

Trabajo Práctico

Integrador: MeWeb

Ingeniería en Sistemas de Información

Cátedra: Programación Web

Alumno: Sanmartino Julián

2024

Índice

Índice	2
MeWeb	3
Descripción General del Proyecto	3
Características del Sitio	3
Requerimientos Funcionales Detallados	3
Requerimientos de Usuarios (Front-end)	3
Arquitectura de la Aplicación	4
Tiempos de Desarrollo	5
Mejoras a Futuro	5



MeWeb

Descripción General del Proyecto

MeWeb es una plataforma diseñada para que los usuarios técnicos puedan crear páginas y obtener un sitio web estático. Esta herramienta es ideal para crear páginas de cosas perdidas, eventos, páginas web, redes sociales, entre otros. El objetivo principal del sitio es proporcionar una solución sencilla y eficiente para la generación de sitios web estáticos personalizados.

Cabe destacar que como el nombre es inglés, se pronuncia como “miweb”.

Características del Sitio

El sitio ofrece una variedad de funcionalidades clave. Los usuarios pueden registrarse y autenticarse en la plataforma, lo que les permite acceder a un sistema de creación y edición de páginas utilizando markdown. Además, cada página creada puede generar un código QR, que los usuarios pueden descargar para su uso. El sitio también incluye un listado de páginas del usuario y la visualización de páginas públicas en un feed.

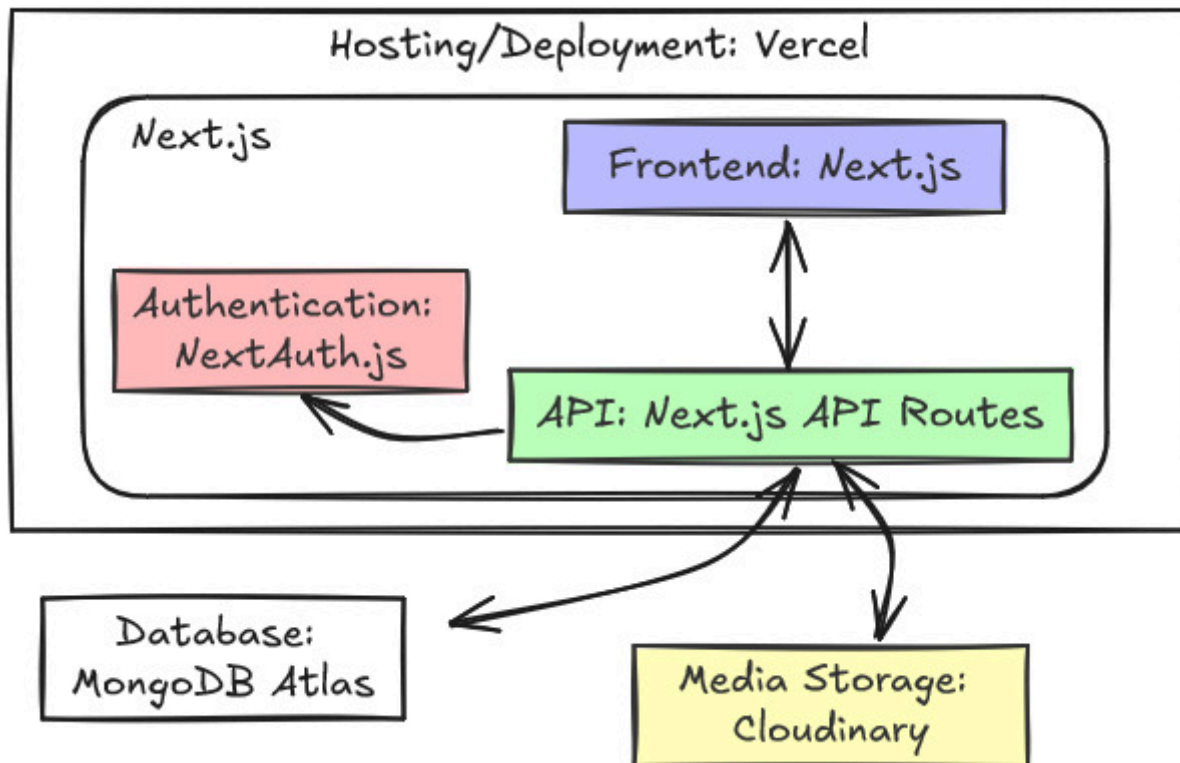
Requerimientos Funcionales Detallados

Requerimientos de Usuarios (Front-end)

La página de inicio muestra el logo y el eslogan de la empresa, junto con enlaces a las secciones más importantes del sitio. Los usuarios pueden registrarse proporcionando su correo y contraseña. Una vez registrados, pueden crear y editar páginas mediante un formulario que permite ingresar un título, una imagen y contenido en markdown. Cada página recibe automáticamente una URL corta y puede ser categorizada mediante etiquetas. Los usuarios también tienen la opción de previsualizar la página antes de guardarla.

