**Documentación Técnica de la Biblioteca Operaciones Matemáticas y Aplicación Web CRUD**

**1. Arquitectura de la Aplicación**

**1.1 Descripción General**

La biblioteca **OperacionesMatematicas** es una aplicación de código Java que ofrece funcionalidades para realizar operaciones matemáticas básicas, como suma, resta, multiplicación y división. La estructura del proyecto sigue una arquitectura simple y organizada, utilizando el paquete **com.biblioteca.matematica** para mantener el código bien modularizado y fácil de reutilizar.

La **Aplicación Web CRUD** es una aplicación web desarrollada con **Java Spring Boot** para el backend y **React** para el frontend. La aplicación permite realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre productos. Los datos se almacenan en una base de datos **MySQL** y se accede a ellos mediante **Hibernate/JPA** para la persistencia. La interacción del usuario se realiza a través de una interfaz web moderna que consume los endpoints REST proporcionados por el backend.

**1.2 Estructura del Proyecto**

**Biblioteca Operaciones Matemáticas**

El proyecto está organizado en una estructura de directorios estándar:

Texto

Descripción generada automáticamente

* **src**: Contiene el código fuente.
  + **com.biblioteca.matematica**: Paquete donde se encuentran las clases de la biblioteca.
* **bin**: Contiene los archivos .class generados tras la compilación.

**Aplicación Web CRUD**

La aplicación web tiene la siguiente estructura:

Imagen que contiene Diagrama

Descripción generada automáticamente

* **backend**: Contiene el servidor desarrollado en **Spring Boot**.
  + **controller**: Maneja las solicitudes HTTP y define los endpoints para las operaciones CRUD.
  + **model**: Define la entidad Producto.
  + **repository**: Interfaz para la interacción con la base de datos usando JPA.
  + **service**: Lógica del negocio para la gestión de los productos.
* **frontend**: Contiene la interfaz de usuario desarrollada con **React**.
  + **components**: Componentes individuales de la interfaz, como agregar y listar productos.

**1.3 Decisiones de Diseño**

* **Biblioteca Matemática con Métodos Estáticos**: Se decidió implementar métodos estáticos para que las funciones matemáticas puedan ser utilizadas sin necesidad de instanciar objetos. Esto facilita el uso de la biblioteca y la hace más eficiente para realizar cálculos rápidos.
* **Arquitectura RESTful para la Aplicación CRUD**: La aplicación web está diseñada con una arquitectura **RESTful**, lo cual permite una separación clara entre la lógica del backend y la presentación del frontend. Esto facilita la integración, pruebas y mantenimiento.
* **Persistencia con Hibernate/JPA**: Se eligió **Hibernate** para la gestión de la persistencia porque facilita la interacción con la base de datos a través de JPA, simplificando las consultas y las operaciones CRUD.
* **Frontend React**: Se utilizó **React** para el frontend debido a su capacidad para crear interfaces de usuario reactivas y modulares, lo cual mejora la experiencia de usuario.

**2. Instrucciones de Instalación y Ejecución**

**2.1 Requisitos Previos**

* **Java Development Kit (JDK)**: Versión 11 o superior.
* **Node.js y npm**: Para ejecutar el frontend en React.
* **MySQL**: Para la base de datos del backend.
* **Visual Studio Code (VS Code)** con las extensiones **Java Extension Pack** y **Node.js Extension Pack** instaladas.

**2.2 Instalación del Proyecto**

**Biblioteca Operaciones Matemáticas**

1. **Clonar o Crear el Proyecto**: Puedes crear manualmente la estructura del proyecto o clonar el repositorio si se encuentra alojado en una plataforma de control de versiones como GitHub.
2. **Configurar el Entorno**:
   * Asegúrate de tener **JDK** instalado y configurado en tu sistema.
   * Abre el proyecto en **VS Code**.

**Aplicación Web CRUD**

1. **Clonar el Proyecto**: Clona el repositorio o descarga los archivos del proyecto.
2. **Backend**:
   * Configura la base de datos **MySQL** y asegúrate de actualizar las credenciales en application.properties.
   * Abre el proyecto en VS Code y asegúrate de tener el **JDK** instalado.
3. **Frontend**:
   * Asegúrate de tener **Node.js** y **npm** instalados.
   * Navega a la carpeta frontend y ejecuta el siguiente comando para instalar las dependencias:



**2.3 Compilación y Ejecución del Proyecto**

**Biblioteca Operaciones Matemáticas**

1. **Abrir la Terminal en VS Code**:
   * Abre la terminal (Ctrl + ) desde la carpeta del proyecto.
2. **Compilar las Clases**:
   * Navega a la carpeta src y ejecuta el siguiente comando:



* + Esto compilará las clases y pondrá los archivos compilados en la carpeta bin.

1. **Ejecutar la Clase Main**:
   * Desde la terminal, ejecuta la clase Main para probar la biblioteca:



**Aplicación Web CRUD**

1. **Ejecución del Backend**:
   * Navega a la carpeta del backend y ejecuta:



* + Esto levantará el servidor en http://localhost:8080.

1. **Ejecución del Frontend**:
   * Navega a la carpeta frontend y ejecuta:



* + Esto levantará la aplicación en http://localhost:3000.

**2.4 Empaquetar la Biblioteca Matemática en un Archivo JAR**

Si deseas reutilizar la biblioteca en otros proyectos, puedes empaquetarla en un archivo JAR:

1. **Crear el JAR**:



1. **Utilizar el JAR en Otros Proyectos**:
   * Copia el archivo OperacionesMatematicas.jar en el proyecto donde desees usar la biblioteca.
   * Añádelo como dependencia y utilízalo importando com.biblioteca.matematica.OperacionesMatematicas.

**3. Consideraciones Finales**

* **Manejo de Excepciones**: En el método division de la biblioteca, se lanza una excepción ArithmeticException si el divisor es cero. Esto asegura que no se produzcan errores de división indefinida durante la ejecución.
* **Modularidad y Reutilización**: Tanto la biblioteca matemática como la aplicación web están diseñadas para ser modulares y fácilmente reutilizables.
* **Separación del Backend y Frontend**: La arquitectura RESTful permite una clara separación entre el backend (servidor) y el frontend (interfaz de usuario), lo cual facilita el mantenimiento y escalabilidad del proyecto.