


Checklist de Deploy - EasyPanel








Antes del Deploy

Git y GitHub

- ☒ Código actualizado localmente
- ☒ Dockerfile optimizado (v9.0)
- ☒ package-lock.json generado
- ☒ Commit realizado
- ☐ **Push a GitHub**  SIGUIENTE PASO

```
# Ejecuta esto con tu token:  
cd /home/ubuntu/escalafin_mvp  
git remote set-url origin https://TU_TOKEN@github.com/qhosting/escalafin-mvp.git  
git push origin main
```

Verificar Archivos

- ☒ Dockerfile 
- ☒ Dockerfile.easypanel 
- ☒ start-easypanel.sh 
- ☒ .dockerignore 
- ☒ app/package-lock.json 
- ☒ app/next.config.js (con standalone) 
- ☒ app/prisma/schema.prisma 

En EasyPanel

1. Crear/Configurar Base de Datos

- ☐ PostgreSQL creado
- ☐ Nombre: `escalafin-db`
- ☐ Usuario y password guardados
- ☐ DATABASE_URL copiada

2. Crear Aplicación

- ☐ App creada desde GitHub
- ☐ Repository: `qhosting/escalafin-mvp`
- ☐ Branch: `main`
- ☐ Build Method: `Dockerfile`
- ☐ Dockerfile: `Dockerfile` (o `Dockerfile.easypanel`)
- ☐ Port: `3000`

3. Variables de Entorno

Obligatorias ⚠

- [] `DATABASE_URL` ← De la base de datos creada
- [] `NEXTAUTH_URL` ← Tu dominio (ej: `https://app.escalafin.com`)
- [] `NEXTAUTH_SECRET` ← Genera con: `openssl rand -base64 32`
- [] `NODE_ENV=production`
- [] `NEXT_TELEMETRY_DISABLED=1`

AWS S3 (Cloud Storage)

- [] `AWS_BUCKET_NAME`
- [] `AWS_FOLDER_PREFIX`
- [] `AWS_REGION`
- [] `AWS_ACCESS_KEY_ID` (si no está en servidor)
- [] `AWS_SECRET_ACCESS_KEY` (si no está en servidor)

Openpay (Pagos)

- [] `OPENPAY_MERCHANT_ID`
- [] `OPENPAY_PRIVATE_KEY`
- [] `OPENPAY_PUBLIC_KEY`
- [] `OPENPAY_BASE_URL`

Evolution API (WhatsApp)

- [] `EVOLUTION_API_URL`
- [] `EVOLUTION_API_TOKEN`
- [] `EVOLUTION_INSTANCE_NAME`

4. Dominio

- [] Dominio agregado (ej: `app.escalafin.com`)
- [] DNS configurado (A record apuntando al servidor)
- [] SSL generado automáticamente

5. Deploy

- [] Click en “Deploy” o “Build”
- [] Esperar build (5-8 minutos primera vez)
- [] Revisar logs para errores



Verificación Post-Deploy

1. Logs

Buscar estos mensajes:

- ✓ Build de `Next.js` completado exitosamente
- ✓ Prisma Client generado
- ✓ Server started on port 3000

2. Health Check

```
curl https://tu-dominio.com/api/health
```

Debe retornar:

```
{"status": "ok"}
```

3. Aplicación

- [] Página de inicio carga correctamente
- [] Login funciona
- [] Dashboard accesible
- [] Sin errores en consola del navegador

4. Base de Datos

- [] Conexión exitosa
- [] Tablas creadas (migraciones ejecutadas)



Si Algo Falla

Build Errors

1. Revisar logs en EasyPanel
2. Verificar que el Dockerfile sea correcto
3. Verificar que package-lock.json exista
4. Forzar rebuild limpio

Database Errors

1. Verificar DATABASE_URL
2. Verificar que el hostname sea el interno (no localhost)
3. Verificar que la DB esté running

500/502 Errors

1. Revisar variables de entorno
2. Revisar logs de la aplicación
3. Verificar que el puerto sea 3000
4. Verificar health check

SSL Errors

1. Verificar DNS (puede tardar hasta 48h)
2. Esperar unos minutos
3. Refresh certificate en EasyPanel



Documentación de Ayuda

Si necesitas más detalles:

1. **INSTRUCCIONES_EASYPANEL.md** - Resumen rápido

2. **EASYPANEL_DEPLOY_GUIDE.md** - Guía completa

3. **RESUMEN_CAMBIOS_v9.0.md** - Detalles técnicos

Resultado Final

Al completar todos los checks:

- ✓ Aplicación desplegada en EasyPanel
 - ✓ Accesible vía HTTPS
 - ✓ Base de datos conectada
 - ✓ Todas las funcionalidades operativas
 - ✓ Listo para producción
-

Versión: 9.0

Fecha: 2025-10-15

Próximo paso: Push a GitHub y configurar en EasyPanel