

iWork es una empresa de tecnología especializada en IT Staff Augmentation. Contamos con una amplia base de desarrolladores (ubicados en un 95% en latinoamérica) que trabajan tercerizados para nuestros clientes.

Buscamos ampliar nuestra cartera de clientes a través de una campaña de email marketing. La aplicación desarrollada permite que dado un archivo de entrada “people.in” con información de perfiles públicos de LinkedIn, determine los 100 potenciales clientes con mayor probabilidad de comprar nuestros servicios. La salida esperada es un archivo “people.out” que contenga los ids de las personas que debemos contactar.

El archivo de entrada contiene, en cada línea, los siguientes campos separados por un pipe: PersonId, Name, LastName, CurrentRole, Country, Industry, NumberOfRecommendations, NumberOfConnections. Es posible que en algunos casos no conozcamos el valor de los campos, en tal caso aparecerán dos pipes consecutivos (| |).

Ejemplo:

people.in

```
4567|arturo|perez|teleport engineering manager|Germany|Telecommunications|2|176
4568|carlos|lobalzo|jefe de gestión funcional|Argentina|Financial Services|2|259
4569|juan|martinez|repositor de mercado|Colombia|Supermarkets|0|5
4580|john|smith|co-founder & cto|United States|Network Security|10|500
4592|alejandro|gonzalez|desarrollador de software|España|Telecommunications||300
...
```

people.out

```
4568
4567
4580
...
```

El Backend contiene un archivo de recursos “config.json” en el cual se pueden configurar las diferentes industrias, roles y cargos con un puntaje, y a través de este puntaje se calcula el rating de cada perfil en el archivo.

A través del frontend es posible cargar el archivo a procesar, y a; finalizar el procesamiento se desplegará una tabla ordenada de mayor puntaje a menor puntaje de 100 perfiles, igualmente al finalizar el procesamiento se descarga el archivo “people.out” automáticamente.

Criterios de Evaluación

1. La aplicación posee un Frontend construido en Angular 6 y un Backend construido en Java, se deben identificar los posibles errores de codificación que impiden que la aplicación funcione correctamente.
2. Se espera que se trate de optimizar el código de la aplicación según los principios de Clean Code.

3. Construcción de pruebas unitarias para el backend
4. Construir un nuevo componente donde se permite conocer el número de personas promedio por país y tipo de cargo que desempeña actualmente
5. (PLUS) Modificar la solución para que utilice Lambdas