

DOCUMENTACIÓN

Taller de Introducción a Heroku

Andrés Felipe Marcelo Rubiano

| | |
|--|----------|
| Introducción | 2 |
| Estructura, diseño y arquitectura del programa. | 3 |
| Estructura | |
| Diseño | |
| Arquitectura | |
| Pruebas | 6 |
| Conclusión | 7 |

Introducción

En el siguiente documento se presentará el diseño, estructura y arquitectura para el Taller de Introducción a Heroku.

El objetivo de este escrito es presentar evidencias acerca de los nuevos conceptos y prácticas que se adquirieron durante la realización de esta actividad, además, documentar la estructura, diseño y arquitectura del entregable para el entendimiento del lector.

La computación en la nube actualmente se ha vuelto una tendencia

A continuación se presentará un listado de definiciones que se tuvieron en cuenta para el desarrollo del programa:

Conceptos Clave

- **Lista Enlazada:** Una lista enlazada es una estructura de datos lineal, en la que los elementos no se almacenan en ubicaciones de memoria contiguas. Los elementos de una lista enlazada también pueden llamarse nodos y cada nodo puede estar vinculado con otros nodos a su derecha, izquierda o ambos. [GeeksForGeeks](#)
- **Desviación estándar:** Es una medida estadística que indica que tan dispersos están los datos de la media. [López](#)
- **Media:** La media es el valor promedio de un conjunto de datos. [Minitab](#)
- **Heroku:** Heroku es una plataforma como servicio (PaaS) que permite a los desarrolladores construir, ejecutar y operar aplicaciones completamente en la nube. [Heroku](#)

Estructura, diseño y arquitectura del programa.

ESTRUCTURA

La estructura principal del proyecto "CalculatorWebApp" consta de los siguientes archivos, además se puede apreciar el **contexto** de cada una de ellos en la figura 1.

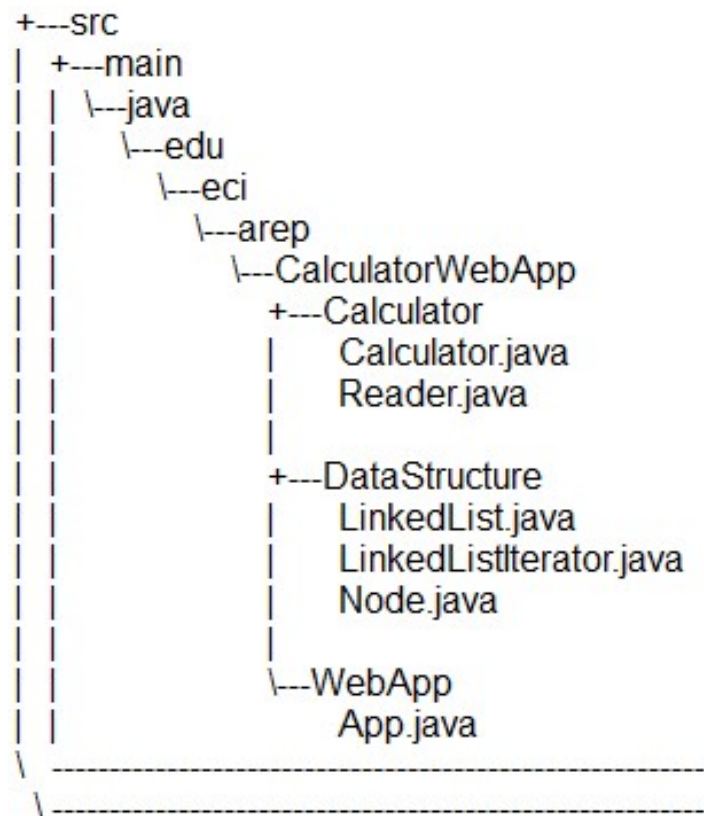


Figure 1: Estructura del Proyecto

El paquete WebApp contiene la clase principal donde se implementan los metodos necesarios para hacer funcional la pagina web con spark. En DataStructure se encuentra implementado una LinkedList que servirá para la calculadora. En el paquete calculator se encuentra la clase que ofrece las operaciones necesarias de Media y desviación estándar, y la clase READER es aquella que lee los datos, los valida y retorna la respectiva solución.

DISEÑO

El diagrama de clases del programa se presenta en la figura 2.

Uno de los elementos principales de una lista encadenada son sus nodos, es por eso que se creó una clase específica para representarlos e instanciarlos en la clase 'LinkedList'.

La lista encadenada fue realizada de forma que pueda ser compatible con la API de Java **Collections**. Para esto se decidió extender la clase '**AbstractCollection**' en la cual se implementaron algunos de los métodos que facilitan las operaciones básicas de una lista encadenada, tales como **add, remove y size**. y también añadir algunos Getters y Setters.

