

Problemas Encontrados en el Proyecto de Giras Técnicas

PROBLEMAS CRÍTICOS

1. No existe API/Backend para Login

- **Archivo afectado:** `js/login.js`
- **Problema:** El login es simulado en el frontend, solo redirige según el email hardcodeado
- **Líneas 26-37:** Simulación de login sin validación real con base de datos
- **Solución necesaria:** Crear archivo PHP para autenticación real (`api/login.php`)

2. No existe API/Backend para Registro

- **Archivo afectado:** `pages/register.html` y `js/register.js`
- **Problema:** No hay backend PHP para procesar el registro de nuevos usuarios
- **Solución necesaria:** Crear archivo PHP para registro (`api/register.php`)

3. No existe API para listar Giras

- **Archivos afectados:** `pages/user-dashboard.html` , `js/user-dashboard.js`
- **Problema:** El frontend necesita cargar las giras disponibles pero no hay endpoint
- **Solución necesaria:** Crear `api/giras.php` con métodos GET, POST, PUT, DELETE

4. No existe API para Inscripciones

- **Problema:** No hay endpoints para crear/listar/actualizar inscripciones
- **Solución necesaria:** Crear `api/inscripciones.php`

5. ComprobanteController.php no es accesible como API

- **Archivo:** `controllers/ComprobanteController.php`
- **Problema:** Es una clase pero no tiene punto de entrada (endpoint) para ser llamado desde JavaScript
- **Línea 31:** La action del formulario apunta a `../controllers/ComprobanteController.php` pero este archivo no ejecuta ninguna acción automáticamente
- **Solución:** Crear un archivo router o modificar el controller para que sea ejecutable

6. Falta archivo .htaccess para reescritura de URLs

- **Archivo:** `.htaccess` está vacío
- **Problema:** No hay configuración de Apache para manejar las rutas y los endpoints PHP
- **Solución:** Crear configuración adecuada de `.htaccess`

7. Páginas HTML hacen referencia a archivos que no existen

- `js/login.js` línea 30: redirige a `estudiante.html` (no existe)
- `js/login.js` línea 33: redirige a `coordinador.html` (no existe)
- **Solución:** Crear estas páginas o corregir las rutas

PROBLEMAS MEDIOS

8. Falta inicialización de sesiones PHP

- No hay manejo de sesiones para mantener usuarios autenticados
- Solución: Crear sistema de sesiones en `config/session.php`

9. Sin protección CSRF

- Los formularios no tienen tokens CSRF para prevenir ataques
- Solución: Implementar tokens CSRF en formularios

10. Rutas hardcodeadas

- Múltiples archivos tienen rutas absolutas como `/giras_tecnicas/`
- Problema: No funcionará en diferentes entornos
- Solución: Crear archivo de configuración con rutas base

11. Sin validación de tipos de usuario/roles

- No hay middleware o sistema para verificar roles antes de acceder a páginas
- Solución: Crear sistema de autenticación y autorización

PROBLEMAS MENORES

12. Código comentado en `admin.php`

- Líneas 2-3: Código de seguridad comentado
- Solución: Implementar verificación de roles

13. Inconsistencia en conexiones a BD

- Hay dos formas de conectar: `conexion.php` (mysqli) y `database.php` (PDO)
- Solución: Estandarizar a un solo método (preferible PDO)

14. Sin manejo de errores centralizado

- Los errores se muestran directamente al usuario
- Solución: Crear sistema de logging y mensajes de error amigables

15. Falta validación de inputs

- No hay sanitización en todos los endpoints
- Solución: Agregar filtros y validaciones

ARCHIVOS QUE FALTAN POR CREAR

1. `api/login.php` - Endpoint de autenticación
2. `api/register.php` - Endpoint de registro
3. `api/giras.php` - CRUD de giras
4. `api/inscripciones.php` - CRUD de inscripciones
5. `api/usuarios.php` - Gestión de usuarios
6. `config/session.php` - Manejo de sesiones
7. `config/config.php` - Configuraciones globales
8. `pages/estudiante.html` - Dashboard de estudiante

9. `pages/coordinador.html` - Dashboard de coordinador
10. `middleware/auth.php` - Verificación de autenticación
11. `utils/validators.php` - Funciones de validación
12. `utils/response.php` - Formateo de respuestas JSON



BASE DE DATOS

- **Estado:** NO CREADA
- Acción: Ejecutar scripts SQL en orden:
 1. `01_schema.sql`
 2. `02_comprobantes.sql`
 3. `03_insert.sql`



PRIORIDAD DE CORRECCIÓN

1. ☒ Crear y configurar base de datos MySQL
2. ☒ Crear APIs faltantes (login, register, giras, inscripciones)
3. ☒ Implementar sistema de sesiones
4. ☒ Corregir ComprobanteController para que funcione como endpoint
5. ☒ Crear páginas HTML faltantes
6. ☒ Configurar .htaccess si es necesario
7. ☒ Estandarizar conexiones a BD
8. ☒ Agregar validaciones y seguridad