

PORTAL EDUCATIVO PARA PADRES

Análisis Funcional Detallado

Documento de especificaciones para desarrollo con Claude Code

Versión: 1.0

Fecha: Diciembre 2024

Tipo: Aplicación Web

Stack: Next.js 14 + TypeScript + PostgreSQL

ÍNDICE

- 1. Visión General del Proyecto
- 2. Arquitectura Técnica
- 3. Estructura de Base de Datos
- 4. Backoffice (Panel del Profesor)
- 5. Portal Público (Para Padres)
- 6. Diseño de Interfaz
- 7. Plan de Desarrollo por Fases
- 8. Estructura de Archivos
- 9. Prompt Inicial para Claude Code
- 10. Prompts por Fase de Desarrollo

1. VISIÓN GENERAL DEL PROYECTO

1.1 Descripción

Plataforma web educativa de acceso público que permite al profesor compartir de forma organizada el contenido pedagógico, materiales de trabajo y criterios de evaluación estructurados por unidades didácticas. Los padres acceden directamente mediante URL sin necesidad de registro ni autenticación.

1.2 Objetivos

- Ofrecer a las familias visibilidad clara sobre lo que se trabaja en clase
- Centralizar recursos educativos (vídeos, documentos, actividades)
- Comunicar criterios de evaluación de forma transparente
- Proporcionar al profesor control total sobre el contenido mediante un backoffice protegido
- Crear una experiencia más cercana y personalizada que Classroom o Sites
- Acceso inmediato para los padres: solo necesitan la URL

1.3 Usuarios del Sistema

Administrador (Profesor): Accede con credenciales al backoffice. Gestiona todo el contenido, estructura las unidades didácticas, sube materiales y define criterios de evaluación.

Usuario Final (Padres): Acceden al portal público sin autenticación. Consultan el contenido de las unidades, acceden a recursos y vídeos, revisan criterios de evaluación.

2. ARQUITECTURA TÉCNICA

2.1 Stack Tecnológico

Capa	Tecnología	Justificación
Frontend	Next.js 14 (App Router)	SSR, SEO, rendimiento
Lenguaje	TypeScript	Tipado seguro, mantenibilidad
Estilos	Tailwind CSS	Desarrollo rápido, responsive
Componentes	shadcn/ui	Componentes accesibles y personalizables
ORM	Prisma	Type-safe, migraciones automáticas
Base de datos	PostgreSQL	Robusta, relacional, gratuita en Supabase
Autenticación	NextAuth.js	Solo backoffice, credenciales
Almacenamiento	Supabase Storage	Archivos PDF, imágenes
Deploy	Vercel	Integración perfecta con Next.js

3. ESTRUCTURA DE BASE DE DATOS

3.1 Modelo de Datos

UsuarioAdmin - Profesor que administra el portal

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
email	String (unique)	Email para login
password	String	Contraseña hasheada
nombre	String	Nombre del profesor
createdAt	DateTime	Fecha de creación

Curso - Curso escolar (ej: 3º Primaria)

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
nombre	String	Nombre del curso
slug	String (unique)	URL amigable
descripcion	String?	Descripción opcional
añoAcademico	String	Ej: 2024/2025
activo	Boolean	Si se muestra en el portal

Asignatura - Materia dentro de un curso

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
cursold	String (FK)	Referencia al curso
nombre	String	Nombre de la asignatura
slug	String	URL amigable
color	String	Color hex identificativo
icono	String	Nombre del icono
orden	Int	Orden de visualización
visible	Boolean	Si se muestra en el portal

UnidadDidactica - Unidad temática

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
asignaturald	String (FK)	Referencia a la asignatura
slug	String	URL amigable
numero	Int	Número de unidad
titulo	String	Título de la unidad
descripcion	String?	Descripción (texto enriquecido)
objetivos	String?	Objetivos de aprendizaje
fechalinicio	DateTime?	Fecha de inicio
fechaFin	DateTime?	Fecha de finalización
estado	Enum	PROXIMA / ACTIVA / COMPLETADA
orden	Int	Orden de visualización
visible	Boolean	Si se muestra en el portal

Contenido - Sección de contenido dentro de una unidad

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
unidadId	String (FK)	Referencia a la unidad
titulo	String	Título del contenido
descripcion	String?	Descripción (texto enriquecido)
tipo	Enum	TEORIA / ACTIVIDAD / REPASO
orden	Int	Orden de visualización
visible	Boolean	Si se muestra en el portal

Recurso - Material asociado a un contenido

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
contenidoId	String (FK)	Referencia al contenido
tipo	Enum	VIDEO / PDF / LINK / IMAGEN / DOCUMENTO
titulo	String	Título del recurso
url	String	URL o ruta del archivo
descripcion	String?	Descripción opcional
orden	Int	Orden de visualización
metadata	Json?	Datos adicionales (thumbnail, duración...)

