

# Detector Mutantes

Creado por

Diego Muñoz Rodríguez

Actualizado por última vez el 24/02/2021

## Problema

Magneto quiere reclutar la mayor cantidad de mutantes para poder luchar contra los X-Men.

Te ha contratado a ti para que desarrolles un proyecto que detecte si un humano es mutante basándose en su secuencia de ADN.

Para eso te ha pedido crear un programa con un método o función con la siguiente firma (En alguno de los siguiente lenguajes: Java / Golang / C-C++ / Javascript (node) / Python / Ruby):

**boolean isMutant(String[] dna); // Ejemplo Java**

En donde recibirás como parámetro un array de Strings que representan cada fila de una tabla de (NxN) con la secuencia del ADN. Las letras de los Strings solo pueden ser: (A,T,C,G), las cuales representan cada base nitrogenada del ADN.

## Niveles

**Nivel 1:** Este nivel abarca la implementación de la capa de negocio básica, sin ningún servicio REST expuesto

**Nivel 2:** Se expone un servicio que permite recibir una matriz y previo a una serie de validaciones determina si es o no mutante.

**Nivel 3:** Se agrega persistencia de datos usando mongodb y se agrega un nuevo servicio /stats que permite obtener las estadísticas de las secuencias registradas en el nivel 2, retornando la siguiente información

**count\_mutant\_dna:** Numero de mutantes detectados

**count\_human\_dna:** Numero de humanos detectados

**ratio:** Relación entre mutantes y humanos mutants / humans

## Solución Propuesta

Se implementará un microservicio en java, desplegado en Elastic Beanstalk de AWS, el cual expondrá los 2 servicios solicitados para responder a cada uno de los niveles que propone el desafío

## Aspectos técnicos

### Stack tecnológico

Tecnología	Motivo	Versión
------------	--------	---------

Java 8	Lenguaje de programación orientado a objetos. Se seleccionó por la robustez, alcance y capacidad de escalamiento que posee.	1.8
Git	Repositorio de versionamiento	--
AWS Elastic Beanstalk	Servicio para implementar y administrar aplicaciones en cloud	
EC2	Servidor virtual	
Swagger	Herramienta para documentación de apis por medio de anotaciones	--
Maven	Herramienta para resolver las dependencias.	3.0.0-M1
Spring Boot	Framework de desarrollo sobre Java. Fue escogido por la agilidad con la que permite desarrollar código de alto rendimiento, estandarizado y escalable. Ideal para desarrollo de microservicios	2.3.3.RELEASE
Jacoco	Jacoco maven plugin es un herramienta que analiza la cobertura de pruebas y genera un reporte de la misma	0.8.3
Lombok	Es una librería para Java que a través de anotaciones nos reduce el código que codificamos	1.18.12
Mongo DB	Es una base de datos no sql ideal para manejo de estructuras json, se usa por su simpleza para configuración de cluster y por su optimización para escrituras rápidas	3.12.7

### Patrones empleados

A continuación, nombramos algunos de los patrones de diseño empleados en la solución

- Observer
- Facade
- Fabrica abstracta





















Consideraciones y supuestos:

- Cada matriz representa una persona(mutante o humano) por lo que se asume que ninguna matriz será igual a una ya ingresada.
- El tamaño mínimo de la secuencia es de 4 caracteres por lo tanto una matriz no podrá tener un tamaño inferior a 4x4.

## Reporte Jacoco

Se configuro para no permitir que se genere una versión si una cobertura de los test unitarios inferior al 80%.

### rest-mutant-meli

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed Cxty	Missed Lines	Missed Methods	Missed Classes
com.meli.mutant.domain		21 %		2 %	35 46	5 17	11 22	0 2
com.meli.mutant.util		29 %		n/a	5 9	15 23	5 9	0 1
com.meli.mutant.model		63 %		50 %	4 12	2 10	2 10	1 3
com.meli.mutant.service.detector		97 %		93 %	5 48	2 86	2 24	0 7
com.meli.mutant.repository.impl		89 %		50 %	2 6	1 19	0 4	0 1
com.meli.mutant.service.validator		87 %		85 %	2 12	2 17	0 5	0 1
com.meli.mutant.exception		92 %		n/a	2 13	4 23	2 13	1 6
com.meli.mutant.controller		82 %		100 %	0 5	2 8	0 4	0 2
com.meli.mutant.service.impl		100 %		100 %	0 8	0 18	0 7	0 2
com.meli.mutant		100 %		n/a	0 4	0 9	0 4	0 1
Total	375 of 1.371	72 %	56 of 122	54 %	55 163	33 230	22 102	2 26

## Recursos

- Todas las indicaciones y recursos del programa se encuentran en el archivo README.md que se encuentra en la raíz del repositorio