Además 'AbstractCollection' requería que se implementara un método **iterator()** por lo cual fué necesario crear un Objeto que implementara la interfaz "**Iterator**" cuya funcionalidad es habilitar las itera-

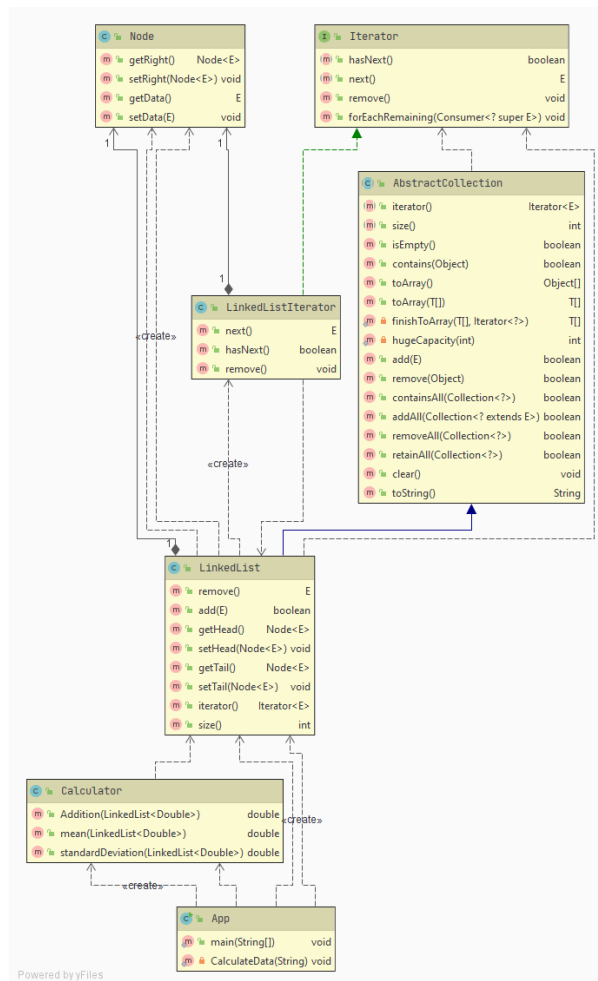


Figure 2: Diagrama de Clases

ciones con **ciclos "for"** sobre la lista encadenada a través de los nodos.

La aplicación principal (**App**) instancia un lector de datos (**Reader**) el cual toma los datos, hace los llamados necesarios y retorna una lista estática con 2 posibles resultados: si la entrada es válida, retorna el input, la media y la desviación estándar, si no, envía un string con 'INVALID INPUT' para informar al usuario, Reader hace uso de una calculadora (**Calculator**) la cual contiene las operaciones disponibles (**Addition, Mean, Standard Deviation**), esta a su vez instancia la lista encadenada para almacenar los datos de entrada y realizar las respectivas operaciones.

Debido a que la aplicación está diseñada para ser desplegada en la web, se usa el micro-framework **spark**, es usado en la clase principal **App**, allí se definen los métodos que tratarán los request que lleguen al servidor y retornará la información en forma de HTML.

ARQUITECTURA

La arquitectura desarrollada en este proyecto es una arquitectura por **descomposición modular** en donde se dividió el problema en distintos componentes para implementar la lista encadenada, para realizar el comportamiento de la aplicación principal y presentar todas las funcionalidades que se mencionaron anteriormente en el diseño del programa.

A continuación empezaremos a mencionar las partes del proyecto en donde es posible aplicar las **tres abstracciones fundamentales** de los sistemas, como ayuda se presenta la figura 3.

Como estructura lógica podemos denotar como medio de **almacenamiento** a la lista enlazada, pues es aquella que guarda la información (**WRITE**) y luego es usada para sus cálculos (**READ**). Además

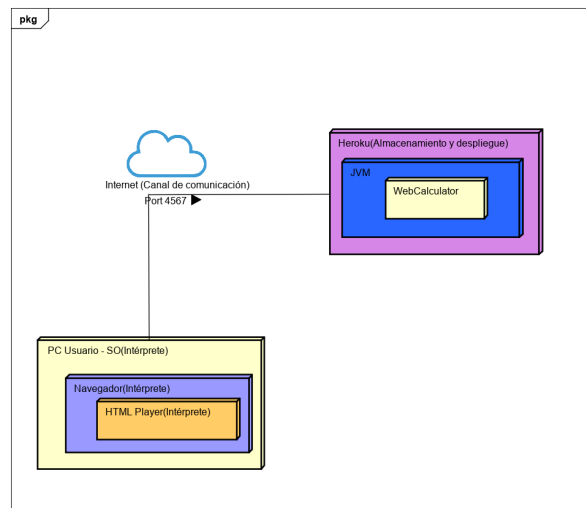


Figure 3: Representación abstracciones

también se tiene en cuenta al servidor de heroku que contiene la información de la aplicación y sus condiciones de ejecución.

