

Checklist Pre-Deployment para Vercel

Seguridad y Credenciales

- [] **NEXTAUTH_SECRET** generado con `openssl rand -base64 32`
- [] **DATABASE_URL** incluye `?sslmode=require` al final
- [] **AWS credentials** tienen los permisos correctos (`s3:PutObject`, `s3:GetObject`, `s3:DeleteObject`)
- [] **Stripe webhook secret** configurado (si usas Stripe)
- [] **Todas las variables secretas** están en Vercel Environment Variables, NO en el código

Base de Datos

- [] **Base de datos PostgreSQL** creada y accesible
 - [] **Connection string** testeada localmente
 - [] **SSL/TLS habilitado** en la conexión de base de datos
 - [] **Backup strategy** definida
 - [] **Migraciones de Prisma** listas para ejecutar:
- ```
bash
yarn prisma db push
```

## Archivos y Configuración

- [ ] **vercel.json** existe en el directorio raíz
- [ ] **.gitignore** configurado correctamente (excluye `.env`, `node_modules`, `.next`)
- [ ] **.env.example** actualizado con todas las variables necesarias
- [ ] **package.json** tiene los scripts correctos:

```
json
{
 "build": "next build",
 "start": "next start"
}
```

## Dominio y DNS

- [ ] **Dominio registrado** (`innova.app`)
- [ ] **Acceso a configuración DNS** disponible
- [ ] **Certificado SSL** se configurará automáticamente por Vercel
- [ ] **NEXTAUTH\_URL** apunta al dominio de producción

## Repositorio Git

- [ ] **Repositorio creado** en GitHub/GitLab/Bitbucket
- [ ] **Código pusheado** a la rama principal (`main`)
- [ ] **No hay archivos .env** en el repositorio
- [ ] **Build local exitoso** antes de pushear

## Servicios Externos

---

### AWS S3 (OBLIGATORIO)

- [ ] **Bucket creado** con el nombre correcto
- [ ] **CORS configurado** para permitir uploads desde el dominio
- [ ] **IAM user** creado con permisos mínimos necesarios
- [ ] **Access Key y Secret** guardados de forma segura

### Stripe (si aplica)

- [ ] **Cuenta de Stripe** configurada
- [ ] **API keys** (test y production) obtenidas
- [ ] **Webhook endpoint** configurado: `https://tu-dominio/api/stripe/webhook`
- [ ] **Eventos del webhook** seleccionados correctamente

### Email (opcional pero recomendado)

- [ ] **Proveedor de email** elegido (SMTP, SendGrid, etc.)
- [ ] **Credenciales** configuradas
- [ ] **Email de remitente** verificado

## Monitoreo y Logs

---

- [ ] **Vercel Analytics** habilitado en el proyecto
- [ ] **Error tracking** configurado (Sentry, LogRocket, etc.) - opcional
- [ ] **Uptime monitoring** configurado (UptimeRobot, etc.) - opcional

## Testing

---

- [ ] **Build local** completado sin errores:  

```
bash
cd nextjs_space && yarn build
```
- [ ] **Prisma generate** ejecutado sin errores:  

```
bash
yarn prisma generate
```
- [ ] **TypeScript** compila sin errores:  

```
bash
yarn tsc --noEmit
```
- [ ] **Env variables** validadas localmente

## Post-Deployment

---

- [ ] **Migraciones ejecutadas** después del primer deploy:  

```
bash
vercel env pull .env.local
cd nextjs_space
yarn prisma db push
yarn tsx scripts/create-super-admin.ts
```
- [ ] **Super admin creado** y credenciales probadas
- [ ] **Login funcional** en producción

- [ ] **Upload de archivos** testeado (verifica S3)
- [ ] **Pagos testeados** (si usas Stripe)
- [ ] **Emails enviados** correctamente (si configurado)
- [ ] **Dominio personalizado** funcionando (si configurado)
- [ ] **SSL activo** y funcionando
- [ ] **Analytics funcionando** (si configurado)

## Documentación

- [ ] **Credenciales documentadas** en gestor de contraseñas seguro
- [ ] **Variables de entorno** documentadas en DEPLOYMENT\_VERCEL.md
- [ ] **Procedure de rollback** definido
- [ ] **Contactos de soporte** de proveedores guardados

## Importante

### Antes del primer deployment:

```
1. Testear build local
cd nextjs_space
yarn install
yarn build

2. Verificar Prisma
yarn prisma generate

3. Testear variables de entorno
cp .env.example .env.local
Editar .env.local con valores reales
yarn dev
Probar login y funcionalidades
```

### Después del primer deployment:

```
1. Bajar env de Vercel
vercel env pull .env.local

2. Migrar base de datos
cd nextjs_space
yarn prisma db push

3. Crear super admin
yarn tsx scripts/create-super-admin.ts

4. Seed data (opcional)
yarn prisma db seed
```

## Red Flags

### NO DEPLOYAR SI:

- Build local falla
- TypeScript tiene errores

- Variables de entorno faltan
- Base de datos no es accesible
- AWS S3 no está configurado
- NEXTAUTH\_SECRET no está generado

## Listo para Deploy

---

Cuando todos los checks estén completos:

1. Haz commit final:

```
bash
git add .
git commit -m "Ready for Vercel deployment"
git push origin main
```

2. Ve a [vercel.com/new](https://vercel.com/new) (<https://vercel.com/new>) e importa tu proyecto

3. Sigue la [Guía de Deployment Completa](#) ([./DEPLOYMENT\\_VERCEL.md](#))
- 

### ¿Dudas? Consulta:

- [Guía Completa de Deployment](#) ([./DEPLOYMENT\\_VERCEL.md](#))
- [Quick Start \(5 minutos\)](#) ([./QUICK\\_START\\_VERCEL.md](#))
- [Variables de Entorno](#) ([./.env.example](#))