CriterioEvaluacion - Criterio de evaluación de una unidad

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
unidadId	String (FK)	Referencia a la unidad
codigo	String	Código del criterio (ej: CE.LCL.3.1)
descripcion	String	Descripción del criterio
peso	Float?	Peso porcentual (opcional)
competencias	String[]	Lista de competencias asociadas
indicadores	String?	Indicadores de logro

Aviso - Notificación para los padres

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
cursold	String? (FK)	Curso específico o null para todos
titulo	String	Título del aviso
mensaje	String	Contenido del aviso
tipo	Enum	INFO / IMPORTANTE / URGENTE
fechaPublicacion	DateTime	Cuándo se publica
fechaExpiracion	DateTime?	Cuándo desaparece
visible	Boolean	Si está activo

Configuracion - Configuración general del portal

Campo	Tipo	Descripción
id	String (cuid)	Identificador único
nombreColegio	String	Nombre mostrado en el portal
logo	String?	URL del logo
mensajeBienvenida	String?	Texto de bienvenida
contactoProfesor	String?	Info de contacto
colorPrimario	String	Color principal del tema

4. BACKOFFICE (PANEL DEL PROFESOR)

4.1 Autenticación

El backoffice estará protegido y solo accesible para el profesor:

- Login con email y contraseña
- Contraseña hasheada con bcrypt
- Sesión persistente con JWT
- Recuperación de contraseña por email
- Middleware de protección en todas las rutas /admin

4.2 Dashboard Principal

Pantalla inicial del backoffice con:

- URL pública del portal para copiar y compartir con los padres
- Resumen visual: número de unidades activas, completadas y próximas
- Accesos rápidos a crear nueva unidad, nuevo aviso
- Avisos activos actualmente
- Últimas modificaciones realizadas

4.3 Gestión de Cursos

CRUD completo de cursos escolares:

- Crear curso con nombre, descripción y año académico
- Generación automática de slug para URL amigable
- Activar/desactivar cursos (los inactivos no aparecen en el portal)
- Editar y eliminar cursos existentes

4.4 Gestión de Asignaturas

Dentro de cada curso, el profesor puede crear asignaturas:

- Nombre de la asignatura
- Selector de color identificativo (paleta predefinida)
- Selector de ícono (libro, calculadora, globo, música, etc.)
- Ordenación de asignaturas mediante drag & drop
- Control de visibilidad

4.5 Gestión de Unidades Didácticas

Cada unidad didáctica contendrá:

Información básica:

- Número de unidad (para ordenación)

- Título descriptivo
- Descripción general (editor de texto enriquecido con formato)
- Objetivos de aprendizaje
- Fechas de inicio y fin previstas
- Estado: Próxima / Activa / Completada
- Visible: permite preparar contenido sin publicar

Secciones de contenido:

Sección	Descripción	Ejemplos
TEORÍA	Explicaciones y conceptos	Resúmenes, esquemas, vídeos explicativos
ACTIVIDADES	Ejercicios y tareas	Fichas descargables, ejercicios online
REPASO	Material pre-examen	Resúmenes finales, vídeos de repaso

4.6 Gestión de Recursos

El profesor puede añadir diferentes tipos de recursos a cada contenido:

Tipo	Funcionalidad	Características
VIDEO	Vídeos de YouTube	Pegar URL, extracción automática de thumbnail y título
PDF	Documentos PDF	Subida de archivos hasta 10MB, visor embebido
LINK	Enlaces externos	URL a recursos educativos, extracción de favicon
IMAGEN	Imágenes	Esquemas, infografías, mapas conceptuales
DOCUMENTO	Archivos descargables	Word, fichas de trabajo imprimibles

4.7 Criterios de Evaluación

Por cada unidad, el profesor definirá los criterios de evaluación según LOMLOE:

