

# Arquitectura de un web services

## Arquitectura Empresarial

Nicolás D. Cárdenas G.<sup>1</sup>

<sup>1</sup> Bogotá D.C., Escuela de ingenieros, Colombia

Fecha: 30/01/2020

**Resumen**— En este artículo desarrollaremos los conceptos de un web server, además de dar la explicación explícita de la implementación de un servicio web con un servicio que calcula la desviación estándar y media aritmética, también pasaremos con las pruebas realizadas.

**Palabras clave**— Web Server, Spark, Api, media, desviación estándar.

**Abstract**— In this article we will develop the concepts of a web server, in addition to giving the explicit explanation of the implementation of a web service with a service that calculates the standard deviation and arithmetic mean, we will also pass through the tests performed.

**Keywords**— Web Server, Spark, Api, Mean, Estandar desviacion.

## INTRODUCCIÓN

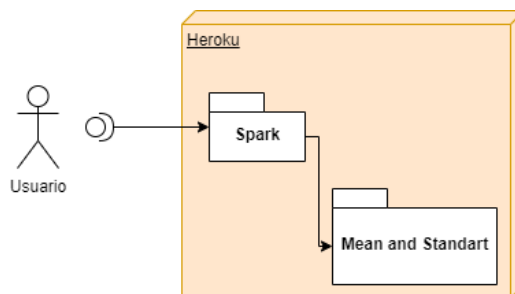
El Paper esta enfocado en la explicación de la arquitectura realizada en el este proyecto.

Se enfoca en realizar una aplicación web que tome datos y realice dos cálculos aritméticos estadísticos el primero es la media y el segundo es la desviación estándar.

Esto con el fin de poder facilitar el calculo de estos dos valores y también mostrar como es su arquitectura interna.

## CONTENIDO

Como veremos en la siguiente figura 1, notamos que la arquitectura es simple, definida por dos componentes una parte realizada con Spark Wendel (2020) y otra con un desarrollo pasado.



**Fig. 1:** Esquema de la arquitectura de la aplicación montada en Heroku.

Con Spark se configuro el api para tener 2 endpoints principales, el primero es /calc que este retorna un documento HTML para la captura de datos.

## Hello Heroku

press the button to go to math module:

**Fig. 2:** Pantalla Principal en Heroku

Como vimos en la figura 2 se puede ver una interfaz simple de bienvenida a la aplicación.

## Pruebas

**TABLA 1:** TABLA DE DATOS DE PRUEBA

Estimate Proxy Size	Development Hours
160	15.0
591	69.9
114	6.5
229	22.4
230	28.4
270	65.9
128	19.4
1657	198.7
624	38.8
1503	138.2

## Data Inputs



**Fig. 3:** Pantalla de captura de datos.

En esta figura 3 vemos una pantalla donde capturaremos los datos, consiste en un input con el cual tenemos que poner separado por comas los números con los cuales queremos realizar la operación de la media y la desviación estándar.

**TABLA 2:** TABLA DE COMPARACIÓN DE LOS DATOS ESPERADOS CON LOS CALCULADOS.

Test	Expected Value	Actual Value
	Mean - Std. Dev	Mean - Std. Dev
Table 1: Column 1	550.6 - 572.03	550.6 - 572.0268
Table 1: Column 2	60.32 - 62.26	60.32 - 62.2558

De la tabla 2 podemos observar que los resultados calculados con la aplicación Cardenas (2020)

## CONCLUSIONES

Para concluir recopilaré que el api desarrollado con Spark para el lenguaje de Java Oracle (2020), es versátil en la creación de endpoints funcionales con definiciones cortas, también permite la integración de clases y apis de forma muy simple.

## REFERENCIAS

- [1] Cardenas, N. (2020). "Calculate mean and standard". Tomado de <https://protected-atoll-31088.herokuapp.com> (30/01/2020).
- [2] Oracle (2020). "Java™ platform, standard edition 8 api specification". Tomado de <https://docs.oracle.com/javase/8/docs/api/> (30/01/2020).
- [3] Wendel, P. (2020). "Documentation - spark framework: An expressive web framework for kotlin and java". Tomado de <http://sparkjava.com/documentation> (30/01/2020).

## Results

**Mean: 2.0**

**Standard: 1.0**

**Fig. 4:** Pantalla de visualización de los resultados.

En esta pantalla visualizaremos los resultado de los datos anteriormente ingresados.