

# Instrucciones para Conectar a Git y Desplegar en Tu Servidor

## ✓ Estado Actual

Tu proyecto **SpeaklyPlan** está listo para ser desplegado. He realizado lo siguiente:

1. ✓ Inicializado repositorio Git
2. ✓ Creado archivo `.gitignore` apropiado
3. ✓ Exportado la base de datos actual
4. ✓ Creado documentación completa de despliegue
5. ✓ Realizado commit inicial

## Ubicación de Archivos

```
/home/ubuntu/speaklyplan/nextjs_space/
├── .git/                # Repositorio Git inicializado
├── database_export.json # Base de datos exportada (8.4 KB)
├── DEPLOYMENT_GUIDE.md  # Guía completa de despliegue
├── DEPLOYMENT_GUIDE.pdf # Guía en PDF
├── README_GIT.md        # README para el repositorio
├── .env.example          # Plantilla de variables de entorno
├── scripts/
│   ├── export-simple.js  # Script para exportar DB
│   └── import-db.js      # Script para importar DB
└── [resto de la aplicación]
```

## Paso 1: Conectar a GitHub/GitLab/Bitbucket

### Opción A: GitHub

#### 1. Crear repositorio en GitHub

- Ve a <https://github.com/new>
- Nombre: `speaklyplan` (o el que prefieras)
- Marca como privado
- NO inicialices con README, .gitignore o licencia

#### 2. Conectar tu proyecto local

```
``bash
cd /home/ubuntu/speaklyplan/nextjs_space
```

```
# Agregar el repositorio remoto (reemplaza TU_USUARIO)
git remote add origin git@github.com:TU_USUARIO/speaklyplan.git
# O si usas HTTPS:
# git remote add origin https://github.com/TU_USUARIO/speaklyplan.git

# Subir el código
git branch -M main
```

```
git push -u origin main
```

```

## Opción B: GitLab

### 1. Crear repositorio en GitLab

- Ve a <https://gitlab.com/projects/new>
- Nombre: `speaklyplan`
- Marca como privado
- Crea el proyecto

### 2. Conectar tu proyecto local

```
```bash
cd /home/ubuntu/speaklyplan/nextjs_space
```

```
# Agregar el repositorio remoto
```

```
git remote add origin git@gitlab.com:TU_USUARIO/speaklyplan.git
```

```
# Subir el código
```

```
git branch -M main
```

```
git push -u origin main
```

```
```
```

## Opción C: Bitbucket

### 1. Crear repositorio en Bitbucket

- Ve a <https://bitbucket.org/repo/create>
- Nombre: `speaklyplan`
- Marca como privado

### 2. Conectar tu proyecto local

```
```bash
cd /home/ubuntu/speaklyplan/nextjs_space
```

```
# Agregar el repositorio remoto
```

```
git remote add origin git@bitbucket.org:TU_USUARIO/speaklyplan.git
```

```
# Subir el código
```

```
git branch -M main
```

```
git push -u origin main
```

```
```
```



## Paso 2: Descargar Archivos para Tu Servidor

### Archivos Esenciales

Descarga estos archivos a tu computadora local:

1. **Base de datos exportada:** `database_export.json`
2. **Guía de despliegue:** `DEPLOYMENT_GUIDE.pdf`
3. **README:** `README_GIT.md`

## Desde tu servidor de desarrollo actual

```
# Comprimir archivos importantes
cd /home/ubuntu/speaklyplan/nextjs_space
tar -czf ../speaklyplan-export.tar.gz \
  database_export.json \
  DEPLOYMENT_GUIDE.md \
  DEPLOYMENT_GUIDE.pdf \
  README_GIT.md \
  .env.example \
  prisma/schema.prisma \
  scripts/

# El archivo estará en: /home/ubuntu/speaklyplan/speaklyplan-export.tar.gz
```

Descarga este archivo a tu computadora usando SFTP, SCP o el método que prefieras.



## Paso 3: Desplegar en Tu Servidor

### En tu servidor de producción:

#### 1. Clonar el repositorio

```
```bash
# Conectar por SSH a tu servidor
ssh usuario@tu-servidor.com

# Ir al directorio de aplicaciones
cd /var/www # o el directorio que prefieras

# Clonar
git clone git@github.com:TU_USUARIO/speaklyplan.git
cd speaklyplan
```
```

#### 1. Seguir la Guía de Despliegue

La guía completa está en `DEPLOYMENT_GUIDE.md` o `DEPLOYMENT_GUIDE.pdf`

### Resumen de pasos críticos:

