

Auditoría de Rutas Absolutas - 28 Octubre 2025









Resumen Ejecutivo

Se realizó una auditoría exhaustiva del código para identificar problemas similares al fix de Prisma output path (rutas absolutas que fallan en Docker).

Estado:  **COMPLETO - SIN PROBLEMAS CRÍTICOS**

Metodología de Búsqueda

Se realizaron las siguientes búsquedas sistemáticas:

- 1.  Búsqueda de `/home/ubuntu` en todo el código fuente
- 2.  Búsqueda de `output =` en archivos de configuración
- 3.  Búsqueda de `/opt/hostedapp` (ruta de symlinks problemáticos)
- 4.  Revisión de archivos de configuración (next.config.js, tsconfig.json)
- 5.  Revisión de Dockerfile (instrucciones COPY, ADD, WORKDIR)
- 6.  Revisión de package.json scripts
- 7.  Búsqueda de symlinks en el proyecto
- 8.  Revisión de archivos .env


Hallazgos

PROBLEMAS RESUELTOS


1. Prisma Schema Output Path (CRÍTICO - RESUELTO)

Archivo: `app/prisma/schema.prisma`

Antes:

```
output = "/home/ubuntu/escalafin_mvp/app/node_modules/.prisma/client" #  ABSOLUTA
```

Después:

```
output = "../node_modules/.prisma/client" #  RELATIVA
```

Impacto: Bloqueaba completamente el build en Docker

Estado:  RESUELTO (commit c274e78)

2. Storage Config Component (MENOR - RESUELTO)

Archivo: app/components/admin/storage-config.tsx

Problema: Valores por defecto y placeholders con rutas absolutas específicas del entorno de desarrollo local.

Antes:

```
uploadDir: '/home/ubuntu/escalafin_mvp/uploads',
placeholder="/home/ubuntu/escalafin_mvp/uploads"
```

Después:

```
uploadDir: '/app/uploads',
placeholder="/app/uploads"
```

Impacto:

- No afecta la compilación
- Solo valores por defecto en UI de administración
- Usuarios pueden cambiar estos valores

Estado:  RESUELTO (este commit)


CONFIGURACIONES CORRECTAS

1. next.config.js

```
experimental: {
  outputFileTracingRoot: path.join(__dirname, '..'), //  RELATIVO
}
```


Estado:  CORRECTO - Usa path.join con rutas relativas


2. tsconfig.json

```
"paths": {
  "@/*": ["./*"] //  RELATIVO
}
```

Estado:  CORRECTO - Alias de paths son relativos


3. lib/prisma.ts

```
import { PrismaClient } from '@prisma/client' //  CORRECTO
export const prisma = globalForPrisma.prisma ?? new PrismaClient()
```



Estado:  CORRECTO - Import estándar de módulo npm

4. Dockerfile

```
WORKDIR /app
COPY app/ ./
COPY --from=builder /app/.next/standalone/app ./
```

Estado:  CORRECTO - Todas las rutas son relativas o del contexto de Docker

5. Variables de Entorno (.env)

-  No se encontraron rutas absolutas problemáticas
-  Todas las configuraciones usan variables de entorno correctamente

6. Uso de process.cwd()

Archivos:









- api/notifications/settings/route.ts
- api/admin/settings/route.ts

```
const SETTINGS_DIR = path.join(process.cwd(), 'user-settings');
```

Estado:  CORRECTO - `process.cwd()` es dinámico y funciona correctamente en Docker



Análisis de Impacto

Problema	Severidad	Impacto en Build	Impacto en Runtime	Estado
Prisma Output Path	 CRÍTICO	 Bloqueante	 Bloqueante	 RESUELTO
Storage Config UI	 MENOR	 Sin impacto	 Sin impacto	 RESUELTO



Archivos Modificados

Commit c274e78 (Prisma Fix):

- app/prisma/schema.prisma - Output path corregido

Este Commit (Storage Config):

- `app/components/admin/storage-config.tsx` - Valores por defecto actualizados
-

✓ Validaciones Realizadas

1. ✓ **Búsqueda de rutas absolutas en código fuente**
 - Comandos: `grep -r "/home/ubuntu"` , `grep -r "/opt/hostedapp"`
 - Resultado: Solo encontrados en UI (no críticos)
 2. ✓ **Revisión de archivos de configuración**
 - `next.config.js` ✓
 - `tsconfig.json` ✓
 - `package.json` ✓
 - `.env` files ✓
 3. ✓ **Revisión de Dockerfile**
 - Todas las instrucciones COPY, ADD, WORKDIR son correctas ✓
 4. ✓ **Verificación de symlinks**
 - No se encontraron symlinks problemáticos ✓
 5. ✓ **Revisión de imports y módulos**
 - Todos los imports usan rutas relativas o módulos npm ✓
-

🎯 Conclusiones

✓ Estado General: SALUDABLE








1. **El único problema crítico** (Prisma output path) ya fue resuelto
 2. **Problema menor** (Storage Config UI) resuelto en este commit
 3. **No se encontraron otros problemas similares**
 4. **La arquitectura del proyecto usa correctamente:**
 - Rutas relativas
 - Variables de entorno
 - Módulos npm estándar
 - Configuraciones portables
-

🚀 Recomendaciones

✓ Ya Implementadas:


1. ✓ Usar rutas relativas en `schema.prisma`
2. ✓ Usar rutas genéricas en valores por defecto de UI
3. ✓ Mantener configuraciones portables

Para el Futuro:

1. **Al agregar nuevas configuraciones:**
 -  NO usar rutas absolutas como `/home/usuario/...`
 -  Sí usar rutas relativas o variables de entorno
 -  Sí usar `process.cwd()` para rutas dinámicas en runtime
2. **Al agregar nuevos schemas/generators:**
 -  NO especificar output paths absolutos
 -  Sí usar output paths relativos al archivo de configuración
3. **Al crear valores por defecto en UI:**
 -  NO hardcodear rutas de desarrollo local
 -  Sí usar rutas genéricas o de producción

Patrones Correctos Identificados


Patrón 1: Rutas Relativas en Configuración

```
//  CORRECTO
"paths": {
  "@/*": ["./*"]
}
```

Patrón 2: Variables de Entorno

```
//  CORRECTO
bucketName: process.env.AWS_BUCKET_NAME || 'default-bucket'
```

Patrón 3: process.cwd() para Runtime

```
//  CORRECTO
const dir = path.join(process.cwd(), 'user-settings');
```

Patrón 4: __dirname para Contexto de Archivo

```
//  CORRECTO
outputFileTracingRoot: path.join(__dirname, '../')
```

Referencias

- **Fix de Prisma:** `FIX_PRISMA_OUTPUT_PATH_FINAL.md`
- **Estado Final:** `ESTADO_FINAL_FIX_PRISMA_28_OCT_2025.txt`
- **Changelog:** `CHANGELOG_VERSION_MERGE.md`



Notas Finales

Esta auditoría confirma que el proyecto está bien arquitecturado y no tiene problemas sistemáticos con rutas absolutas. El único problema crítico era el de Prisma schema, el cual ya fue resuelto.

El proyecto está listo para deployment en Docker/Kubernetes/Cloud sin problemas de portabilidad de rutas.

Fecha de Auditoría: 28 de octubre de 2025, 05:45 UTC

Última Actualización: 28 de octubre de 2025, 05:45 UTC

Próxima Revisión Recomendada: Antes de cada release mayor