

Taller Maven, Heroku, Spark y Git Hub

Autor: Daniel Felipe Alfonso Bueno¹

¹Escuela Colombiana de Ingenieria Julio Garavito

*Arquitectura Empresarial Laboratorio Dos

18 de agosto de 2020

Resumen

En este proyecto, podremos realizar dos operaciones muy importantes para el análisis de datos, como lo son el cálculo de la media y la desviación estándar de estos datos. Mediante la ayuda de un servicio Web implementando en Spark, un microframework para la creación de aplicaciones web con JAVA .

1. Diseño

La intención de Sparks es proporcionar una alternativa para los desarrolladores de Kotlin / Java que desean desarrollar sus aplicaciones web de la manera más expresiva posible y con un mínimo de repetición. Con una filosofía clara, Spark está diseñado no solo para hacerlo más productivo, sino también para mejorar su código bajo la influencia de la sintaxis elegante, declarativa y expresiva de Spark.

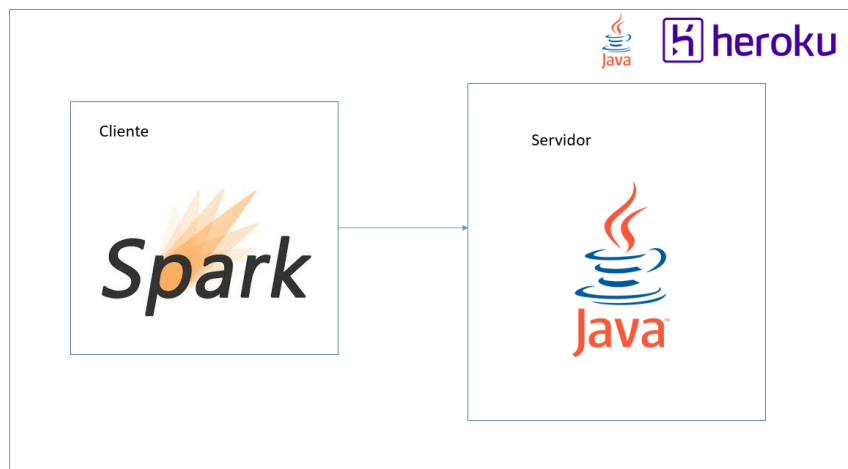


Figura 1: Diagrama de Componentes

La aplicación estará desplegada en un contenedor de Heroku la parte de cliente fue desarrollada con Spark (Frontend) y haciendo uso de plantillas de Thymeleaf la parte del servidor en JAVA(Backend) y el cliente consume los servicios de backend

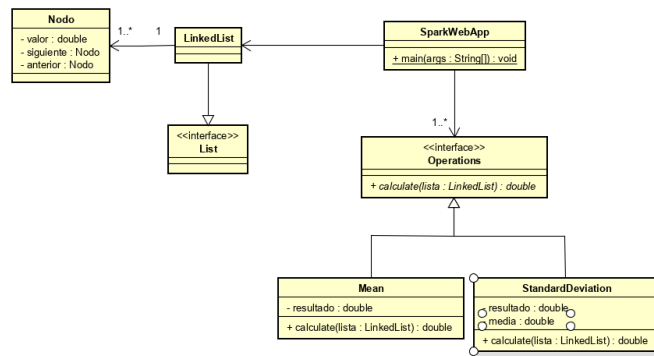


Figura 2: Diagrama de Clases

1.1. Descripción

1.1.1. Interfaz

- **Operations:** En esta interfaz contaremos con todas las operaciones que queramos realizar, simplemente debemos implementar esta para la realización del método `calculate()` que será lo que las subclases implementaran diferente.
- **List:** Interfaz nativa de Java, la cual implementaremos, para poder integrar nuestra implementación de lista enlazada en la API de Java

1.1.2. Clases

- **LinkedList:** Implementación de lista enlazada relacionada con nodos ya que este se compone de uno o muchos nodos. Tendremos dos atributos nodo los cuales apuntaran al nodo inicio y nodo fin. esta implementa la interfaz `List` para así poderla integrar a la API de Java
- **Nodo** Cada nodo tendrá un valor, un nodo siguiente y un nodo anterior para movernos en nuestra lista
- **StandardDeviation:** Esta clase implementa la interfaz `operations`, sobrescribe el método `calculate`, para el cálculo de la desviación estándar recibe por constructor un atributo `media`.
- **Mean:** Clase que implementa la interfaz `operations`, sobrescribe el método `calculate` para el cálculo de la media.
- **SparkWebApp:** Clase principal denominada fachada, la cual es la encargada de la implementación de los clientes (Servicio Web) mediante el uso de plantilla de Thymeleaf

1.2. Conclusiones

Spark Web es un microframework ligero que permitirá el desarrollo de aplicaciones web sin la necesidad de depender de un servicio de frontend en particular, algunas empresas ya están haciendo uso de Spark por su alto rendimiento en sus micro servicios. Los micro servicios funcionan mejor con micro frameworks, y Spark tiene su API REST lista para servir JSON en menos de diez líneas de código. Spark se utiliza principalmente para crear API REST. Tiene compatibilidad con lenguajes de programación importantes en el mercado como Javascript (Typescript), Java, Kotlin