**DOCUMENTACION INTEGRACION**

**JUAN DAVID SERRANO**

**MIGUEL ANGEL ROMERO**

**LAURA GAONA CASTAÑO**

**INGENIERIA DE SOFTWARE**

**WebSite Entrelineas**

**1. Introducción**

Esta documentación describe el proceso de integración del sitio web de la tienda online de libros, enfocándose en la comunicación entre frontEnd, backEnd y base de datos, así como el control de versiones con Git.

**2. Estructura del Proyecto**

El repositorio está organizado en las siguientes ramas:

* **FrontEnd**: Contiene los archivos HTML, CSS y JavaScript utilizando sus diferentes Frameworks.
* **Backend**: Contiene la lógica del servidor y la conexión a la base de datos.
* **Documentación**: Contiene los archivos de documentación del proyecto.
* **Integración**: Rama encargada de unir las diferentes partes del proyecto y realizar pruebas de funcionamiento.

**3. Pruebas de Integración**

Antes de realizar despliegues, se deben realizar pruebas en la rama de integración:

* Verificar la comunicación entre frontEnd y backend.
* Comprobar que los datos se almacenan correctamente en la base de datos.
* Revisar que el control de versiones mantiene el código organizado.

**POST MAN:**

Para las pruebas del Back-End usaremos la herramienta Postman para probar API RESTful , entonces usaremos postman para realizar peticiones HTTP y asi mismo obtener una respuesta sin tener aun la interfaz de usuario (Front-End) entonces eso quiere decir que podemos empezar a hacer pruebas al backend sin necesidad de depender del grupo de frontend.

El proceso de pruebas con Postman nos permitirá como grupo de integracion depurar errores en la comunicacion con la base de datos antes de conectar con el backend con el frontEnd.

**APIS DE TESTING:**

Para la integracion de la base de datos con el backend , se usara una api rest en node.js para manejar autenticaciones , validar datos y realizar operaciones sobre la db.

Para la inegracion de frontend con backen tenemos pensado usar la API fetch() de javascript , que permite enviar y recibir datos de maneta asincronica. Entonces buscamos que al momento de la integracion quede correctamente , el envio de peticiones HTTP dese el front al backend , el manejo de respuestas para actualizar la interfaz de usuario y el almacenamiento de tokens JWT para la autenticacion y peticiones al backend.

**USO DE FETCH:**

La correcta integración entre frontend y backend asegura una comunicación fluida en la aplicación. Postman facilita la prueba de los endpoints antes de la conexión con la interfaz de usuario, mientras que el uso de fetch() en JavaScript permite consumir la API REST, garantizando una interacción dinámica y segura entre ambas capas del sistema.