

Plan de Integración Back-end y Front-end para la Plataforma de Aprendizaje para Todos

Autor	Versión	Fecha
Neira Carquin, Rogger Kevin	V1.0	7 sept 2024

1. Introducción	1
2. Arquitectura General	1
3. Estrategia de Integración	1
3.1. API RESTful	1
3.2. Endpoints principales	2
3.3. Autenticación	2
4. Integración por Módulo	2
4.1. Gestión de Usuarios	2
4.2. Gestión de Recursos Educativos	2
4.3. Gestión de Foros	2
4.4. Funcionalidades de Administración	3
5. Manejo de Estados y Datos	3
6. Seguridad	3
7. Pruebas	3
8. Despliegue	3
9. Monitoreo y Logging	3
10. Conclusión	3

1. Introducción

En este documento se explica el plan inicial para integrar la funcionalidad del back-end con el front-end de la Plataforma de Aprendizaje para Todos. El objetivo es asegurar una comunicación fluida entre ambas partes del sistema, dando así una experiencia amigable y eficiente al usuario.

2. Arquitectura General

Para esta plataforma, se utilizará una arquitectura de tres capas

1. Presentación (Front-end): React
2. Aplicación (Back-end): Node.js con Express.js
3. Persistencia: MongoDB

3. Estrategia de Integración

3.1. API RESTful

Se va a utilizar una API RESTful empleando Express.js en el back-end. Esta API funcionará como conector entre el front-end y la base de datos, gestionando todas las operaciones CRUD (Crear, Leer, Actualizar, Eliminar)

3.2. Endpoints principales

- `/api/users` : Gestión de usuarios(registro, autenticación, actualización de perfil)
- `/api/resources` : Gestión de recursos educativos.
- `/api/forums` : Gestión de foros y discusiones.
- `/api/forums` : Funcionalidades específicas del administrador.

3.3. Autenticación

Se empleará JSON Web Tokens(JWT) para la autenticación de usuarios. El back-end va a generar y verificar estos tokens, mientras que el front-end los va a almacenar y enviar con cada solicitud que requiera autenticación.

4. Integración por Módulo

4.1. Gestión de Usuarios

- Back-end:
 - Implementar rutas para el registro, login y actualizaciones de perfil.
 - Validar datos de entrada
 - Generar y verificar JWT
 - Interactuar con la base de datos MongoDB
- Front-end
 - Crear formularios de registro y login
 - Implementar gestión de estado de usuario (usando Context API o Redux)
 - Almacenar JWT en localStorage
 - Mostrar información de perfil y permitir su edición

4.2. Gestión de Recursos Educativos

- Back-end
 - Implementar rutas para CRUD de recursos educativos.
 - Validar los permisos de usuario.
 - Interactuar con la base de datos MongoDB
 - Integrar con los servicios de GCP para almacenamiento de archivos.

- Front-end
 - Crear interfaces para buscar, ver y el perfil docente, subir recursos.
 - Implementar visualización de diferentes tipos de recursos (libros, videos, etc.)
 - Integrar con la API de GCP para la carga de archivos

4.3. Gestión de Foros

- Back-end
 - Implementar rutas para CRUD de foros, temas y respuestas
 - Validar los permisos de usuario
 - Implementar un sistema de notificaciones en tiempo real
- Front-end
 - Crear interfaces para ver, crear y responder en el foro
 - Implementar actualización en tiempo real de las nuevas respuestas
 - Mostrar notificaciones de las nuevas actividades en los foros

4.4. Funcionalidades de Administración

- Back-end
 - Implementar rutas para aprobación de recursos, gestión de usuarios, etc.
 - Validar permisos de administrador
 - Implementar lógica de negocio para tareas administrativas
- Front-end
 - Crear interfaz de administración
 - Implementar flujos de trabajo para aprobación de recursos
 - Mostrar estadísticas y reportes

5. Manejo de Estados y Datos

- Utilizar React Context API o Redux para el manejo del estado global en el front-end
- Implementar caché en el cliente para mejorar el rendimiento
- Utilizar mongoose en el back-end para modelar y validar datos antes de la persistencia en MongoDB

6. Seguridad

- Implementar HTTPS para todas las comunicaciones
- Usar bcrypt para el hash de contraseñas en el back-end
- Implementar validación y sanitización de datos tanto en el front-end como en el back-end.
- Utilizar CORS en el back-end para controlar el acceso a la API

7. Pruebas

- Implementar pruebas unitarias para el lado back-end (usando Jest)
- Implementar pruebas de integración para la API
- Realizar pruebas de usuario en el lado front-end
- Implementar pruebas E2E (usando Cypress)

8. Despliegue

- Utilizar Docker para tener un sistema de contenedores en la aplicación
- Desplegar el back-end y la base de datos en Google Cloud Platform
- Desplegar el front-end en Firebase Hosting

9. Monitoreo y Logging

- Implementar logging en el back-end

- Utilizar Google Cloud Monitoring para monitorear el rendimiento y la salud del sistema
- Implementar manejo de errores y reporte de excepciones tanto en el front-end como en el back-end

10. Conclusión

Este plan nos da una visión general de la ruta que se tiene planeada para la integración efectiva del back-end y el front-end de la Plataforma de Aprendizaje para Todos. Al seguir estas directrices, se planea asegurar una experiencia de usuario fluida y una plataforma robusta y escalable.