

# GUÍA DE DEPLOYMENT EN VPS (CONTROL TOTAL)

**Fecha:** 13 de Diciembre de 2024

**Estado:**  SOLUCIÓN DEFINITIVA

## POR QUÉ VPS ES LA SOLUCIÓN DEFINITIVA

Después de intentar Railway y Vercel sin éxito, **un VPS te da control total**:

-  **Tú decides** dónde y cómo construir
-  **No hay "Root Directory"** problemático
-  **Escalable** - aumenta RAM/CPU cuando quieras
-  **Estructura flexible** - funciona con `nextjs_space/` o cualquier estructura
-  **PostgreSQL incluido** en Docker Compose
-  **Nginx como proxy** con SSL automático
-  **\$12-20/mes** - costo predecible



## PASOS PARA DEPLOYMENT (30 MINUTOS)

### 1. Crear VPS

Opciones recomendadas:

#### A) DigitalOcean Droplet (Más popular)

1. Ve a [DigitalOcean.com](https://www.digitalocean.com) (<https://www.digitalocean.com>)
2. Create → Droplets
3. Elige:
  - **Imagen:** Ubuntu 22.04 LTS
  - **Plan:** Basic \$12/mes (2GB RAM, 1 CPU)
  - **Región:** Frankfurt (más cercano a España)
  - **SSH Key:** Añade tu clave pública (o usa password)
4. Create Droplet

#### B) Hetzner (Más barato - €4.15/mes)

1. Ve a [Hetzner.com](https://www.hetzner.com/cloud) (<https://www.hetzner.com/cloud>)
2. Servidor similar, región Falkenstein (Alemania)

#### C) Linode (Alternativa)

1. [Linode.com](https://www.linode.com) (<https://www.linode.com>)
2. Plan Nanode \$12/mes

## 2. Conectarte al VPS

```
# SSH al servidor
ssh root@tu_ip_del_vps

# Actualizar sistema
apt update && apt upgrade -y
```

## 3. Instalar Docker y Docker Compose

```
# Instalar Docker
curl -fsSL https://get.docker.com -o get-docker.sh
sh get-docker.sh

# Instalar Docker Compose
apt install docker-compose -y

# Verificar instalación
docker --version
docker-compose --version
```

## 4. Clonar Tu Repositorio

```
# Instalar Git
apt install git -y

# Clonar repo
git clone https://github.com/dvillagrablanco/inmova-app.git
cd inmova-app
```

## 5. Configurar Variables de Entorno

```
# Copiar ejemplo
cp .env.production.example .env.production

# Editar con nano o vim
nano .env.production
```

**Configurar:**

```
DATABASE_URL=postgresql://inmova_user:CAMBIA_ESTO@postgres:5432/inmova
POSTGRES_PASSWORD=CAMBIA_ESTO
NEXTAUTH_SECRET=GENERA_ESTO
NEXTAUTH_URL=http://tu_ip_vps:3000
```

**Generar NEXTAUTH\_SECRET :**

```
openssl rand -base64 32
```

## 6. Construir y Lanzar

```
# Build y start
docker-compose -f docker-compose.production.yml up -d --build

# Ver logs
docker-compose -f docker-compose.production.yml logs -f
```

**Tiempo estimado:** 10-15 minutos para build

## 7. Verificar

```
# Ver contenedores corriendo
docker ps

# Deberías ver 3 contenedores:
# - inmova-app-1 (Next.js)
# - inmova-postgres-1 (PostgreSQL)
# - inmova-nginx-1 (Nginx proxy)
```

**Abre en navegador:**

```
http://tu_ip_vps
```

 ¡Debería funcionar!

## CONFIGURAR SSL (HTTPS)

### Opción 1: Let's Encrypt con Certbot (GRATIS)

```
# Instalar Certbot
apt install certbot python3-certbot-nginx -y

# Detener Nginx temporal
docker-compose -f docker-compose.production.yml stop nginx

# Obtener certificado
certbot certonly --standalone -d tu-dominio.com

# Copiar certificados
mkdir -p ssl
cp /etc/letsencrypt/live/tu-dominio.com/fullchain.pem ssl/
cp /etc/letsencrypt/live/tu-dominio.com/privkey.pem ssl/

# Editar nginx.conf (descomentar sección HTTPS)
nano nginx.conf

# Reiniciar
docker-compose -f docker-compose.production.yml up -d
```

### Opción 2: Cloudflare (GRATIS + CDN)

1. Añade tu dominio a Cloudflare

2. Apunta DNS A record a tu IP del VPS
  3. Cloudflare maneja SSL automáticamente
  4. **Bonus:** CDN gratis global
- 

## ACTUALIZACIONES Y MAINTENANCE

### Deploy Nueva Versión:

```
cd inmova-app

# Pull últimos cambios
git pull origin main

# Rebuild y redeploy
docker-compose -f docker-compose.production.yml up -d --build
```