El listado de páginas privadas permite a los usuarios ver todas las páginas que han creado, con opciones para filtrar, buscar, editar y eliminar páginas. Además, cada página puede generar un código QR que los usuarios pueden descargar. Las páginas públicas se pueden visualizar con su contenido en markdown y se puede acceder a ellas mediante las URLs cortas generadas.

Arquitectura de la Aplicación



La aplicación está construida utilizando Next.js como framework principal, lo que permite una gestión eficiente de las páginas y rutas. Los componentes de la interfaz de usuario están desarrollados con React y estilizados con TailwindCSS para asegurar un diseño responsivo y moderno. La autenticación de usuarios se maneja mediante NextAuth.js, y se utilizan componentes de interfaz de usuario de shadcn/ui.

En el backend, las rutas API de Next.js se utilizan para manejar las solicitudes, mientras que NextAuth.js gestiona el registro, inicio de sesión y cierre de sesión de los usuarios. La base de datos utilizada es MongoDB, que almacena la información de los usuarios y las páginas. Para el almacenamiento y gestión de imágenes, se utiliza Cloudinary.

La aplicación está desplegada en Vercel, una plataforma que facilita el despliegue de aplicaciones Next.js, permitiendo una integración continua y un despliegue eficiente tanto del frontend como del backend.

El flujo de datos en la aplicación está claramente definido. La autenticación de usuarios se gestiona en el frontend y se verifica en el backend. Los usuarios pueden crear, editar y eliminar páginas, cada una con una URL corta y un código QR. Las páginas se almacenan en MongoDB y las imágenes en Cloudinary. Las páginas públicas se pueden visualizar mediante URLs cortas, mientras que las páginas privadas se gestionan en el dashboard del usuario.

Esta arquitectura permite una aplicación escalable y mantenible, con una clara separación entre frontend y backend, utilizando tecnologías modernas para la autenticación, gestión de datos y despliegue.



Tiempos de Desarrollo

El tiempo utilizado para el diseño e implementación de la aplicación fue el siguiente:

- Análisis y diseño: 2 horas
- Implementación de autenticación: 2 horas
- Conexión a base de datos y CRUD de páginas: 5 horas
- Agregado de navegación y mejoras en la interfaz: 2 horas
- Corrección de errores: 1 horas

Mejoras a Futuro

Para mejorar y expandir las funcionalidades de MeWeb, se pueden considerar las siguientes mejoras a futuro:

1. **Integración con Redes Sociales:** Permitir a los usuarios compartir sus páginas directamente en redes sociales como Facebook, Twitter, y LinkedIn.
2. **Estadísticas de Visitas:** Implementar un sistema de estadísticas que permita a los usuarios ver el número de visitas a sus páginas, así como otros datos relevantes como la ubicación geográfica de los visitantes.
3. **Plantillas de Páginas:** Ofrecer una variedad de plantillas predefinidas que los usuarios puedan utilizar para crear sus páginas de manera más rápida y con un diseño profesional.
4. **Soporte Multilenguaje:** Añadir soporte para múltiples idiomas, permitiendo a los usuarios crear y visualizar páginas en diferentes idiomas.
5. **Notificaciones por Correo Electrónico:** Implementar un sistema de notificaciones por correo electrónico para informar a los usuarios sobre eventos importantes, como la creación de nuevas páginas o cambios en sus páginas existentes.
6. **Mejoras en la Seguridad:** Añadir autenticación de dos factores (2FA) para mejorar la seguridad de las cuentas de usuario.
7. **Editor de Markdown Mejorado:** Incluir un editor de markdown más avanzado con funcionalidades como la previsualización en tiempo real, resaltado de sintaxis y soporte para plugins.
8. **Integración con Servicios de Pago:** Permitir a los usuarios vender productos o servicios directamente desde sus páginas mediante la integración con servicios de pago como PayPal o Stripe.
9. **API Pública:** Desarrollar una API pública que permita a terceros integrar sus aplicaciones con MeWeb, facilitando la automatización y la creación de páginas.