- Código del criterio (ej: CE.LCL.3.1)
- Descripción clara y comprensible para las familias
- Peso porcentual sobre la nota de la unidad (opcional)
- Competencias asociadas (selección múltiple de lista predefinida)
- Indicadores de logro (opcional)

Competencias disponibles:

- CCL - Comunicación lingüística
- STEM - Competencia matemática y en ciencia y tecnología
- CD - Competencia digital
- CPSAA - Competencia personal, social y de aprender a aprender
- CC - Competencia ciudadana
- CE - Competencia emprendedora
- CCEC - Competencia en conciencia y expresión culturales

4.8 Gestión de Avisos

Sistema de notificaciones para comunicar información a las familias:

- Título y mensaje (con editor de texto enriquecido)
- Tipo: Informativo (azul) / Importante (naranja) / Urgente (rojo)
- Fecha de publicación (puede programarse a futuro)
- Fecha de expiración (desaparece automáticamente)
- Asociar a un curso específico o mostrar en todos
- Control de visibilidad

4.9 Configuración del Portal

Personalización del aspecto del portal público:

- Nombre del colegio o clase
- Logo (subida de imagen)
- Mensaje de bienvenida personalizado
- Información de contacto del profesor
- Color primario del tema

5. PORTAL PÚBLICO (PARA PADRES)

5.1 Características Generales

- Acceso libre sin autenticación - cualquier persona con la URL puede entrar
- Diseño responsive optimizado para móvil (los padres consultarán desde el teléfono)
- Navegación intuitiva y clara para usuarios no técnicos
- Carga rápida incluso en conexiones móviles lentas
- Accesible: contraste adecuado, tamaños de texto legibles

5.2 Estructura de URLs

Página	URL	Contenido
Home	tudominio.com	Avisos + cursos/asignaturas
Curso	tudominio.com/3-primaria	Asignaturas del curso
Asignatura	tudominio.com/3-primaria/lengua	Unidades didácticas
Unidad	tudominio.com/3-primaria/lengua/unidad-1	Contenido completo
Avisos	tudominio.com/avisos	Histórico de avisos

5.3 Página Principal (Home)

Cabecera:

- Nombre del colegio/clase
- Logo (si está configurado)
- Mensaje de bienvenida del profesor

Contenido principal:

- Banner de avisos urgentes (si los hay)
- Avisos recientes destacados
- Tarjetas de asignaturas con ícono, color e indicador de unidad activa
- Si hay múltiples cursos, mostrar selector de curso primero

5.4 Vista de Asignatura

- Cabecera con nombre, ícono y color de la asignatura
- Breadcrumb de navegación (Inicio > Curso > Asignatura)
- Listado de unidades didácticas ordenadas
- Cada unidad muestra: número, título, estado (badge de color), fechas
- Estados visuales: ■ Activa (verde), ■ Completada (gris), ■ Próxima (azul)

5.5 Vista de Unidad Didáctica

Página completa de la unidad organizada en secciones (pestañas o acordeón):

Sección 'Qué estamos trabajando':

- Descripción de la unidad
- Objetivos de aprendizaje en lista clara
- Fechas importantes

Sección 'Teoría':

- Contenidos teóricos organizados
- Vídeos YouTube con reproductor embebido
- PDFs visualizables inline
- Imágenes y esquemas ampliables

Sección 'Actividades':

- Listado de actividades
- Material descargable
- Enlaces a recursos interactivos

Sección 'Repaso':

- Material de repaso pre-examen
- Resúmenes
- Vídeos explicativos adicionales

Sección 'Evaluación':

- Criterios de evaluación claros
- Competencias que se trabajan
- Peso de cada criterio (si definido)

6. DISEÑO DE INTERFAZ

6.1 Portal Público - Lineamientos de Diseño

Principios:

- Limpio, moderno y acogedor
- Fácil de usar para personas no tecnológicas
- Mobile-first (la mayoría accederá desde móvil)
- Accesible (WCAG AA)

Paleta de colores sugerida:

Uso	Color	Hex
Primario	Azul educativo	#3B82F6
Éxito/Activo	Verde	#10B981
Advertencia	Naranja	#F59E0B
Urgente	Rojo	#EF4444
Fondo	Gris muy claro	#F8FAFC
Texto	Gris oscuro	#1E293B

Tipografía: Inter o Nunito (legibles, amigables, disponibles en Google Fonts)