### Ver Logs:

```
# Todos los servicios
docker-compose -f docker-compose.production.yml logs -f

# Solo app
docker-compose -f docker-compose.production.yml logs -f app

# Solo postgres
docker-compose -f docker-compose.production.yml logs -f postgres
```

### Backup de Base de Datos:

```
# Dump de PostgreSQL
docker-compose -f docker-compose.production.yml exec postgres pg_dump -U inmova_user inmova > backup_$(date +%Y%m%d).sql

# Restaurar
docker-compose -f docker-compose.production.yml exec -T postgres psql -U inmova_user inmova < backup_20241213.sql
```

### Reiniciar Servicios:

```
# Reiniciar app
docker-compose -f docker-compose.production.yml restart app

# Reiniciar todo
docker-compose -f docker-compose.production.yml restart

# Detener todo
docker-compose -f docker-compose.production.yml down

# Iniciar todo
docker-compose -f docker-compose.production.yml up -d
```



## MONITORING Y ESCALABILIDAD

### Monitorear Recursos:

```
# CPU y RAM
htop

# Uso de disco
df -h

# Logs de Docker
docker stats
```

### Escalar (cuando necesites más potencia):

#### DigitalOcean:

1. Droplet → Resize
2. Elige plan superior (\$24/mes = 4GB RAM)
3. **Sin downtime** (live resize)

#### Alternativa: Escalar horizontalmente

- Añade otro droplet
- Usa Load Balancer de DigitalOcean (\$12/mes)



## COSTOS MENSUALES

### Configuración Básica:

VPS (2GB RAM) :	\$12/mes
Total:	\$12/mes

### Configuración Profesional:

VPS (4GB RAM) :	\$24/mes
Backups automáticos:	\$5/mes
Load Balancer:	\$12/mes (si necesitas)
Total:	\$29-41/mes

**vs Railway:** \$20/mes (sin DB) + \$15/mes (DB) = \$35/mes

**vs Vercel + DB:** \$20/mes (Pro) + \$20/mes (DB) = \$40/mes

**VPS es más barato Y tienes control total.**

## TROUBLESHOOTING

### App no inicia:

```
# Ver logs específicos
docker-compose -f docker-compose.production.yml logs app

# Verificar .env
cat .env.production

# Verificar que DATABASE_URL es correcto
```

### Postgres no conecta:

```
# Verificar que postgres está corriendo
docker ps | grep postgres

# Entrar a postgres y probar conexión
docker-compose -f docker-compose.production.yml exec postgres psql -U inmova_user in-
mova
```

### Puerto 3000 no accesible:

```
# Abrir firewall
ufw allow 80
ufw allow 443
ufw allow 22
ufw enable

# Verificar nginx
curl localhost:80
```

### Build falla:

```
# Ver logs completos del build
docker-compose -f docker-compose.production.yml build --no-cache app

# Si falla por memoria, añade swap:
fallocate -l 2G /swapfile
chmod 600 /swapfile
mkswap /swapfile
swapon /swapfile
```

## VENTAJAS DE VPS

1. **Control Total:** Tú decides TODO
2. **Sin Restricciones:** No hay “Root Directory” problemático
3. **Flexible:** Acepta cualquier estructura de carpetas
4. **Escalable:** Aumenta recursos con un click
5. **Predecible:** Costo fijo mensual
6. **Transparente:** Ves exactamente qué está pasando

7. **Portable:** Puedes migrar a otro VPS fácilmente

## CHECKLIST FINAL

### Pre-Deployment:

- VPS creado
- Docker instalado
- Repositorio clonado
- .env.production configurado

### Deployment:

- docker-compose up exitoso
- 3 contenedores corriendo
- App accesible en http://IP
- Database conectada

### Post-Deployment:

- SSL configurado (Certbot o Cloudflare)
- Dominio apuntando al VPS
- Backups configurados
- Monitoring activo

## SOPORTE

Si tienes problemas:

1. Revisa logs: `docker-compose logs -f`
2. Verifica .env.production
3. Confirma que los 3 contenedores están corriendo: `docker ps`

## CONCLUSIÓN

**VPS con Docker Compose es la solución más robusta y flexible.**

- ⏳ Tiempo de setup: **30 minutos**
- 💰 Costo: **\$12/mes**
- 🎯 Probabilidad de éxito: **99%**
- 😊 Control: **100% tuyo**

**No más problemas de “Root Directory”. No más frustraciones.**

**Deploy una vez, funciona para siempre.** 

**Timestamp:** 2024-12-13 20:00 UTC

**Archivos creados:**

- docker-compose.production.yml
- Dockerfile.production
- nginx.conf
- .env.production.example