

RESUMEN DE CONFIGURACIÓN DE PRODUCCIÓN - INMOVA


Fecha: Diciembre 2025

Dominio: inmov.a.app


Estado: Parcialmente configurado

CONFIGURACIONES COMPLETADAS

1. Entorno de Producción

NODE_ENV=production ( Configurado)



2. Base de Datos

DATABASE_URL con parámetros optimizados ( Configurado)

- connection_limit=20
- pool_timeout=15
- connect_timeout=15


Nota: DATABASE_URL es una variable reservada del sistema, los parámetros de pool deben configurarse directamente en Prisma si es necesario.

3. Dominio


Dominio principal: inmov.a.app ( Activo)
NEXTAUTH_URL: https://inmov.a.app ( Configurado)

4. Archivos Creados

Health Check Endpoint

-  /app/api/health/route.ts
- Verifica estado de aplicación
 - Verifica conexión a base de datos
 - Endpoint: https://inmov.a.app/api/health

Configuración de Email

-  /lib/email-config.ts
- Soporte para SendGrid
 - Soporte para SMTP genérico
 - Templates predefinidos
 - Funciones de envío

Scripts de Backup

- ✓ /home/ubuntu/scripts/backup-daily.sh
 - Backup diario de PostgreSQL
 - Compresión gzip
 - Subida a S3 (opcional)
 - Limpieza automática (7 días)
- ✓ /home/ubuntu/scripts/backup-weekly.sh
 - Backup semanal
 - Retención de 12 semanas

Scripts de Deployment

- ✓ /home/ubuntu/scripts/deploy.sh
 - Deployment con zero downtime
 - Backup pre-deploy
 - Health check post-deploy
 - Rollback automático en caso de fallo
- ✓ /home/ubuntu/scripts/rollback.sh
 - Revierte al commit anterior
 - Rebuild automático
 - Verificación de salud
- ✓ /home/ubuntu/scripts/restore-from-backup.sh
 - Restauración desde backup
 - Backup de seguridad antes de restaurar
 - Soporte para S3

Todos los scripts tienen permisos de ejecución (✓ +x)

5. Documentación

- ✓ CONFIGURACION_PRODUCCION.md
 - Checklist completo
 - Configuración detallada de cada servicio
 - Plan de rollback
 - Métricas y KPIs
- ✓ GUIA CONFIGURACION_PASO_A_PASO.md
 - Instrucciones paso a paso
 - Screenshots y ejemplos
 - Comandos listos para copiar/pegar
 - Troubleshooting



CONFIGURACIONES PENDIENTES

Requieren Acción Manual

1. Sentry (Error Tracking)

Estado: ⚠ Pendiente configurar credenciales

Qué hacer:

1. Crear cuenta en <https://sentry.io>
2. Crear proyecto "inmova-production"
3. Obtener DSN y Auth Token
4. Configurar variables de entorno:

```
env
```

```
NEXT_PUBLIC_SENTRY_DSN=[DSN]
```

```
SENTRY_ORG=[ORG]
```

```
SENTRY_PROJECT=inmova-production
```

```
SENTRY_AUTH_TOKEN=[TOKEN]
```

Tiempo estimado: 15 minutos**Prioridad:** ● Alta**Guía:** Ver sección 1 en [GUIA_CONFIGURACION_PASO_A_PASO.md](#)

2. UptimeRobot (Monitoring)

Estado: ⚠ Pendiente configurar**Qué hacer:**

1. Crear cuenta en <https://uptimerobot.com>
2. Crear monitor para <https://inmova.app/api/health>
3. Configurar alertas por email/SMS

Tiempo estimado: 10 minutos**Prioridad:** ● Alta**Guía:** Ver sección 2 en [GUIA_CONFIGURACION_PASO_A_PASO.md](#)

3. Backups Automáticos

Estado: ⚠ Scripts creados, pendiente configurar cron**Qué hacer:**

1. Instalar PostgreSQL client: `sudo apt-get install postgresql-client`
2. Crear directorios: `mkdir -p /home/ubuntu/backups/{daily,weekly}`
3. Probar script: `/home/ubuntu/scripts/backup-daily.sh`
4. Configurar cron jobs:

```
bash
```

```
crontab -e
```

```
# Añadir:
```

```
0 3 * * * /home/ubuntu/scripts/backup-daily.sh >> /home/ubuntu/logs/backup-daily.log 2>&1
```

```
0 2 * * 0 /home/ubuntu/scripts/backup-weekly.sh >> /home/ubuntu/logs/backup-weekly.log 2>&1
```

Tiempo estimado: 10 minutos**Prioridad:** ● Alta**Guía:** Ver sección 3 en [GUIA_CONFIGURACION_PASO_A_PASO.md](#)

4. Cloudflare CDN

Estado: ⚠ Pendiente configurar

Qué hacer:

1. Crear cuenta en <https://cloudflare.com>
2. Añadir dominio `inmova.app`
3. Cambiar nameservers en registrador de dominio
4. Configurar SSL/TLS (Full strict)
5. Configurar reglas de caché
6. Activar optimizaciones (Brotli, HTTP/3, etc.)

