

Solución Alternativa: Deploy Sin Standalone

Fecha: 9 de Octubre, 2025

Problema: El build con standalone está fallando en EasyPanel

Solución: Usar `next start` en lugar de standalone

Problema Identificado

El build de Next.js está fallando con `exit code: 1` durante la generación del standalone output. Los logs muestran posibles problemas con:





1. **Peer dependencies** incompatibles:
 - `@typescript-eslint/parser` version 7.0.0 vs requested ^6.0.0-alpha
 - `eslint` version 9.24.0 vs requested ^8.57.0
 - `date-fns` version 4.1.0 vs requested ^2.28.0 || ^3.0.0
2. **Build timeout** o fallo durante `yarn build`

Solución: Usar Next Start



En lugar de usar standalone output, podemos usar el comando normal de Next.js:

```
yarn start # En lugar de node server.js
```

Ventajas

-  No depende de standalone output
-  Más simple y directo
-  Funciona con cualquier versión de Next.js
-  Menos propenso a errores de configuración

Desventajas

-  Imagen Docker más grande (incluye todos los node_modules)
-  Ligeramente más lento en cold start

Cómo Implementar

Opción 1: Usar Dockerfile.simple (Recomendado)

1. **Renombrar Dockerfile actual:**

```
bash
cd /home/ubuntu/muebleria_la_economica
```

```
mv Dockerfile Dockerfile.standalone.backup
mv Dockerfile.simple Dockerfile
```

2. Commit y push:

```
bash
git add -A
git commit -m "🔄 Usar next start en lugar de standalone"
git push origin main
```

3. Rebuild en EasyPanel

Opción 2: Modificar Dockerfile Actual

Reemplazar la sección de build con:

```
# Build Next.js - SIN verificar standalone
ENV NEXT_TELEMETRY_DISABLED=1
ENV NODE_ENV=production
RUN yarn build
```

Y en el runner stage, usar:

```
# Copy todo desde builder
COPY --from=builder --chown=nextjs:nodejs /app ./

USER nextjs
EXPOSE 3000
CMD ["yarn", "start"]
```




Comparación

Aspecto	Standalone	Next Start
Tamaño de imagen	~200MB	~500MB
Complejidad	Alta	Baja
Confiabilidad	Depende de config	Muy alta
Cold start	Más rápido	Normal
Mantenimiento	Complejo	Simple

Recomendación

Usar `next start` (**Dockerfile.simple**) porque:

1.  Funciona garantizado

2.  Menos configuración
3.  Más fácil de debuggear
4.  Menos dependencias de versiones

El tamaño extra de imagen (~300MB) es aceptable para tener un deploy que funciona de forma confiable.



Para el Futuro

Una vez que la aplicación esté funcionando en producción, podemos:

1. Investigar el problema de las peer dependencies
2. Actualizar las versiones conflictivas
3. Intentar standalone nuevamente si lo deseas

Pero primero, **prioridad #1: tener la aplicación funcionando** 



Pasos Inmediatos

1. Usar Dockerfile.simple
2. Push a GitHub
3. Rebuild en EasyPanel
4. ¡Aplicación funcionando! 🎉

Luego optimizamos si es necesario.

Pragmatismo > Perfección 🚀