6.2 Backoffice - Lineamientos

- Panel administrativo funcional y limpio
- Sidebar de navegación lateral colapsable
- Header con info del usuario y acciones rápidas
- Tablas con paginación y búsqueda
- Formularios con validación en tiempo real
- Editor WYSIWYG (TipTap o similar) para textos enriquecidos
- Drag & drop para ordenar elementos
- Modales para confirmaciones
- Toast notifications para feedback de acciones

7. PLAN DE DESARROLLO POR FASES

Fase	Descripción	Duración estimada
1	Infraestructura: Next.js, Prisma, Auth, estructura	1 día
2	Backoffice estructura: Layout, Dashboard, CRUD Cursos y Asignaturas	1 día
3	Backoffice contenido: CRUD Unidades, Contenidos, Recursos	2 días
4	Backoffice final: Criterios, Avisos, Configuración	1 día
5	Portal público: Home, vistas de curso, asignatura, unidad	2 días
6	Polish: Responsive, estados vacíos, errores, seed, deploy	1 día

Tiempo total estimado: 8 días de desarrollo

8. ESTRUCTURA DE ARCHIVOS

```
portal-educativo/
  prisma/
    schema.prisma
    seed.ts
  src/
    app/
      (public)/ # Portal público
      page.tsx # Home
      [cursoSlug]/
        page.tsx # Vista curso
        [asignaturaSlug]/
          page.tsx # Vista asignatura
        [unidadSlug]/
          page.tsx # Vista unidad
        avisos/page.tsx
      admin/ # Backoffice protegido
      page.tsx # Dashboard
      layout.tsx # Layout con sidebar
      cursos/
      asignaturas/
      unidades/
      avisos/
      configuracion/
      api/
      auth/[...nextauth]/
      cursos/
      asignaturas/
      unidades/
      contenidos/
      recursos/
      criterios/
      avisos/
      upload/
      login/page.tsx
      layout.tsx
      components/
      ui/ # shadcn/ui
      admin/
      public/
      lib/
      prisma.ts
      auth.ts
      utils.ts
      types/index.ts
  package.json
```

9. PROMPT INICIAL PARA CLAUDE CODE

Copia este prompt completo para iniciar el proyecto en Claude Code:

Necesito crear un Portal Educativo para Padres con las siguientes características:

DESCRIPCIÓN

Portal web donde un profesor comparte contenido educativo organizado por unidades didácticas.

Los padres acceden sin autenticación (solo necesitan la URL).

El profesor tiene un backoffice protegido para gestionar todo el contenido.

STACK TECNOLÓGICO

- Next.js 14 con App Router y TypeScript
- Tailwind CSS + shadcn/ui
- Prisma ORM con PostgreSQL
- NextAuth.js (solo para backoffice)
- Supabase Storage para archivos

USUARIOS

- Admin (Profesor): accede con login al backoffice /admin
- Padres: acceden al portal público sin autenticación

MODELOS DE DATOS PRINCIPALES

- UsuarioAdmin: id, email, password, nombre
- Curso: id, nombre, slug, descripcion, añoAcademico, activo
- Asignatura: id, cursoId, nombre, slug, color, icono, orden, visible
- UnidadDidactica: id, asignaturaId, slug, numero, titulo, descripcion, objetivos, fechaInicio, fechaFin, estado (PROXIMA/ACTIVA/COMPLETADA), orden, visible
- Contenido: id, unidadId, titulo, descripcion, tipo (TEORIA/ACTIVIDAD/REPASO), orden, visible
- Recurso: id, contenidoId, tipo (VIDEO/PDF/LINK/IMAGEN/DOCUMENTO), titulo, url, descripcion, orden
- CriterioEvaluacion: id, unidadId, codigo, descripcion, peso, competencias, indicadores
- Aviso: id, cursoId, titulo, mensaje, tipo (INFO/IMPORTANTE/URGENTE), fechaPublicacion, fechaExpiracion, visible
- Configuracion: id, nombreColegio, logo, mensajeBienvenida, contactoProfesor, colorPrimario

FUNCIONALIDADES BACKOFFICE

- Dashboard con URL pública y resumen de unidades
- CRUD de Cursos con slug automático
- CRUD de Asignaturas con selector de color e icono
- CRUD de Unidades con editor de texto enriquecido
- CRUD de Contenidos (teoría, actividades, repaso)
- CRUD de Recursos (YouTube embebido, PDF, enlaces, imágenes)
- CRUD de Criterios de evaluación con competencias
- CRUD de Avisos con programación y expiración
- Configuración del portal (nombre, logo, colores)

FUNCIONALIDADES PORTAL PÚBLICO

- Home con avisos y tarjetas de asignaturas
- Vista de asignatura con listado de unidades
- Vista de unidad con secciones: descripción, teoría, actividades, repaso, evaluación
- Reproductor YouTube embebido
- Visor PDF embebido
- Diseño responsive mobile-first

- URLs amigables con slugs

PRIMERA TAREA

Configura el proyecto Next.js 14 con TypeScript, Tailwind, shadcn/ui, Prisma y NextAuth.

