

Resultado del Test de Conexión a Neon



Fecha: 2 de octubre de 2025

Connection String probada:

```
postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/neondb?sslmode=require
```





RESPUESTA A TU PREGUNTA

SÍ, esta conexión sirve para AMBAS cosas:

1.  **Conectar la aplicación** (queries normales)
 - SELECT, INSERT, UPDATE, DELETE
 - Operaciones DML funcionan perfectamente
2.  **Ejecutar migraciones** (crear/modificar tablas)
 - CREATE TABLE, ALTER TABLE, DROP TABLE
 - Operaciones DDL funcionan perfectamente
 - Puede ejecutar las migraciones de Prisma

Resultados de las Pruebas





Test 1: Conexión Básica

-  Conexión exitosa
-  PostgreSQL: 15.13
-  Database: neondb
-  User: neondb_owner

Test 2: Tablas Existentes

Base de datos vacía (sin tablas todavía)

Test 3: Operaciones DDL

-  CREATE TABLE exitoso
-  INSERT exitoso
-  SELECT exitoso
-  DROP TABLE exitoso

⚠ PROBLEMA DETECTADO

Diferencia en el Nombre de la Base de Datos

Tu conexión actual usa:

```
Base de datos: neondb
```

Pero el proyecto espera:

```
Base de datos: hotel_pms_production
```

🔧 SOLUCIÓN: Dos Opciones

Opción 1: Usar “neondb” (Más Simple) RECOMENDADO

Simplemente usa la base de datos que ya tienes (“neondb”). Es más fácil y funciona igual.

Pasos:

1. Usar esta URL para **AMBAS** variables en Vercel:

DATABASE_URL:

```
postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/neondb?sslmode=require
```

DIRECT_DATABASE_URL:

```
postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/neondb?sslmode=require
```

💡 **NOTA:** Como probamos, esta conexión (aunque tenga `-pooler`) Sí soporta operaciones DDL, así que podemos usar la MISMA URL para ambas variables. No necesitas dos URLs diferentes.

1. Las otras variables en Vercel:

Variable	Valor
NEXTAUTH_URL	https://hotel-pms-paseo-las-mercedes.vercel.app
NEXTAUTH_SECRET	aG8t3lP@s30L@sM3rc3d3s! 2024#PMS\$V3rc3l%S3cr3t&K3y*S3cur3!
NODE_ENV	production

1. Re-deployar en Vercel

Opción 2: Crear Base de Datos “hotel_pms_production”

Si prefieres usar el nombre original del proyecto:

Pasos:

1. Ve a: <https://console.neon.tech>
2. Selecciona tu proyecto
3. Ve a la sección “**Databases**”
4. Haz clic en “**Create Database**”
5. Nombre: `hotel_pms_production`
6. Haz clic en “**Create**”
7. **Actualizar la URL:**
`postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/hotel_pms_production?sslmode=require`
 (Solo cambia `neondb` por `hotel_pms_production` al final)



DESCUBRIMIENTO IMPORTANTE

Neon Pooler Soporta DDL

A diferencia de otros proveedores, **Neon permite operaciones DDL incluso a través del pooler** (conexión con `-pooler`).

Esto significa:

- ☒ **NO necesitas dos URLs diferentes** para Neon
- ☒ Puedes usar la **misma URL** para `DATABASE_URL` y `DIRECT_DATABASE_URL`
- ☒ Simplifica la configuración

Otras bases de datos (como Supabase con pgbouncer) Sí requieren dos URLs, pero **Neon no**.



Configuración Simplificada para Vercel

Dado que la conexión soporta DDL, puedes usar esta configuración simple:

Environment Variables en Vercel:

```

DATABASE_URL=postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/neondb?sslmode=require

DIRECT_DATABASE_URL=postgresql://neondb_owner:npg_vYwoRk4e5sLQ@ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech/neondb?sslmode=require

NEXTAUTH_URL=https://hotel-pms-paseo-las-mercedes.vercel.app

NEXTAUTH_SECRET=aG8t3lP@s30L@S3M3rc3d3s!2024#PMS$V3rc3l%S3cr3t&K3y*S3cur3!

NODE_ENV=production
  
```

Nota: Las URLs son idénticas porque Neon pooler soporta DDL.

Consideración sobre schema.prisma

He actualizado anteriormente el `schema.prisma` para usar `directUrl`. Esto NO causará problemas, pero es técnicamente redundante con Neon ya que ambas URLs son iguales.

Opción A: Dejar como está (funciona perfectamente)







```
datasource db {
  provider = "postgresql"
  url      = env("DATABASE_URL")
  directUrl = env("DIRECT_DATABASE_URL")
}
```

Opción B: Simplificar (también funciona)

```
datasource db {
  provider = "postgresql"
  url      = env("DATABASE_URL")
}
```

Recomendación: Dejar Opción A por compatibilidad futura si migras a otra BD.

Próximos Pasos Inmediatos

1.  **La conexión funciona** (ya probada)
 2.  **Configurar las 5 variables en Vercel** (con la URL que te di)
 3.  **Re-deployar en Vercel**
 4.  **Verificar que el build sea exitoso**
 5.  **Ver las tablas creadas en Neon**
 6.  **Probar la aplicación**
-



Resumen Comparativo

Aspecto	Tu Conexión	Resultado
Hostname	ep-shy-tree-adruv4z1-pool-er...	✓ Funciona
Pooler	Sí (incluye -pooler)	✓ Soporta DDL de todos modos
Base de datos	neondb	⚠ Diferente del nombre esperado
Usuario	neondb_owner	✓ Correcto
Región	us-east-1	✓ Funciona
SSL	sslmode=require	✓ Correcto
Queries (DML)	-	✓ Funciona
Migraciones (DDL)	-	✓ Funciona



Conclusión Final

¡Excelente noticia! Tu conexión funciona perfectamente para todo lo que necesitas.

Lo que debes hacer:

1. Usar la misma URL para ambas variables (DATABASE_URL y DIRECT_DATABASE_URL)
2. Configurar las otras 3 variables en Vercel
3. Re-deployar
4. ¡Listo! 🎉

No necesitas hacer cambios complicados ni obtener otra URL diferente.



Diferencias con la Configuración Anterior

Antes:

```
Host: ep-cool-sea-a5jt7r2q-pooler.us-east-2.aws.neon.tech
DB: hotel_pms_production
Región: us-east-2
```

Ahora:

```
Host: ep-shy-tree-adruv4z1-pooler.c-2.us-east-1.aws.neon.tech  
DB: neondb  
Región: us-east-1
```

¿Es un problema? No. Son dos proyectos diferentes en Neon. Usa el que acabas de probar (el nuevo).

Última actualización: 2 de octubre de 2025 - 17:00

Estado: Conexión probada y validada 

Acción requerida: Configurar variables en Vercel