Resultados

El sistema "Smart Boleta" se encuentra actualmente **operativo al 100%**. Se ha logrado:

- Restablecer el acceso a la base de datos.
- Eliminar bugs críticos de JavaScript (Sintaxis).
- Modernizar la infraestructura de despliegue (CI/CD).

El sistema es ahora más estable, seguro y fácil de mantener gracias a la integración con herramientas de desarrollo modernas.