Crea el schema.prisma completo con todos los modelos.

Configura la conexión a PostgreSQL (usar variable DATABASE_URL).

Crea el archivo de configuración de NextAuth con provider de credenciales.

Genera una estructura de carpetas organizada siguiendo las convenciones de Next.js App Router.

10. PROMPTS POR FASE DE DESARROLLO

Usa estos prompts secuencialmente para guiar a Claude Code en cada fase:

FASE 2: Backoffice - Estructura

Continúa el desarrollo del Portal Educativo.

TAREA FASE 2: Backoffice estructura básica

1. Crea el layout del backoffice /admin con:

- Sidebar lateral con navegación (Dashboard, Cursos, Asignaturas, Unidades, Avisos, Config)
- Header con nombre del usuario y botón de logout
- Área de contenido principal

2. Crea el Dashboard /admin con:

- Tarjeta con URL pública del portal (copiable)
- Estadísticas: unidades activas, completadas, próximas
- Accesos rápidos a crear unidad y crear aviso
- Lista de avisos activos

3. Implementa CRUD completo de Cursos:

- Listado con tabla (nombre, año, estado, acciones)
- Formulario de creación/edición
- Generación automática de slug
- Toggle de activar/desactivar
- Confirmación de eliminación

4. Implementa CRUD completo de Asignaturas:

- Listado filtrable por curso
- Formulario con selector de color (paleta predefinida)
- Selector de ícono (usar Lucide icons)
- Ordenación drag & drop
- Toggle de visibilidad

Usa shadcn/ui para todos los componentes.

Implementa validación con react-hook-form y zod.

Crea las API routes necesarias en /api.

FASE 3: Backoffice - Contenido

Continúa el desarrollo del Portal Educativo.

TAREA FASE 3: Gestión de contenido educativo

1. CRUD de Unidades Didácticas:

- Listado filtrable por asignatura
- Formulario con campos: número, título, descripción (editor TipTap), objetivos
- Selector de fechas inicio/fín
- Selector de estado (Próxima/Activa/Completada)
- Toggle de visibilidad
- Generación automática de slug

2. CRUD de Contenidos (dentro de cada unidad):

- Listado de contenidos de la unidad

- Crear contenido con tipo (Teoría/Actividad/Repaso)
- Editor de texto enriquecido para descripción
- Ordenación drag & drop
- Toggle de visibilidad

3. CRUD de Recursos (dentro de cada contenido):

- Añadir recurso VIDEO: input de URL YouTube, extracción de metadata (título, thumbnail)
- Añadir recurso PDF: subida de archivo a Supabase Storage
- Añadir recurso LINK: input de URL, extracción de título y favicon
- Añadir recurso IMAGEN: subida con preview
- Ordenación y eliminación de recursos

4. Sistema de subida de archivos:

- Configurar Supabase Storage
- Endpoint /api/upload para subir archivos
- Validación de tipos y tamaño (máx 10MB)
- Devolver URL pública del archivo

Mantén la consistencia visual con el resto del backoffice.

FASE 4: Backoffice - Finalización

Continúa el desarrollo del Portal Educativo.

TAREA FASE 4: Criterios de evaluación, avisos y configuración

1. CRUD de Criterios de Evaluación (por unidad):

- Listado de criterios de la unidad
- Formulario: código, descripción, peso (%), indicadores
- Selector múltiple de competencias:
 - * CCL - Comunicación lingüística
 - * STEM - Matemática y ciencia
 - * CD - Digital
 - * CPSAA - Personal y social
 - * CC - Ciudadana
 - * CE - Emprendedora
 - * CCEC - Cultural

2. CRUD de Avisos:

- Listado de avisos con filtro por tipo y estado
- Formulario: título, mensaje (editor), tipo (Info/Importante/Urgente)
- Selector de fecha publicación (programable)
- Selector de fecha expiración (opcional)
- Asociar a curso específico o todos
- Preview de cómo se verá

3. Página de Configuración:

- Formulario con: nombre colegio, mensaje bienvenida, contacto profesor
- Subida de logo
- Selector de color primario
- Preview del portal con los cambios

4. Mejoras generales:

- Toast de confirmación en todas las acciones
- Estados de carga en formularios
- Manejo de errores con mensajes claros

FASE 5: Portal Público

Continúa el desarrollo del Portal Educativo.

