



ACCIÓN REQUERIDA: Completar Deployment



Lo que YA está COMPLETO

Cambios en el Repositorio

1. **PR #60 mergeado** - Logging verbose para debugging
2. **PR #61 mergeado** - Fix de configuración standalone
3. **Documentación actualizada** - DEPLOYMENT_COMPLETION_SUMMARY.md
4. **Todos los archivos críticos verificados** - seed.ts, schema.prisma, etc.

Estado del Código

- Branch `main` actualizado con todos los fixes
 - `next.config.js` con fallback standalone garantizado
 - `build-with-standalone.sh` simplificado y con logging verbose
 - `Dockerfile` con `BUILD_TIMESTAMP` actualizado
 - Configuración de Prisma correcta
-



PROBLEMA: Webhook No Disponible

El webhook de Easypanel proporcionado **NO responde**:

```
URL: https://easypanel.abacus.ai/webhooks/deploy/clzwwqxqy0001uo8aqvvvvvvv
Error: Could not resolve host: easypanel.abacus.ai
```

Posibles causas:

1. El dominio `easypanel.abacus.ai` no existe o no es accesible
 2. El webhook URL es incorrecto
 3. El servicio de Easypanel está en un dominio diferente
 4. Problemas de red/DNS
-




ACCIÓN REQUERIDA DEL USUARIO

Opción 1: Auto-Deploy (Más Probable) ★

Si configuraste auto-deploy en Easypanel:

1. **Verifica en Easypanel Dashboard:**
 - Accede a tu panel de Easypanel
 - Ve al proyecto CitaPlanner
 - Revisa la sección "Deployments" o "Activity"
 - Busca un deployment automático iniciado hace unos minutos

2. Si ves un deployment en progreso:

-  ¡Perfecto! El auto-deploy está funcionando
- Monitorea los logs del build
- Espera a que complete

3. Si NO ves ningún deployment:

- Continúa con la Opción 2

Opción 2: Deploy Manual

1. Accede a Easypanel Dashboard

- URL de tu Easypanel (no easypanel.abacus.ai)



2. Navega a tu proyecto:

- Busca "CitaPlanner" o "citaplanner"
- Entra al proyecto

3. Inicia el Deploy:

- Busca el botón "Deploy" o "Redeploy"
- Haz clic para iniciar el deployment
- Selecciona branch `main` si te lo pregunta

4. Monitorea los Logs:

- Abre la consola de logs
- Verás el output verbose del build
- Busca mensajes con  (éxito) o  (error)

Opción 3: Verificar Webhook Correcto

1. En Easypanel Dashboard:

- Ve a Settings o Configuration
- Busca "Webhooks" o "Deploy Hooks"
- Copia el webhook URL correcto

2. Ejecuta el webhook manualmente:

```
bash
```

```
curl -X POST [TU_WEBHOOK_URL_CORRECTO]
```



Qué Esperar Durante el Build

Con el logging verbose implementado, verás:

✓ Mensajes de Éxito

```
=====
🚀 Building Next.js app with standalone output...
=====
✓ Changed to /app directory
✓ Essential files verified
✓ node_modules exists
🔧 Starting Next.js build...
✓ yarn build completed successfully
✓ Standalone build successful!
✓ server.js found in standalone directory!
🎉 Build completed successfully with standalone output!
```

✗ Si Hay Errores

Los logs mostrarán exactamente:

- Qué archivo falta
- Qué comando falló
- El error específico
- El estado de los directorios

🔍 Verificación Post-Deployment

Una vez que el deployment complete:

1. Verificar la Aplicación

```
# Accede a tu URL de CitaPlanner
https://[tu-dominio-easypanel]
```

2. Verificar Logs del Container

En Easypanel:

- Ve a “Logs” o “Console”
- Busca mensajes de inicio:

```
🚀 Iniciando CITAPLANNER...
✓ Prisma CLI encontrado
🌱 Verificando si necesita seed...
🚀 EJECUTANDO: node server.js
```

3. Verificar Base de Datos

Si es la primera vez:

- El seed debería ejecutarse automáticamente
- Verifica que hay usuarios en la base de datos

Si Encuentras Errores

Error: “server.js NOT FOUND”

Causa: El build standalone no se generó correctamente

Solución:

1. Verifica que `NEXT_OUTPUT_MODE=standalone` está configurado
2. Revisa los logs del build para ver dónde falló
3. El fallback en next.config.js debería prevenir esto

Error: “Prisma CLI not found”

Causa: Dependencias de Prisma no copiadas correctamente

Solución:

1. Verifica que el Dockerfile copió `node_modules/@prisma`
2. El Dockerfile actual ya tiene esto configurado

Error: “Cannot connect to database”

Causa: Variable de entorno DATABASE_URL incorrecta

Solución:

1. Verifica DATABASE_URL en Easypanel environment variables
2. Asegúrate de que la base de datos está accesible

Información de Contacto

Si necesitas ayuda adicional:

1. Revisa la documentación:

- `DEPLOYMENT_COMPLETION_SUMMARY.md` - Resumen completo
- `EASYPANEL-DEPLOYMENT-GUIDE.md` - Guía de Easypanel

2. Logs Verbose:




- Los logs ahora son muy detallados
- Copia el error exacto si necesitas ayuda

3. GitHub:

- Todos los cambios están en: <https://github.com/qhosting/citaplanner>
- PRs mergeados: #60 y #61



Resumen de Mejoras Implementadas

Robustez




-  Fallback garantizado para standalone output
-  Verificaciones de archivos esenciales
-  Manejo de errores mejorado

Debugging

-  Logging verbose en cada paso

-  Mensajes de error claros y descriptivos
-  Verificaciones de estado detalladas

Simplicidad



-  Código más limpio y mantenible
-  Menos manipulación de archivos
-  Configuración centralizada



Conclusión

Todo el código está listo y optimizado para deployment.

Lo único que falta es **iniciar el deployment desde Easypanel**, ya sea:

-  Automáticamente (si auto-deploy está configurado)
-  Manualmente (haciendo clic en “Deploy”)

Una vez iniciado, el build debería completarse exitosamente gracias a todas las mejoras implementadas.

¡Buena suerte con el deployment! 

Última actualización: 7 de Octubre, 2025

Commits en main: 0f360c3 (docs: add deployment completion summary)

PRs mergeados: #60, #61