## <

# ECLIPSE CONDOS

### proyecto de desarrollo de página web

### visualización de apartamentos y de información de eclipse condos.

Eclipse Condominios WEB Page

DOCUMENTOS SRS

|  |  |
| --- | --- |
| PROYECTO NO. | 01 |
| VERSIÓN: 1.0 | Fecha**:** 24/06/2025 |
| RESPONSABLES: | ANDRÉS E. MOREIRA LEE |
| PROYECTO: | DESARROLLO DE PLATAFORMA WEB ECLIPSE CONDOS |

## 1. Introducción

Este documento tiene como objetivo describir las decisiones tecnológicas tomadas hasta el momento para la construcción del sitio web oficial de Eclipse Condominios en Playa del Carmen. Se basa en los entregables anteriores como el Plan de Alcance y la Estructura de Trabajo (WBS). El repositorio oficial del proyecto se encuentra en GitHub:  
🔗 <https://github.com/Leemorand1410/eclipse-condos-web>

## 2. Stack Tecnológico v1.1

### 2.1 Frontend

**Framework:** Next.js (basado en React)

**Lenguaje:** TypeScript

**Motivos:** Next.js permite renderizado estático y del lado del servidor (SSR), ideal para SEO, multilenguaje y velocidad de carga. TypeScript ofrece mayor robustez en el desarrollo.

### 2.2 Backend

**Framework:** Laravel 10.x

**Lenguaje:** PHP 8+

**Motivos:** Laravel permite el desarrollo de APIs REST modernas, administración segura y organización clara de la lógica del negocio.

### 2.3 Base de Datos

**Sistema de Base de Datos:** PostgreSQL (temporalmente definido)

**Motivos:** Compatibilidad con Laravel, estructura relacional potente y adecuada para datos estructurados del sitio.

### 2.4 Servidor

**Arquitectura Inicial:** El despliegue de Next.js y Laravel se realizará utilizando un **proxy reverso** (Nginx o Apache) para unificar el acceso bajo un mismo dominio y evitar el doble hosteo visible.

Las rutas como /api/\* o /admin/\* serán dirigidas hacia Laravel

Las rutas públicas /, /departamentos, /galeria serán servidas desde Next.js

Ambas aplicaciones se ejecutarán en puertos internos distintos (por ejemplo 3000 para Next.js y 8000 para Laravel)

Todo el tráfico se canalizará desde Nginx como punto de entrada

El dominio actual que se reutilizará es: **eclipsecondosmexico.com**

Tareas de implementación: Definir el archivo de configuración de Nginx, asegurar HTTPS con certificados Let's Encrypt, configurar la redirección del dominio eclipsecondosmexico.com al VPS o servidor de producción, y preparar scripts de despliegue conjunto.

### 2.5 Seguridad Mínima

**Protocolo Seguro:** HTTPS obligatorio

**Prácticas:** Sanitización de entradas, encabezados seguros, bloqueo de métodos innecesarios.

## 3. Estructura del Repositorio Técnico

El proyecto está organizado como un **monorepo**, con separación de aplicaciones (frontend y backend), librerías compartidas y configuración de infraestructura:

eclipse-condos-web/

├── apps/

│ ├── frontend/ # Next.js (View + ViewModel)

│ │ ├── package.json

│ │ └── src/

│ │ ├── models/ # Model: tipos y entidades

│ │ ├── viewmodels/ # ViewModel: hooks y estado

│ │ ├── views/ # View: pages y componentes

│ │ │ ├── pages/

│ │ │ └── components/

│ │ ├── services/ # API clients, side‐effects

│ │ └── utils/ # helpers genéricos

│ └── backend/ # Laravel (DDD + capas)

│ ├── app/

│ │ ├── Domain/ # Entidades, repositorios, reglas de negocio

│ │ ├── Application/ # Casos de uso, servicios de aplicación

│ │ └── Infrastructure/ # Adaptadores (Eloquent, correo, etc.)

│ ├── routes/ # Definición de rutas

│ └── …

├── libs/ # Código compartido (opcional)

│ ├── ui/ # Componentes React

│ └── data/ # Tipos y clientes HTTP

├── infra/ # Configuración infra (nginx, docker…)

│ └── nginx/

│ └── default.conf

├── docs/ # Documentación técnica

│ ├── SRS-v1.md

│ └── Infraestructura.md

├── .gitignore

└── README.md

Cada aplicación tiene su propio entorno, dependencias y punto de inicio. Las librerías bajo libs/ permiten mantener código reutilizable (como componentes de UI y estructuras de datos comunes).

## 4. Funcionalidades fuera del alcance inicial

Las siguientes funcionalidades están consideradas para futuras fases, pero **no forman parte del alcance del presente sprint de desarrollo**:

* ****Formulario de contacto funcional:**** se prevé el uso de una API propia para el envío de datos (email, nombre, mensaje). Esta funcionalidad se documentará y probará en un sprint posterior.
* ****Panel de administración completo:**** sólo se entregará una estructura básica para gestión futura.
* ****Sistema de login/autenticación:**** se mantiene fuera del alcance actual.
* ****Soporte multilenguaje completo:**** se preparará la arquitectura, pero el contenido inicial solo estará en español.

## 5. Pendientes técnicos por definir

|  |  |  |
| --- | --- | --- |
| Tema | Estado Actual | Acción Propuesta |
| Configuración de Nginx | En proceso | Crear archivo nginx.conf con reglas de proxy reverso |
| Certificados HTTPS | En proceso | Automatizar con Let's Encrypt y renovar con cron |
| Redirección de dominio | En proceso | Actualizar registros DNS de eclipsecondosmexico.com para apuntar al servidor |
| Elección final de BD | Temporalmente PostgreSQL | Confirmar en la fase de desarrollo del panel |
| CMS o Admin Panel | No definido | Evaluar entre panel propio vs. CMS Headless |
| Sistema de traducción | No implementado | Implementar next-i18next o similar |

## 6. Conclusión

Este documento establece las bases del stack tecnológico para comenzar con la estructura del repositorio, el desarrollo de componentes iniciales y la preparación del backend. Se actualizará de forma iterativa conforme avance el desarrollo del proyecto.