

# Dockerfile v8.6 - Next.js Build Debug



## PROGRESO: Prisma Generate Funciona!

**v8.5 fue exitoso** - Prisma generate ahora completa correctamente.



## Nuevo Problema: Next.js Build

```
Build failed because of webpack errors
exit code: 1
```

Pero no vemos el detalle del error webpack.



## Debug Agregado en v8.6

### Logs Completos del Build

```
RUN npm run build 2>&1 | tee /tmp/build.log || \
(echo "❌ BUILD FALLÓ - LOGS COMPLETOS:" && \
cat /tmp/build.log && \
echo "=== VERIFICANDO ARCHIVOS ===" && \
ls -la app/ && \
echo "=== NODE_MODULES ===" && \
ls -la node_modules/@prisma/ && \
exit 1)
```

#### Qué hace:

- ☒ Captura STDOUT y STDERR completos
- ☒ Guarda logs en /tmp/build.log
- ☒ Si falla, muestra logs completos
- ☒ Verifica estructura de archivos
- ☒ Verifica que Prisma Client esté generado



## Logs Esperados

### Si hay error de TypeScript:

```
❌ BUILD FALLÓ - LOGS COMPLETOS:
Type error: Cannot find module '@prisma/client'
  at app/lib/db.ts:1:1
```

## Si hay error de webpack:

❌ BUILD FALLÓ - LOGS COMPLETOS:  
webpack compiled with 1 error

```
./app/lib/db.ts
Module not found: Can't resolve '@prisma/client'
```

## Si hay error de Next.js:

❌ BUILD FALLÓ - LOGS COMPLETOS:  
Error: Build optimization failed

## 🎯 Posibles Causas

### 1. Prisma Client No Disponible

**Síntoma:** Cannot find module '@prisma/client'

**Solución:** Verificar que Prisma generate creó los archivos correctamente

### 2. Imports Incorrectos

**Síntoma:** Module not found en archivos específicos

**Solución:** Corregir imports en el código

### 3. Variables de Entorno

**Síntoma:** Error al acceder a process.env

**Solución:** Verificar SKIP\_ENV\_VALIDATION=true

### 4. Dependencias Faltantes

**Síntoma:** Package X not found

**Solución:** Agregar al package.json

## ⚡ Diferencia con v8.5

Aspecto	v8.5	v8.6
Prisma generate	✅ Funciona	✅ Funciona
Logs de build	Mínimos	<b>Completos</b>
Debug info	No	<b>Sí</b>
Error visibility	Baja	<b>Alta</b>

---

## Próximos Pasos

1. **Rebuild con v8.6** → Veremos logs completos del error webpack
  2. **Identificar causa exacta** → Con los logs detallados
  3. **Aplicar fix específico** → Según el error encontrado
  4. **Build exitoso** → Deployment listo
- 

## Warnings de Seguridad

- SecretsUsedInArgOrEnv: ARG "OPENPAY\_PRIVATE\_KEY"
- SecretsUsedInArgOrEnv: ARG "EVOLUTION\_API\_TOKEN"
- SecretsUsedInArgOrEnv: ARG "NEXTAUTH\_SECRET"


**Estos son warnings, NO errores.**

Son valores placeholder para el build. Los valores reales se inyectan en runtime desde EasyPanel environment variables.

**No afectan el build, son solo advertencias de buenas prácticas.**

---

## Resumen

**LOGRO:**  Prisma generate ahora funciona (v8.5 exitoso)


**SIGUIENTE:**  Identificar error específico de Next.js build (v8.6 con logs)

**ESTADO:** Progresando, más cerca de la solución

---

**Versión:** 8.6

**Fecha:** 2025-10-06 18:55 GMT

**Estado:**  DEBUG NEXTJS BUILD

**Cambios:**

- Add: Logs completos de npm run build
  - Add: Captura de STDOUT/STDERR
  - Add: Verificación de archivos si falla
  - Maintain: Prisma generate funcional (v8.5)
- 

**Próximo paso:** Rebuild con v8.6 para ver el error específico de webpack/Next.js.