Tiempo estimado: 20 minutos

Prioridad: 🟡 Media (mejora rendimiento)

Guía: Ver sección 4 en `GUIA_CONFIGURACION_PASO_A_PASO.md`

5. Stripe Producción

Estado: ⚠ Configurado con claves de TEST

Qué hacer:

1. Activar modo LIVE en dashboard de Stripe
2. Completar verificación de negocio
3. Obtener claves LIVE (pk_live_... y sk_live_...)
4. Configurar webhook en producción
5. Actualizar variables de entorno:

`env`

`STRIPE_SECRET_KEY=sk_live_[CLAVE]`

`STRIPE_PUBLISHABLE_KEY=pk_live_[CLAVE]`

`STRIPE_WEBHOOK_SECRET=whsec_[SECRET]`

`NEXT_PUBLIC_STRIPE_PUBLISHABLE_KEY=pk_live_[CLAVE]`

Tiempo estimado: 15 minutos

Prioridad: 🔴 Crítica (para pagos reales)

Guía: Ver sección 5 en `GUIA_CONFIGURACION_PASO_A_PASO.md`

6. SendGrid (Email Transaccional)

Estado: ⚠ Módulo creado, pendiente credenciales

Qué hacer:

1. Crear cuenta en <https://sendgrid.com>
2. Crear API Key
3. Verificar dominio `inmova.app`
4. Configurar registros DNS (CNAME, SPF, DKIM)
5. Configurar variables de entorno:

`env`

`SENDGRID_API_KEY=SG.[CLAVE]`

`SENDGRID_FROM_EMAIL=noreply@inmova.app`

`SENDGRID_FROM_NAME=INMOVA`

6. Probar envío de email

Tiempo estimado: 20 minutos

Prioridad: ● Alta (para emails de sistema)

Guía: Ver sección 6 en `GUIA_CONFIGURACION_PASO_A_PASO.md`



RESUMEN DE PRIORIDADES



Críticas (Hacer Primero)

1. **Stripe Producción** - Sin esto, no se pueden procesar pagos reales
2. **SendGrid** - Necesario para emails de sistema (recuperación de contraseña, etc.)
3. **Backups Automáticos** - Protección de datos
4. **Sentry** - Detección temprana de errores
5. **UptimeRobot** - Monitoring 24/7



Importantes (Hacer Pronto)

1. **Cloudflare CDN** - Mejora rendimiento y seguridad



TIEMPO TOTAL ESTIMADO

Configuración completa: 2-3 horas

Desglose:

- Sentry: 15 min
- UptimeRobot: 10 min
- Backups: 10 min
- Cloudflare: 20 min
- Stripe: 15 min
- SendGrid: 20 min
- Verificaciones: 15 min



CÓMO PROCEDER

Opción 1: Configuración Secuencial

Seguir la guía paso a paso:

```
cat /home/ubuntu/homming_vidaro/GUIA_CONFIGURACION_PASO_A_PASO.md
```

Completar cada sección en orden.

Opción 2: Configuración por Prioridad

1. Configurar Stripe (crítico para pagos)
2. Configurar SendGrid (crítico para emails)
3. Configurar backups (protección de datos)
4. Configurar monitoring (Sentry + UptimeRobot)

5. Configurar CDN (optimización)

Opción 3: Delegar

Asignar tareas a diferentes miembros del equipo:

- **DevOps/SysAdmin:** Backups, scripts, cron jobs
- **Backend Dev:** Sentry, monitoring, logs
- **Frontend Dev:** CDN, optimizaciones
- **Product Owner:** Stripe, SendGrid, configuraciones de negocio

CHECKLIST FINAL

Una vez completadas todas las configuraciones:

- ☐ Sentry capturando errores
- ☐ UptimeRobot monitoreando uptime
- ☐ Backups ejecutándose diariamente
- ☐ Cloudflare CDN activo
- ☐ Stripe en modo LIVE funcionando
- ☐ SendGrid enviando emails
- ☐ Health check respondiendo
- ☐ Scripts de deploy probados
- ☐ Scripts de rollback probados
- ☐ Documentación actualizada
- ☐ Equipo entrenado en procedimientos
- ☐ Contactos de emergencia documentados

SOPORTE

Documentación Disponible

1. **CONFIGURACION_PRODUCCION.md** - Configuración técnica completa
2. **GUIA_CONFIGURACION_PASO_A_PASO.md** - Instrucciones detalladas
3. **RESUMEN_CONFIGURACION.md** - Este documento

Scripts Disponibles

```
/home/ubuntu/scripts/
├─ backup-daily.sh      # Backup diario
├─ backup-weekly.sh    # Backup semanal
├─ deploy.sh           # Deploy con zero downtime
├─ rollback.sh         # Rollback a versión anterior
└─ restore-from-backup.sh # Restaurar desde backup
```

Logs

```
# Ver logs de aplicación
pm2 logs innova-app

# Ver logs de backups
tail -f /home/ubuntu/logs/backup-daily.log

# Ver logs de deploy
ls -lh /home/ubuntu/logs/deploy-*.log
```

Documento creado: Diciembre 2025

Última actualización: Diciembre 2025

Mantenido por: Equipo DevOps INNOVA

Versión: 1.0