



# Checklist de Mantenimiento INMOVA

---



## Mantenimiento Semanal

---

### Limpieza de Cachés

- [ ] Ejecutar script de limpieza: `bash scripts/clean-caches.sh`
- [ ] Verificar espacio en disco: `df -h`
- [ ] Limpiar logs antiguos: `find . -name "*.log" -type f -mtime +7 -delete`

### Verificación de Sistema

- [ ] Ejecutar análisis de base de datos: `yarn tsx --require dotenv/config scripts/optimize-data-base.ts`
- [ ] Validar esquema Prisma: `yarn prisma validate`
- [ ] Ver estado de migraciones: `yarn prisma migrate status`
- [ ] Ejecutar tests: `yarn test:ci`

### Monitoreo

- [ ] Revisar logs de aplicación
  - [ ] Verificar errores en consola del navegador
  - [ ] Revisar métricas de rendimiento
  - [ ] Verificar uso de memoria y CPU
- 



## Mantenimiento Mensual

---

### Análisis Profundo

- [ ] Ejecutar análisis de dependencias: `npx depcheck`
- [ ] Revisar actualizaciones disponibles: `yarn outdated`
- [ ] Verificar vulnerabilidades: `yarn audit`
- [ ] Analizar tamaño de bundles

### Optimización de Base de Datos

- [ ] Ejecutar VACUUM ANALYZE: `psql $DATABASE_URL -c "VACUUM ANALYZE;"`
- [ ] Verificar queries lentas
- [ ] Revisar índices de tablas
- [ ] Analizar tamaño de base de datos

### Limpieza de Datos

- [ ] Archivar registros antiguos (>6 meses)
- [ ] Eliminar datos temporales no necesarios
- [ ] Verificar integridad referencial
- [ ] Limpiar archivos subidos no referenciados

## Backup y Seguridad

- ☐ Verificar backups automáticos
- ☐ Probar restauración de backup
- ☐ Revisar logs de seguridad
- ☐ Actualizar credenciales si es necesario



## Mantenimiento Trimestral

### Revisión Mayor

- ☐ Actualizar dependencias críticas
- ☐ Evaluar eliminación de dependencias no usadas
- ☐ Refactorizar código obsoleto
- ☐ Optimizar queries más lentas

### Performance

- ☐ Ejecutar audit de rendimiento completo
- ☐ Analizar y optimizar bundle size
- ☐ Revisar y optimizar imágenes
- ☐ Evaluar necesidad de CDN o caching

### Documentación

- ☐ Actualizar README
- ☐ Documentar nuevas features
- ☐ Actualizar guías de deployment
- ☐ Revisar y actualizar API docs



## Tareas Urgentes (Alta Prioridad)



### Acción Inmediata

#### 1. Limpiar directorio .build (2.8 GB)

```
cd /home/ubuntu/homming_vidaro/nextjs_space
find .build -type f -delete 2>/dev/null || true
yarn build # Regenerar build limpio
```

**Status:** 🕒 Pendiente

**Impacto:** Libera 2.8 GB de espacio

**Prioridad:** Alta

**Fecha límite:** Esta semana

## 2. Revisar 5 edificios sin unidades

```
# Identificar edificios sin unidades
psql $DATABASE_URL -c "SELECT id, nombre, direccion FROM \"Building\" WHERE id NOT IN
(SELECT DISTINCT \"buildingId\" FROM \"Unit\");"
```

**Status:** ⏳ Pendiente

**Impacto:** Limpieza de datos

**Prioridad:** Media

**Fecha límite:** Este mes

## 3. Optimizar PostgreSQL

```
psql $DATABASE_URL -c "VACUUM ANALYZE;"
```

**Status:** ⏳ Pendiente

**Impacto:** Mejora de rendimiento 30-50%

**Prioridad:** Alta

**Fecha límite:** Esta semana

# 🟡 Tareas Recomendadas (Media Prioridad)

## 1. Evaluar Eliminación de Dependencias No Usadas

**Dependencias probablemente seguras para eliminar:**

```
yarn remove @dnd-kit/core @dnd-kit/sortable @dnd-kit/utilities \
formik gray-matter i18next i18next-browser-languagedetector \
react-i18next isomorphic-dompurify mapbox-gl next-intl \
swagger-jsdoc tailwind-scrollbar-hide yup
```

**Status:** ⏳ Pendiente

**Impacto:** 50-100 MB liberados

**Prioridad:** Media

**Nota:** Probar en rama separada primero

## 2. Configurar Mantenimiento Automático

Crear cron job para limpieza semanal:

```
# Agregar a crontab
0 2 * * 0 cd /home/ubuntu/homming_vidaro/nextjs_space && bash scripts/clean-caches.sh
```

**Status:** ⏳ Pendiente

**Impacto:** Mantenimiento preventivo

**Prioridad:** Media

## Tareas de Mejora (Baja Prioridad)

---

### 1. Implementar Monitoring

- [ ] Configurar alertas de espacio en disco
- [ ] Implementar tracking de performance
- [ ] Configurar alertas de errores

### 2. Optimizaciones de Código

- [ ] Implementar code splitting más granular
  - [ ] Optimizar imágenes con next/image
  - [ ] Implementar lazy loading donde sea posible
- 

## Métricas de Éxito

---

### Espacio en Disco

- 🎯 **Objetivo:** Liberar 2.9 GB
- 📊 **Progreso actual:** 0% (0 GB liberados)
- ⌚ **Plazo:** 1 semana

### Rendimiento de Queries

- 🎯 **Objetivo:** Reducir tiempo promedio 30%
- 📊 **Estado:** Por medir (ejecutar VACUUM ANALYZE primero)
- ⌚ **Plazo:** 1 semana

### Tamaño de Build

- 🎯 **Objetivo:** Reducir 5-10%
  - 📊 **Estado:** Por medir (eliminar deps no usadas)
  - ⌚ **Plazo:** 2 semanas
- 

## Notas de Mantenimiento

---

### Última Limpieza Completa

**Fecha:** 5 de Diciembre 2025

**Ejecutado por:** DeepAgent

**Resultados:** Ver RESUMEN\_LIMPIEZA.txt

### Próxima Limpieza Programada

**Fecha:** 12 de Diciembre 2025

**Tipo:** Semanal

**Script:** `bash scripts/clean-caches.sh`

### Observaciones

- Base de datos en buen estado (✅)
- 560 índices ya configurados (✅)

- Schema Prisma válido (✅)
- 2.8 GB en .build requiere atención (⚠️)
- 35 dependencias no usadas detectadas (🟡)

---

## Enlaces Útiles

- [REPORTE\\_LIMPIEZA.md](#) (./REPORTE\_LIMPIEZA.md) - Análisis detallado
- [COMANDOS\\_MANTENIMIENTO.md](#) (./COMANDOS\_MANTENIMIENTO.md) - Comandos útiles
- [RESUMEN\\_LIMPIEZA.txt](#) (./RESUMEN\_LIMPIEZA.txt) - Resumen ejecutivo
- [scripts/optimize-database.ts](#) (./scripts/optimize-database.ts) - Script de optimización DB
- [scripts/clean-caches.sh](#) (./scripts/clean-caches.sh) - Script de limpieza

---

## Progreso General

[#####-----] 55% Completado

Completadas: 5/9 tareas principales

Pendientes:

- Limpieza de .build (2.8 GB)
- Optimización PostgreSQL
- Revisión de edificios sin unidades
- Eliminación de dependencias no usadas

---

**Última actualización:** 5 de Diciembre 2025

**Responsable:** Equipo de Desarrollo INMOVA

**Revisión:** Semanal