



Guía de Configuración para Producción en Vercel



Problemas Identificados

Tu deployment en Vercel está fallando porque **faltan variables de entorno críticas**:

1. `DATABASE_URL` - Configurada correctamente
2. `DIRECT_URL` - Configurada correctamente
3. `NEXTAUTH_SECRET` - **VACÍA** (causa errores de autenticación)
4. `NEXT_PUBLIC_BASE_URL` - **VACÍA** (causa problemas de URLs)
5. `EMAIL_USER` - Configurada
6. `EMAIL_PASSWORD` - Configurada
7. `EMAIL_HOST` - Configurada
8. `EMAIL_PORT` - Configurada



Solución Paso a Paso

1 Generar NEXTAUTH_SECRET

Necesitas un string aleatorio de al menos 32 caracteres. Usa uno de estos métodos:

Opción A: Usando OpenSSL (en tu terminal local)

```
openssl rand -base64 32
```

Opción B: Usando Node.js

```
node -e "console.log(require('crypto').randomBytes(32).toString('base64'))"
```

Opción C: Generador online

Visita: <https://generate-secret.vercel.app/32>

Ejemplo de resultado:

```
Kx8vZ2mP9qR3sT5wY7nB1cD4fG6hJ8kL0mN2pQ4rS6tU8vW0xY2zA4bC6dE8fG0h
```

2 Obtener tu URL de Vercel

Tu URL de producción será algo como:

```
https://derlybarber.vercel.app
```

```
https://tu-dominio-personalizado.com
```

Para encontrarla:

1. Ve a tu proyecto en Vercel
2. En la pestaña “Deployments”, copia la URL del deployment activo
3. O ve a “Settings” > “Domains” para ver tu dominio

3 Configurar Variables en Vercel

1. **Ve a tu proyecto en Vercel:** <https://vercel.com/dashboard>
2. **Selecciona tu proyecto** “derlybarber-“
3. **Ve a Settings > Environment Variables**
4. **Agrega/Actualiza estas variables:**

Variable	Valor	Entorno
NEXTAUTH_SECRET	[el secret que generaste en paso 1]	Production, Preview, Development
NEXT_PUBLIC_BASE_URL	https://tu-dominio.vercel.app	Production, Preview, Development

IMPORTANTE:

- Marca las 3 opciones de entorno: Production, Preview, Development
- Haz clic en “Save” después de cada variable

4 Verificar Migraciones de Prisma

El archivo `vercel.json` ya está configurado correctamente con:

```
{
  "buildCommand": "prisma generate && next build"
}
```

Esto asegura que Prisma genere el cliente antes del build.

PERO, si las tablas no existen en tu base de datos de producción, necesitas ejecutar las migraciones:

Opción A: Desde tu terminal local (RECOMENDADO)

```
# 1. Asegúrate de tener las variables de entorno correctas
export DATABASE_URL="postgresql://postgres:Qu!t@s0l2025*@db.qjlydtbbragytngrjvd.supabase.co:5432/postgres"

# 2. Ejecuta las migraciones
npx prisma db push

# 3. Verifica que las tablas se crearon
npx prisma studio
```

Opción B: Agregar script de build en Vercel

Actualiza `vercel.json` :

```
{
  "buildCommand": "prisma generate && prisma db push --accept-data-loss && next build",
  "framework": "nextjs",
  "installCommand": "npm install --legacy-peer-deps && npm install -g prisma",
  "regions": ["iad1"]
}
```

⚠ **ADVERTENCIA:** `--accept-data-loss` puede borrar datos. Úsalo solo si la base de datos está vacía.

5 Re-deployar en Vercel

Después de configurar las variables:

1. **Ve a la pestaña “Deployments”**
2. **Haz clic en los 3 puntos** del último deployment
3. **Selecciona “Redeploy”**
4. **Marca “Use existing Build Cache”** (opcional, más rápido)
5. **Haz clic en “Redeploy”**

O simplemente haz un nuevo commit y push:

```
git add .
git commit -m "Fix: Add missing environment variables"
git push origin main
```

6 Verificar que Funciona

Una vez re-deployado:

1. **Prueba el formulario de contacto:** <https://tu-dominio.vercel.app/contact>
2. **Prueba el sistema de citas:** <https://tu-dominio.vercel.app/booking>
3. **Revisa los logs en Vercel:**
 - Ve a tu proyecto > “Deployments” > Click en el deployment > “Functions”
 - Busca errores en los logs de las funciones `/api/appointments` y `/api/contact`

Diagnóstico de Problemas

Si el formulario de contacto no funciona:

1. **Verifica los logs en Vercel:**
 - Ve a Functions > `/api/contact`
 - Busca errores relacionados con email o base de datos
2. **Verifica la configuración de email:**
 - Asegúrate de que `EMAIL_PASSWORD` sea una “App Password” de Gmail
 - Verifica que `EMAIL_USER` sea correcto
3. **Prueba la conexión a la base de datos:**
 - Los logs mostrarán si hay problemas de conexión
 - Verifica que `DATABASE_URL` sea correcta

Si las citas no funcionan:

1. Verifica que las tablas existan:

```
bash
```

```
npx prisma studio
```

Deberías ver las tablas: `appointments`, `contact_submissions`, etc.

2. Revisa los logs de la función:

- Ve a Functions > `/api/appointments`

- Busca errores de Prisma



Checklist Final

- [] `NEXTAUTH_SECRET` configurado en Vercel (mínimo 32 caracteres)
- [] `NEXT_PUBLIC_BASE_URL` configurado con tu URL de Vercel
- [] Variables de email configuradas correctamente
- [] Migraciones de Prisma ejecutadas (`npx prisma db push`)
- [] Re-deployment realizado en Vercel
- [] Formulario de contacto probado y funcionando
- [] Sistema de citas probado y funcionando
- [] Emails de confirmación llegando correctamente



Soporte Adicional

Si después de seguir estos pasos aún tienes problemas:

1. **Revisa los logs en Vercel** (muy importante)
2. **Verifica la consola del navegador** (F12) para errores de JavaScript
3. **Prueba en modo incógnito** para descartar problemas de caché
4. **Verifica que Supabase esté activo** y aceptando conexiones



Resumen de Variables Requeridas

```
# Base de datos
DATABASE_URL="postgresql://postgres:Qu!t@s0l2025*@db.qjlydtbbragytngrjvd.supabase.co:5432/postgres"
DIRECT_URL="https://qjlydtbbragytngrjvd.supabase.co"

# NextAuth (CRÍTICO)
NEXTAUTH_SECRET="[genera uno con: openssl rand -base64 32]"

# URL base (CRÍTICO)
NEXT_PUBLIC_BASE_URL="https://tu-dominio.vercel.app"

# Email
EMAIL_USER=minina.ia25@gmail.com
EMAIL_PASSWORD=hrdumlvswpigvvgc
EMAIL_HOST=smtp.gmail.com
EMAIL_PORT=587
```

¡Listo! Después de configurar estas variables y re-deployar, tu aplicación debería funcionar perfectamente en producción. 🎉