Prueba técnica de aceptación para fabrica externa.

En la siguiente prueba técnica buscamos conocer su forma de trabajo y evaluar los siguientes aspectos: claridad del código, buenas prácticas, la simplicidad y la ejecución de las soluciones que se desarrollen.

Entregable

Los siguientes casos se deben desarrollar en **Node.js,** siéntete libre de ocupar las librerías que requieras para cumplir con el objetivo de los casos. Una vez finalizados:

* Compartir el código, por el medio que considere más adecuado.
* Desplegar en un servidor las soluciones.
* Documentar cada desarrollo.
* Proporcionar las ligas de los *endpoints* para poder realizar pruebas.

IMPORTANTE para ambos Casos

* Se deberá de crear un cliente con la arquitectura REST para la comunicación externa de la aplicación.
* Para la comunicación interna, se deberá configurar alguna herramienta para la transmisión de datos (deseable Kafka).

**Caso 1**

Se requiere realizar un intérprete de **comandos** que al introducir uno o varios de estos, nos retorne un resultado, Ej.:

INTERES MORATORIO 10000 + INTERES ORDINARIO 10000

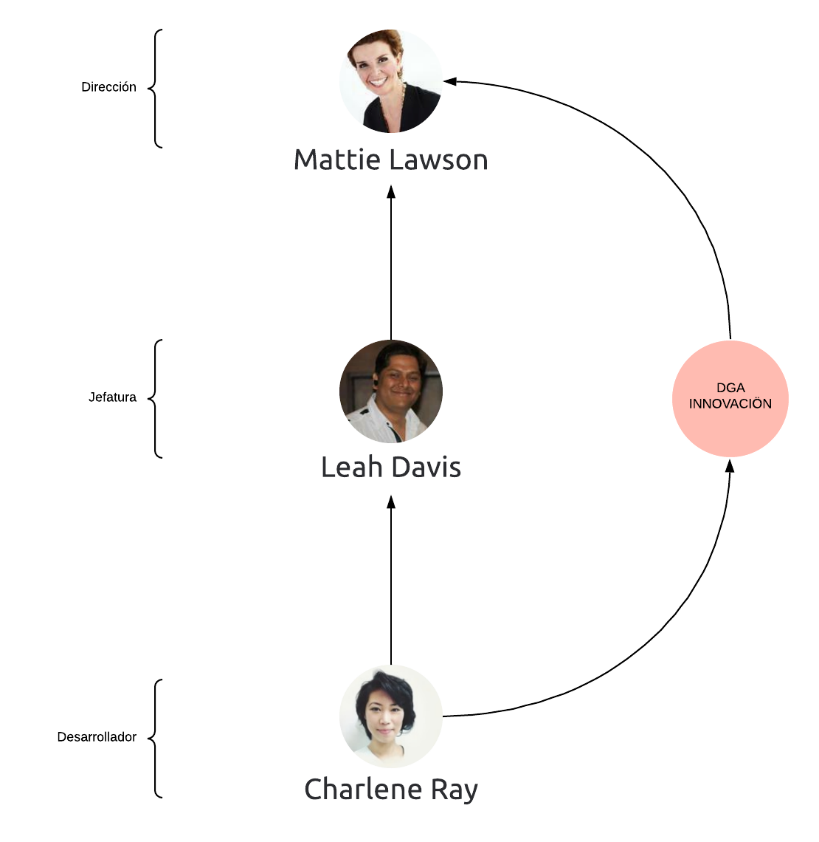
**Requerimientos**

* Calcular Pago Mínimo (<https://comparador.banxico.org.mx/ComparadorCrediticio/terminos-tarjeta-de-credito.jsp>).
* Calcular interés ordinario y moratorio.
* Calcular Fecha de pago.
* Calcular Fecha límite de pago.

**Caso 2**

En el área de Recursos humanos quieren implementar una página donde los colaboradores puedan ingresar al sistema y puedan visualizar el organigrama del área desde su posición hacia posiciones más arriba, es decir si un desarrollador pertenece al área de Innovación, podamos con solo buscar su número de empleado podamos ver la estructura del organigrama hasta la posición del director y el área a donde pertenecen.

En la siguiente imagen tenemos a *Charlene Ray,* cuando busquemos el número de empleado el programa deberá ser capaz de traer los paths hacia el nodo padre (Dirección) en formato JSON.



**Requerimientos**

* El programa deberá retornar la ruta o rutas posibles de un nodo dado al nodo raíz.
* Crear al menos 30 colaboradores asignados a diferentes niveles y relacionarlos para realizar diferentes consultas.
* La dirección de las rutas siempre debe ser hacia el nivel más alto (Nodo Raíz)
* Este servicio deberá de ser capaz de soportar 15,000 peticiones por segundo.