TAREA FASE 5: Portal público para padres

1. Layout público:

- Header con logo/nombre del colegio
- Navegación con breadcrumbs
- Footer con contacto del profesor
- Responsive: menú hamburguesa en móvil

2. Home (/) :

- Banner de avisos urgentes (si hay)
- Sección de avisos recientes
- Grid de tarjetas de asignaturas con ícono, color y unidad activa
- Si hay múltiples cursos, mostrar selector

3. Vista de Asignatura (/[cursoSlug]/[asignaturaSlug]):

- Cabecera con nombre e ícono

- Listado de unidades con:
 - * Número y título
 - * Badge de estado (Activa verde, Completada gris, Próxima azul)
 - * Fechas si existen
4. Vista de Unidad (/[cursoSlug]/[asignaturaSlug]/[unidadSlug]):
- Tabs o acordeón con secciones:
 - * "Qué trabajamos": descripción y objetivos
 - * "Teoría": contenidos tipo TEORIA con recursos
 - * "Actividades": contenidos tipo ACTIVIDAD
 - * "Repaso": contenidos tipo REPASO
 - * "Evaluación": criterios de evaluación
 - Reproductor YouTube embebido (react-player o iframe)
 - Visor PDF embebido (react-pdf o iframe)
 - Galería de imágenes con lightbox
5. Página de Avisos (/avisos):
- Listado completo de avisos
 - Filtro por tipo

Diseño limpio, moderno, mobile-first. Usa la paleta de colores definida.

FASE 6: Polish y Deploy

Continúa el desarrollo del Portal Educativo.

TAREA FASE 6: Finalización y despliegue

1. Estados vacíos y loading:
 - Componente EmptyState para listas vacías
 - Skeletons de carga en todas las páginas
 - Spinners en botones durante acciones
2. Manejo de errores:
 - Página 404 personalizada
 - Página de error genérica
 - Mensajes de error amigables en formularios
3. Responsive:
 - Revisar todas las páginas en móvil
 - Ajustar tablas del backoffice (scroll horizontal o cards)
 - Probar navegación táctil
4. SEO y metadata:
 - Títulos dinámicos por página
 - Meta descriptions
 - Open Graph tags
5. Script de seed (prisma/seed.ts):
 - Usuario admin: profesor@ejemplo.com / demo1234
 - Curso: "3º Primaria 2024/2025"
 - 4 asignaturas: Lengua, Matemáticas, Conocimiento del Medio, Inglés
 - 2 unidades por asignatura con contenido de ejemplo
 - Vídeos YouTube educativos reales
 - 3 avisos de ejemplo
6. Preparar para deploy:
 - Revisar variables de entorno (.env.example)
 - Configurar para Vercel
 - Documentar proceso de despliegue en README
7. README.md completo:
 - Descripción del proyecto
 - Requisitos
 - Instalación local
 - Variables de entorno necesarias
 - Comandos disponibles
 - Guía de despliegue

NOTAS FINALES

Recomendaciones para el desarrollo:

- Ejecuta cada fase de forma secuencial, no avances hasta completar la anterior
- Revisa el código generado antes de pasar a la siguiente fase
- Prueba las funcionalidades en el navegador después de cada fase
- Si Claude Code genera errores, pégale el mensaje de error para que lo corrija
- Guarda commits frecuentes en Git para poder revertir si es necesario
- Al finalizar, prueba todo el flujo: crear curso → asignatura → unidad → contenido → recurso

Variables de entorno necesarias:

- DATABASE_URL: URL de conexión a PostgreSQL
- NEXTAUTH_SECRET: Secreto para NextAuth (generar con openssl rand -base64 32)
- NEXTAUTH_URL: URL del sitio (http://localhost:3000 en desarrollo)
- SUPABASE_URL: URL de tu proyecto Supabase
- SUPABASE_ANON_KEY: Clave anónima de Supabase

— Fin del documento —