

Guía de Deployment en EasyPanel - CITAPLANNER

Esta guía te llevará paso a paso para desplegar tu aplicación CitaPlanner en **EasyPanel**.

Requisitos Previos

- ✓ EasyPanel instalado y funcionando
- ✓ Acceso a la interfaz web de EasyPanel
- ✓ Repositorio de GitHub: <https://github.com/qhosting/citaplanner>

PASO 1: Crear un Nuevo Proyecto en EasyPanel

1. Accede a la interfaz de EasyPanel
2. Haz clic en **“Create Project”**
3. Nombre del proyecto: `citaplanner`
4. Haz clic en **“Create”**

PASO 2: Crear la Base de Datos PostgreSQL

Opción A: Desde la Interfaz de EasyPanel

1. Dentro de tu proyecto, haz clic en **“Add Service”**
2. Selecciona **“Database”** → **“PostgreSQL”**
3. Configuración:
 - Service Name: `citaplanner-postgres`
 - Database Version: 17 (o la más reciente)
 - Database Name: `citaplanner`
 - Username: `postgres`
 - Password: [Genera una contraseña segura]
4. Haz clic en **“Create”**
5. Espera a que la base de datos esté **“Running”** (verde)

Opción B: Agregar PostgreSQL desde la Biblioteca de Servicios

1. En tu proyecto, ve a **“Services”**
2. Haz clic en **“Add from Template”**
3. Busca **“PostgreSQL”**
4. Selecciona la versión 17
5. Completa los datos:
 - **Service Name:** `citaplanner-postgres`

- **Database Name:** citaplanner
- **Root Password:** Una contraseña segura

6. Haz clic en **“Deploy”**



Guarda la Información de la Base de Datos

Copia estos datos, los necesitarás en el siguiente paso:

```
Host: citaplanner-postgres (nombre del servicio)
Port: 5432
Database: citaplanner
User: postgres
Password: [tu contraseña]
```

La **DATABASE_URL** será:

```
postgresql://postgres:[TU_PASSWORD]@citaplanner-postgres:5432/citaplanner
```



PASO 3: Desplegar la Aplicación Next.js

3.1. Agregar el Servicio de la Aplicación

1. En tu proyecto, haz clic en **“Add Service”**
2. Selecciona **“App”** → **“From GitHub”**
3. Conecta tu repositorio de GitHub
4. Selecciona el repositorio: citaplanner
5. Branch: main o deploy-easypanel

3.2. Configuración de Build

```
Build Method: Dockerfile
Dockerfile Path: ./Dockerfile
Build Context: .
```

3.3. Configuración del Contenedor

```
Service Name: citaplanner-app
Port: 3000
Healthcheck Path: /api/health
```



PASO 4: Configurar Variables de Entorno

En la sección **“Environment Variables”** de tu servicio de la aplicación, agrega:

Variables Obligatorias

```
NODE_ENV=production
PORT=3000

# Database
DATABASE_URL=postgresql://postgres:[TU_PASSWORD]@citaplanner-postgres:5432/citaplanner

# NextAuth
NEXTAUTH_URL=https://[TU_DOMINIO_EASYPANEL]
NEXTAUTH_SECRET=[Genera uno nuevo con: openssl rand -base64 32]

# JWT (si se usa en tu aplicación)
JWT_SECRET=[Genera uno nuevo con: openssl rand -base64 32]
```

Cómo Generar Secrets Seguros

En tu terminal local o SSH:

```
# Para NEXTAUTH_SECRET
openssl rand -base64 32

# Para JWT_SECRET
openssl rand -base64 32
```

Copia los valores generados y pégalos en las variables correspondientes.

PASO 5: Configurar el Dominio

5.1. En EasyPanel

1. En tu servicio de la aplicación, ve a **“Domains”**
2. Haz clic en **“Add Domain”**
3. Ingresa tu dominio: `app.citaplanner.com`
4. Activa **“Enable SSL/TLS”** (Let’s Encrypt automático)
5. Haz clic en **“Save”**

5.2. En tu Proveedor de Dominio (DNS)




Agrega un registro A o CNAME apuntando a tu servidor EasyPanel:

```
Type: A
Name: app
Value: [IP de tu servidor EasyPanel]
TTL: 3600
```

O si EasyPanel te proporciona un dominio:

```
Type: CNAME
Name: app
Value: [dominio-proporcionado-por-easypanel]
TTL: 3600
```

PASO 6: Deploy

1. Revisa que todos los servicios estén configurados:
 -  PostgreSQL: Running
 -  App: Configurada con variables de entorno
 -  Dominio: Configurado
 2. Haz clic en **“Deploy”** en tu servicio de la aplicación
 3. Monitorea los logs en tiempo real para verificar el deployment
-

PASO 7: Inicializar la Base de Datos (Primera vez)