```
```bash
# Instalar dependencias
yarn install

# Configurar .env (copiar de .env.example y llenar con tus datos)
cp .env.example .env
nano .env # o vim .env

# Configurar base de datos
yarn prisma generate
yarn prisma migrate deploy

# Importar datos (si tienes database_export.json)
# Copiar database_export.json a la carpeta del proyecto
node scripts/import-db.js
```

```
# O poblar con datos de ejemplo
yarn tsx scripts/seed.ts

# Construir
yarn build

# Iniciar con PM2
pm2 start yarn --name "speaklyplan" -- start
pm2 save
pm2 startup
...
```



## Paso 4: Configurar Variables de Entorno

**IMPORTANTE:** Debes configurar estas variables en tu servidor de producción:

```
# Base de Datos (PostgreSQL en tu servidor)
DATABASE_URL="postgresql://usuario:contraseña@localhost:5432/speaklyplan"

# NextAuth (tu dominio de producción)
NEXTAUTH_URL="https://tudominio.com"
NEXTAUTH_SECRET="genera_uno_nuevo_con_openssl_rand_-base64_32"

# AWS S3 (puedes usar el mismo bucket o crear uno nuevo)
AWS_ACCESS_KEY_ID="tu_key"
AWS_SECRET_ACCESS_KEY="tu_secret"
AWS_REGION="us-east-1"
AWS_BUCKET_NAME="tu-bucket"
AWS_FOLDER_PREFIX="speaklyplan/"

# Abacus AI
ABACUSAI_API_KEY="tu_api_key"
```



## Comandos Útiles Post-Despliegue

```
# Ver logs
pm2 logs speaklyplan

# Reiniciar
pm2 restart speaklyplan

# Ver estado
pm2 status

# Detener
pm2 stop speaklyplan

# Actualizar la aplicación
cd /var/www/speaklyplan
git pull
yarn install
yarn build
pm2 restart speaklyplan

# Backup de base de datos
node scripts/export-simple.js
```

## Checklist de Despliegue

---

- [ ] Repositorio Git creado y código subido
- [ ] Servidor con Node.js 18+ instalado
- [ ] PostgreSQL instalado y base de datos creada
- [ ] Variables de entorno configuradas ( `.env` )
- [ ] AWS S3 configurado con credenciales correctas
- [ ] Dependencias instaladas ( `yarn install` )
- [ ] Cliente Prisma generado ( `yarn prisma generate` )
- [ ] Migraciones ejecutadas ( `yarn prisma migrate deploy` )
- [ ] Datos importados o sembrados
- [ ] Aplicación construida ( `yarn build` )
- [ ] PM2 configurado y aplicación iniciada
- [ ] Nginx configurado (opcional pero recomendado)
- [ ] SSL configurado con Let's Encrypt (opcional pero recomendado)
- [ ] Usuario admin creado ( `yarn tsx scripts/make-admin.ts` )

## Soporte

---

### Documentos de Referencia

1. **DEPLOYMENT\_GUIDE.md**: Guía completa paso a paso
2. **README\_GIT.md**: Información general del proyecto
3. **ARQUITECTURA\_TECNICA.md**: Detalles técnicos de la arquitectura

### Problemas Comunes

#### Error de conexión a PostgreSQL

- Verificar que PostgreSQL esté corriendo
- Verificar `DATABASE_URL` en `.env`
- Verificar permisos del usuario

#### Error con AWS S3

- Verificar credenciales
- Verificar permisos del bucket
- Verificar que el bucket existe

#### La aplicación no inicia

- Ver logs: `pm2 logs speaklyplan`
- Verificar puerto 3000 disponible
- Verificar todas las variables de entorno

## ¡Listo!

---

Una vez completados todos los pasos, tu aplicación estará corriendo en producción.

Accede a: `https://tudominio.com` (o `http://ip-servidor:3000` si no configuraste dominio)

#### Credenciales de admin:

- Email: `admin@speakeyplan.com`
- Password: (el que configuraste con el script `make-admin`)

---

**¿Necesitas ayuda?** Consulta la documentación o contacta al equipo de desarrollo.