

# INTRODUCTION TO COMPUTER SYSTEM DESIGN

Nicolás Bernal Giraldo

Febrero 2021

## 1 Introducción

En este documento se describirá el diseño utilizado para resolver el problema planteado en el segundo laboratorio de AREP, el cual consta de que a través de una página web, se ingresen  $n$  numeros enteros, y con estos se calculará su media y desviación estandar. Además de esto la aplicación tendrá que ser desplegada con ayuda de Heroku.

## 2 Solución de Requerimientos

Para poder solucionar los requerimientos planteados se dividio el problema en 3 partes importantes, la implementación de la LinkedList que ya teníamos del primer laboratorio, el diseño y funcionamiento de la página web creada con la ayuda de Spark, y por último la conexión de estos para que funcione correctamente.

## 3 Diseño Web

Para el diseño de la página se utilizo un index.html, el cual fue siendo modificado para que éste tuviera un cuadro en donde nosotros podamos ingresar los números de la lista, un botón el cual active los métodos necesarios para calcular la media y la desviación estandar, y por último, un espacio en donde se mostrarian los respectivos resultados.

## 4 Aplicación Web

Para esta aplicación se utilizó JQuery framework de JavaScript, y este redirige el texto introducido hacia nuestra aplicación de Java.

## 5 Aplicación Java con Spark

Para esto se utilizó de base un código brindado por el profesor, y este fue modificado con un método post el cual nos conecta con una url específica, y ya con esto hecho con ayuda la función lambda, calcularemos lo pedido y lo retornaremos a unas variables, las cuales se obtendrán desde nuestra aplicación web con JavaScript.

## 6 Implementación LinkedList

Mi implementación consta de dos clases diferentes LinkedList y Node, en las cuales se crean las funciones y elementos necesarios para que esta funcione, como lo son la cabeza y la cola de cada nodo, y se implementaron tres métodos esenciales, los cuales fueron addNode, size y get.

### 6.1 Método addNode

Con este método se agregan nuevos valores a nuestra LinkedList. A este método le ingresa un parámetro que es de tipo double, y es el valor que se va a agregar. Si la cabeza del nodo está vacía, este valor se agrega ahí, de lo contrario se agregará al siguiente nodo.

### 6.2 Método size

Con este método se calcula el tamaño de nuestra LinkedList. Retorna un valor de tipo entero. Funciona con un contador y un ciclo while, el cual va iterando a través de nuestra lista.

### 6.3 Método get

Con este método obtenemos el valor que se encuentra en una posición de la LinkedList que nosotros necesitemos. A este método le ingresa una variable llamada index, la cual va a ser la posición requerida. Para esto utilizamos un ciclo while, el cual va a iterar hasta que el valor dado coincida con el valor de una variable asignada, y en cada iteración se cambiará al nodo siguiente hasta que sea la posición deseada.

## 7 Conclusión

Nos damos cuenta que para realizar éste ejercicio se deben tener diferentes conocimientos sobre diferentes tecnologías, como lo son HTML, JavaScript y Spark, las cuales son necesarias para la resolución del problema. Se cumple el objetivo de desplegar la aplicación con Heroku gracias a las instrucciones dadas en el documento.

## 8 Referencias

<https://www.javatpoint.com/java-program-to-create-and-display-a-singly-linked-list>

[https://www.overleaf.com/learn/latex/Creating\\_a\\_document\\_in\\_LaTeX](https://www.overleaf.com/learn/latex/Creating_a_document_in_LaTeX)

<http://sparkjava.com/documentation/getting-started>