Debido a que es un servicio web, y tal como se mencionó anteriormente la aplicación se desplegó en un servidor web de Heroku, se puede decir que el proyecto maneja una arquitectura cliente-servidor. En este punto podemos encontrar el primordial **canal de comunicación** con el usuario de este proyecto: El internet.

Los **interpretes** están de lado del cliente y del servidor: por parte del cliente se encuentra el navegador(Chrome, Firefox, Edge, Opera ...) que se está usando para consumir los servicios y por parte del servidor, está el lenguaje en el cual está desarrollada la aplicación (JAVA) y aquel que se encarga de compilar y ejecutar la aplicación (JVM).

A continuación se presenta un diagrama que representa la arquitectura del programa:

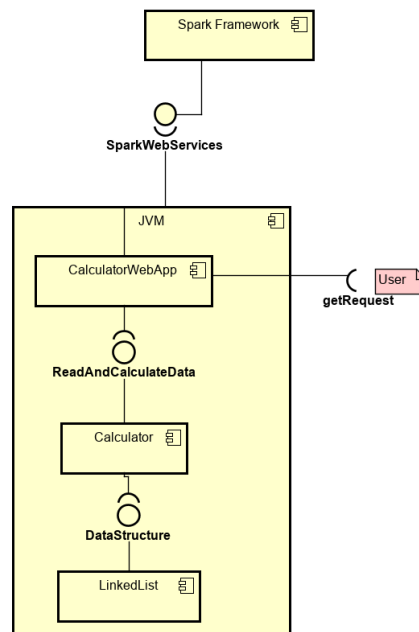


Figure 4: Arquitectura del Proyecto

Pruebas

El programa tiene algunos casos de prueba para la lista enlazada y para los calculos en general(Si quieres realizar tus propios casos de prueba visita: Taller 1 Arep), sin embargo, a continuación se presentará evidencia de los resultados teniendo en cuenta nuevos casos registrados en la siguiente tabla y una captura de pantalla de los valores retornados por el programa en el ambiente web. donde se evidencie un caso con un input inválido y otro donde se vea la salida correcta del programa

| Prueba | v1 | v2 | v3 | v4 | v5 | media | Desviación E |
|------------------|-------|-----|-------|-----|----|-------|-----------------|
| 1(Invalid Input) | Hello | 878 | 7.8.7 | 222 | 0 | N/A | N/A |
| 2.2 | 7.5 | 78 | 5 | 2.9 | 22 | 19.6 | 29.521314333884 |

Los resultados de estos datos fueron calculados en una calculadora web Arcidiacono

A continuacion se muestra la salida del programa:

← → ↺ 🏠 desolate-fjord-84057.herokuapp.com/inputdata

AREP Mean and Standard deviation Calculator

Only Integers and float(.) separated by "," (No [space] and letters allowed):

If you click the "Submit" button, the form-data will be sent to a page called "/results".

← → ↺ 🏠 desolate-fjord-84057.herokuapp.com/results?data=Hello%2C87

Your Input:

- Invalid Input!

Mean

- N/A

Standard Deviation

- N/A

[Try Another set of numbers!](#)

1 Test Case

← → ↺ 🏠 desolate-fjord-84057.herokuapp.com/inputdata

AREP Mean and Standard deviation Calculator

Only Integers and float(.) separated by "," (No [space] and letters allowed):

If you click the "Submit" button, the form-data will be sent to a page called "/results".

← → ↺ 🏠 desolate-fjord-84057.herokuapp.com/results?data=

Your Input:

- 2.2,7.5,78,5,2.9,22

Mean

- 19.6

Standard Deviation

- 29.521314333884256

[Try Another set of numbers!](#)

2 Test Case

Conclusión

Este taller-práctica fué realmente importante para retomar los conceptos básicos e implementar un proyecto con las tecnologías más usadas actualmente además de repasar el deploy con heroku, el framework spark y los nuevos conceptos aprendidos en clase acerca de las abstracciones de los sistemas.

References

G. Arcidiacono. Calculadora estadística. <http://www.alcula.com/es/calculadoras/estadistica/desviacion-estandar/>.

GeeksForGeeks. Linked list data structure. <https://www.geeksforgeeks.org/data-structures/linked-list>.

Heroku. What is heroku? <https://www.heroku.com/what>.

J. F. López. Media. <https://economipedia.com/definiciones/media.html>.

Minitab. ¿qué es la desviación estándar? <https://economipedia.com/definiciones/media.html>.