

# Especificación de Arquitectura e Implementación



Documento base para alcance, arquitectura, operación y criterios de entrega del portal web institucional.

PROYECTO	Portal Web CEINPET	FECHA DE EMISIÓN	13/03/2026
DOCUMENTO	Especificación de Arquitectura e Implementación	FORMATO	PDF ejecutivo
IDIOMA	Español	CLASIFICACIÓN	Entregable de proyecto

Fecha: 2026-03-07

## Especificación de Implementación Integral

### Portal Web CEINPET (Headless Drupal + Next.js)

Fecha: 2026-03-07 Idioma oficial del proyecto: Español Fuente base de planificación: [cronograma-desarrollo-portal-web.pdf](#)

#### 1. Objetivo

Implementar un portal institucional moderno, performante, bilingüe (ES/EN), 100% gestionado desde CMS para contenido y configuración visual/estructural clave, recuperando y modernizando todas las páginas del sitio anterior.

#### 2. Alcance funcional acordado

- Recuperar todas las páginas del sitio legado (con libertad de modernizar nombre/slug/estructura/contenido).
- Modernización general del portal.
- Home con bloques obligatorios:
- Hero.
- Acceso rápido a secciones clave.
- Soluciones/servicios destacados.
- Líneas de investigación.
- Publicaciones recientes.
- Misión / Visión / Valores (diseño tríptico o equivalente profesional).
- CTA de contacto.
- Aliados/instituciones relacionadas.
- Menú principal objetivo:
- Acerca de.
- Investigación - Desarrollo (posible división).

- Gestión del conocimiento.
- Líneas de investigación.
- Publicaciones.
- Contacto.
- Navegación con submenús modernos/profesionales, alineados al modelo CMS.
- Soporte de páginas legales desde lanzamiento:
- Política de privacidad.
- Términos de uso.
- Política de cookies.
- Declaración de accesibilidad.

### 3. Modelo editorial y CMS

#### 3.1 Principios

- Header y Footer 100% CMS-driven (sin hardcodeo frontend).
- Uso preferente de tipos genéricos ( `page / article` ) + `Paragraphs` como building blocks.
- Libertad para crear tipos adicionales cuando el caso lo requiera.

#### 3.2 Tipos de contenido

- Existentes prioritarios:
- `solucion_cientifico_tecnica`
- `pagina_de_coleccion`
- `page`
- `article`
- Nuevos/ajustes previstos:
- `contacto` (recepción de envíos del formulario público).
- Paragraph bundles para composición flexible.
- Ajustes globales de sitio (header/footer/SEO/branding) vía entidad de settings.

#### 3.3 Paragraphs (mínimo inicial)

- Hero.
- Rich text.
- Grid de tarjetas.
- CTA.
- Listado destacado.
- Descargables/documentos.
- KPI/estadísticas.
- FAQ.
- Timeline.
- Misión/Visión/Valores.
- Logos de aliados.

### 4. Flujo de contacto

- Frontend envía formulario con campos:
- nombre, email, teléfono, asunto, mensaje.

- Backend crea contenido `Contacto` en Drupal.
- Moderación editorial en CMS.
- Notificación por correo configurable en CMS (regla/workflow Drupal).

## 5. Internacionalización

- Requisito de lanzamiento: Español + Inglés.
- Contenido traducible desde CMS.
- Rutas y metadatos con soporte i18n.

## 6. SEO y descubribilidad

Implementación "lo más completa posible" gestionada desde Drupal:

- Metadatos por página (title, description, OG, Twitter).
- Canonical y hreflang.
- Sitemap.xml y robots.txt.
- Schema.org (Organization, WebSite, Breadcrumb, Article/CreativeWork según tipo).
- Soporte para redirecciones y slugs históricos cuando aplique.

## 7. Accesibilidad

- Regla general obligatoria: WCAG 2.1 AA en lanzamiento.

## 8. Rendimiento y calidad

### 8.1 Metas

- Lighthouse móvil:
- Performance  $\geq 85$
- SEO  $\geq 95$
- Best Practices  $\geq 95$
- Accessibility  $\geq 95$
- Core Web Vitals (p75):
- LCP  $< 2.5s$
- CLS  $< 0.1$
- INP  $< 200ms$

### 8.2 Estrategia técnica

- SSG/ISR + revalidación por `path` y `tags` (implementada en endpoint `/api/revalidate` y en hooks CMS).
- Optimización de imágenes (formatos modernos, responsive, lazy loading).
- Alt text y focal point desde CMS.

## 9. Búsqueda

- Alcance inicial opcional y focalizado: publicaciones, investigaciones y soluciones.
- Se implementa si aporta valor funcional claro en UX.

## 10. Seguridad y visibilidad

- Sitio público (sin áreas autenticadas frontend en esta fase).
- Flujo editorial simple:
- Rol editor crea/modifica.
- Publicar => visible en frontend + revalidación.

## 11. Entornos y despliegue

- CMS local: DDEV.
- CMS remoto: `ceinpet-cms.drr.nat.cu`.
- Frontend remoto y producción: Vercel ( `ceinpet-web.vercel.app` ).

### Variables técnicas adicionales

- `DRUPAL_BASIC_AUTH_USER` y `DRUPAL_BASIC_AUTH_PASS` en frontend para integrar el endpoint `/api/contact` con JSON:API de Drupal ( `node - - contacto` ).

## 12. Analítica y monitoreo

- GA4.
- Vercel Analytics.
- Logs en VPS Drupal.
- Alertas básicas de uptime.

## 13. Fuentes para migración/backfill

- Referencia pública: `https://www.ceinpet.cupet.cu/es/`.
- Recursos remanentes locales: carpeta `pceinpet/`.
- Se elaborará plan de backfill con plantillas editables por tipo/idioma.

## 14. Definición de Done

- Frontend utilizable, navegable y moderno.
- SEO funcional y completo.
- Secciones enriquecidas más allá del contenido mínimo histórico.
- CMS capaz de gestionar todo contenido y ajustes necesarios del frontend.
- Documentación y artefactos del cronograma listos en español.

## 15. Entregables documentales obligatorios

Ver `docs/cronograma-entregables.md` como checklist base mínimo. Además, se crearán artefactos complementarios para trazabilidad, calidad y handoff.

## 16. Estrategia de implementación por bloques

1. Base CMS-driven global (header/footer/settings + schema).
2. Page builder (Paragraphs + mapeo frontend).
3. i18n end-to-end (CMS + Next.js routing/contenido).
4. Páginas objetivo y recuperación de estructura legacy.
5. SEO técnico integral.
6. Formulario de contacto con persistencia en Drupal.

7. Optimización, pruebas y hardening.
8. Backfill de contenido y documentación final.

Documento generado automáticamente a partir de la versión fuente en Markdown ubicada en `docs/`.