Una vez que la aplicación esté corriendo:

Opción A: Desde la Terminal de EasyPanel

1. Ve a tu servicio de la aplicación
2. Haz clic en **“Terminal”** o **“Console”**
3. Ejecuta los comandos Prisma:

```
npx prisma generate
npx prisma migrate deploy
npx prisma db seed
```

Opción B: Desde SSH (si tienes acceso)

```
# Encuentra el contenedor
docker ps | grep citaplanner

# Accede al contenedor
docker exec -it [CONTAINER_ID] sh

# Ejecuta los comandos
cd /app
npx prisma generate
npx prisma migrate deploy
npx prisma db seed
```

PASO 8: Verificación

8.1. Verifica el Health Check

Accede a:

```
https://app.citaplanner.com/api/health
```

Deberías ver una respuesta exitosa del servidor.

8.2. Accede a la Aplicación

Abre en tu navegador:

```
https://app.citaplanner.com
```

Deberías ver la página principal de CitaPlanner.

8.3. Credenciales de Prueba (después del seed)

Verifica en tu archivo de seed (`app/scripts/seed.ts`) las credenciales de prueba configuradas.



PASO 9: Configurar Auto-Deploy (Opcional)

Para que EasyPanel despliegue automáticamente cuando hagas push a GitHub:

1. Ve a tu servicio de la aplicación
2. En **“Settings”**, busca **“Auto Deploy”**
3. Activa **“Enable Auto Deploy”**
4. Selecciona el branch: `main` o `deploy-easypanel`
5. Guarda los cambios

Ahora cada push a la rama seleccionada desplegará automáticamente.



Monitoreo y Logs

Ver Logs en Tiempo Real

1. Ve a tu servicio en EasyPanel
2. Haz clic en **“Logs”**
3. Selecciona **“Live Logs”**

Métricas de Rendimiento

1. Ve a **“Monitoring”**
2. Revisa:
 - CPU Usage
 - Memory Usage
 - Network Traffic
 - Response Time



Troubleshooting

Problema: La app no inicia

Verifica:

1. Los logs en EasyPanel → **“Logs”**
2. Que la `DATABASE_URL` sea correcta

3. Que PostgreSQL esté corriendo
4. Las variables de entorno estén completas

Solución:

```
# Reinicia el servicio
En EasyPanel: Click en "Restart"
```

Problema: Error de conexión a la base de datos**Verifica:**

1. PostgreSQL está corriendo
2. La DATABASE_URL usa el nombre del servicio (citaplanner-postgres)
3. El password sea correcto

Solución:

```
# Conéctate a PostgreSQL para verificar
docker exec -it citaplanner-postgres-easypanel psql -U postgres -d citaplanner
```

Problema: El dominio no resuelve**Verifica:**

1. Los registros DNS estén propagados (usa <https://dnschecker.org>)
2. El dominio esté agregado en EasyPanel
3. SSL esté activado

Solución:

```
# Espera 5-10 minutos para propagación DNS
# Luego fuerza la renovación del certificado en EasyPanel
```

Problema: Error de Prisma Client**Verifica:**

1. Que el cliente Prisma se haya generado correctamente
2. Los logs del contenedor para ver errores específicos

Solución:

```
# Accede al contenedor y regenera el cliente
docker exec -it [CONTAINER_ID] sh
cd /app
npx prisma generate
```

**Recursos Adicionales**





- [Documentación oficial de EasyPanel](https://easypanel.io/docs) (<https://easypanel.io/docs>)
- [Next.js Deployment](https://nextjs.org/docs/deployment) (<https://nextjs.org/docs/deployment>)
- [Prisma Deployment](https://www.prisma.io/docs/guides/deployment) (<https://www.prisma.io/docs/guides/deployment>)

- [Repositorio CitaPlanner](https://github.com/qhosting/citaplanner) (<https://github.com/qhosting/citaplanner>)
-









Tu aplicación **CITAPLANNER** ahora está desplegada en EasyPanel.

URLs Importantes

-  **Aplicación:** <https://app.citaplanner.com>
-  **Health Check:** <https://app.citaplanner.com/api/health>
-  **Panel de EasyPanel:** [Tu URL de EasyPanel]
-  **Repositorio:** <https://github.com/qhosting/citaplanner>

Próximos Pasos

1.  Cambia las credenciales de administrador por defecto
 2.  Configura backups automáticos de la base de datos
 3.  Configura monitoreo y alertas
 4.  Prueba todas las funcionalidades de la aplicación
 5.  Configura notificaciones (SMS, WhatsApp, Email)
 6.  Personaliza el branding de tu negocio
-

Última actualización: 1 de octubre, 2025

Versión: 1.0.0

Proyecto: CitaPlanner - Sistema de Gestión de Citas