**DOCUMENTACIÓN INTEGRACIÓN**

**PONENTES**

**JUAN DAVID SERRANO**

**MIGUEL ANGEL ROMERO**

**LAURA GAONA CASTAÑO**

**INGENIERIA DE SOFTWARE**

**UNIVERSIDAD MANUELA BELTRAN**

**2025**

**1. Introducción**

Esta documentación describe el proceso de integración del sitio web de la tienda online de libros **Entrelineas**, centrándose en la conexión entre el **FrontEnd, BackEnd y la base de datos**, además de la gestión del control de versiones con **Git**. La correcta integración de estos elementos es esencial para garantizar la estabilidad, eficiencia y sincronización del sistema, facilitando el desarrollo colaborativo.

**2. Estructura del Proyecto**

El repositorio del proyecto se organiza en distintas ramas, cada una con un propósito específico:

* **FrontEnd:** Contiene los archivos HTML, CSS y JavaScript, utilizando frameworks como **React** o **Vue.js** para la construcción de la interfaz de usuario.
* **BackEnd:** Incluye la lógica del servidor desarrollada en **Node.js con Express.js**, así como la configuración y conexión con la base de datos.
* **Documentación:** Almacena toda la información relevante sobre el desarrollo y la estructura del proyecto.
* **Integración:** Es la rama donde se unen todas las partes del sistema y se realizan pruebas antes del despliegue de nuevas versiones.

**3. Pruebas de Integración**

Antes de lanzar una nueva versión, se ejecutan pruebas en la rama de integración para validar el correcto funcionamiento del sistema. Algunas pruebas clave incluyen:

* **Verificación de la comunicación entre FrontEnd y BackEnd:** Se comprueba que las solicitudes y respuestas sean correctas.
* **Validación del almacenamiento de datos:** Se revisa que la base de datos almacene y recupere información correctamente.
* **Revisión del control de versiones con Git:** Se garantiza que todas las modificaciones se integren sin conflictos y de manera organizada.

**4. Pruebas con Postman**

Para validar la funcionalidad del BackEnd, se usa **Postman**, una herramienta que permite realizar pruebas sobre APIs RESTful de manera eficiente. A través de Postman, se ejecutan solicitudes **HTTP (GET, POST, PUT, DELETE)** para verificar el correcto funcionamiento del servidor antes de integrarlo con el FrontEnd.

**Ventajas del uso de Postman:**

* Permite probar el BackEnd sin depender del FrontEnd.
* Facilita la identificación y corrección de errores en la comunicación con la base de datos.
* Posibilita la automatización de pruebas mediante colecciones de solicitudes.

**5. Integración de APIs y Comunicación entre Componentes**

Para conectar la base de datos con el BackEnd, se desarrolla una **API REST** en **Node.js con Express.js**, encargada de gestionar autenticaciones, validación de datos y operaciones **CRUD**.

Para la conexión entre el FrontEnd y el BackEnd, se usa la función **fetch()** de JavaScript, que permite el envío y recepción de datos de manera asincrónica. Esto es crucial para un flujo de información eficiente y una mejor experiencia de usuario.

**Consideraciones clave en la integración:**

* Configuración de solicitudes HTTP desde el FrontEnd al BackEnd.
* Manejo adecuado de respuestas para actualizar dinámicamente la interfaz.
* Implementación de **JWT (JSON Web Token)** para una autenticación segura.

Una integración efectiva entre el FrontEnd y el BackEnd es fundamental para el buen funcionamiento del sistema. El uso de herramientas como **Postman** para pruebas de API y **fetch()** para la comunicación asincrónica optimiza la validación de la funcionalidad antes del despliegue.

A continuación de muestran los diagramas en la base de datos y La utilización de API’s:

