

ESPECIFICACIONES Y CONSIDERACIONES EN LA SOLUCION

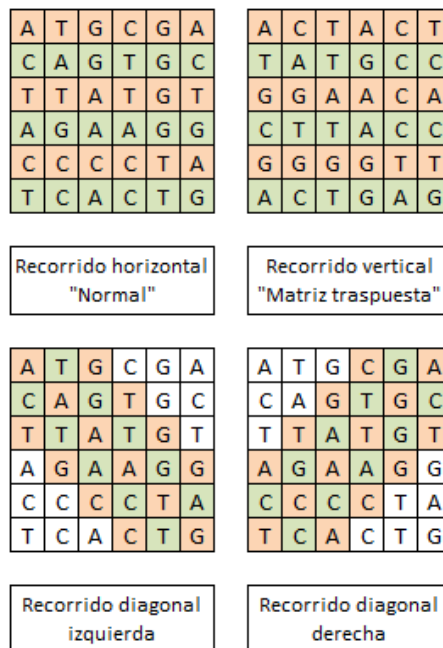
Planteamiento de la solución

Observando el problema base a abordar se identifica que:

- Manejo de información alfabética con restricción a solo 4 caracteres específicos.
- Es una matriz simétrica sin dimensión especificada.
- Se debe evaluar verticalmente, horizontalmente y diagonalmente donde este ultimo son dos recorridos (De izquierda a derecha y de derecha a izquierda).

Bases para la solución:

- Se recibe un arreglo de strings, donde o puedo abordar cada registro caracter a caracter mediante ciclos o ver toda la línea como una palabra y ubicar un patrón y tratar de extraer lo que queremos mediante expresiones regulares.
- Al ser una matriz de solo letras y ciertas letras en sí, se aplican una serie de validaciones previas a evaluar la razón de la API en sí, para este caso como solo se especifica que devuelva un http 200 para un mutante y http 403 para un humano, las validaciones que indiquen que la matriz no es correcta para evaluar devolverán un false (http 403).
- Si se aborda con análisis de expresiones regulares puedo tomar la primer matriz como viene (horizontal) y las demás las puedo transformar donde, la vertical es la traspuesta de la original y la diagonal solo se debe recorrer desde la cadena más larga hasta 4 que representa la secuencia de mismos caracteres a analizar.



- Se desarrolla un servicio web de tipo REST con capas, donde se uso FACADE para exponer los servicios mediante interfaces.

En las contemplaciones técnicas:

- Se da el nombre de la firma y el tipo de dato de retorno (Fijo), entonces ya tenemos la base del primer endpoint.
- Se observa la sugerencia de alta carga de trabajo. Para esto la solución requerirá en algún momento escalado, por lo cual se puede desplegar el artefacto o solución en varios contenedores docker controlados por un balanceador de carga.

Tecnologías usadas

Desarrollo:

- Java 8
- Maven
- JUnit
- Spring Boot
- MySql

Cobertura en pruebas unitarias:

- Java Code Coverage (JaCoCo): Se da más detalle de esta herramienta y las pruebas en el documento " *Test coverage/Ejecucion de cobertura.pdf*"

Pruebas de carga:

- Jmeter: Se da más detalle de esta herramienta y como se utilizo en el documento " *Load testing/Ejecucion de pruebas con jmeter.pdf*"

Cloud:

- Google cloud platform

Escalar la aplicación:

Para contemplar el punto referente a la alta concurrencia se sugiere desplegar la solución en uno o varios contenedores **docker** según la necesidad y que entre quien/es realizan la petición agregar un balanceador de carga. En este apartado se explicara como desplegar la solución en un contenedor.

Prerrequisitos:

- Tener Jdk 8 instalado
- Tener instalado maven
- Tener Docker instalado

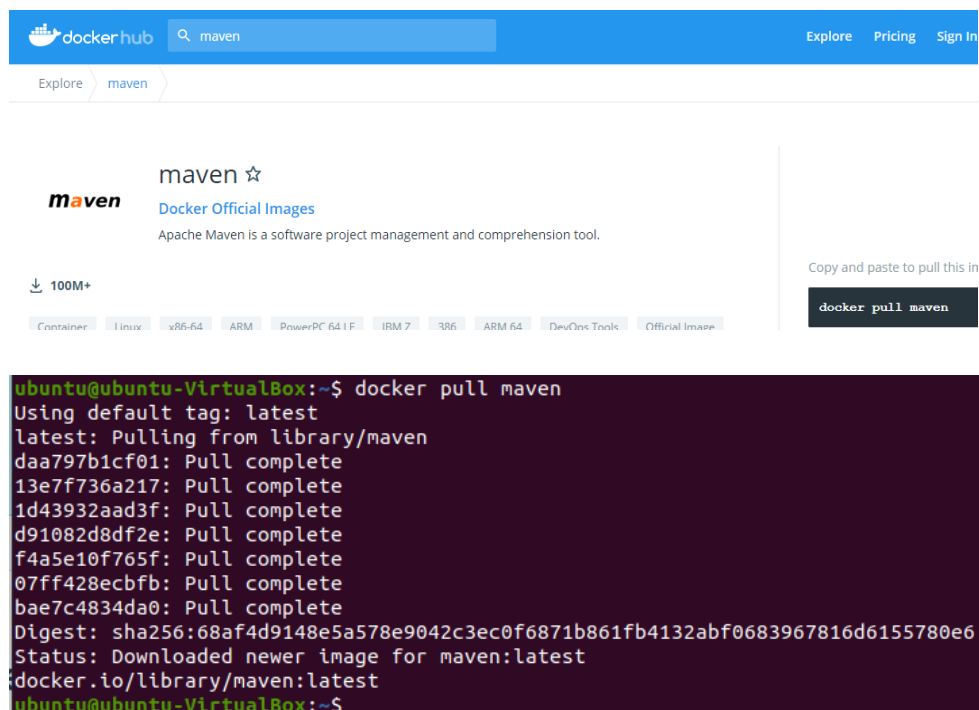
Para agregar la solución se debe:

1. Este proyecto es hecho con maven, entonces luego de configurado en un IDE o solo descargarlo, se puede hacer uso del comando "**mvn clean install**" cuando se tiene la variable de entorno del maven configurada o el comando "**mvnw.cmd clean install**" sino lo reconoce y lo anterior es para que genere el empaquetado **.jar**.

```
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 3.681 s - in com.magonzalez.xmen.XmenApplicationTests
[INFO] Results:
[INFO] Tests run: 29, Failures: 0, Errors: 0, Skipped: 0
[INFO] --- maven-jar-plugin:3.2.0:jar (default-jar) @ xmen-ws ---
[INFO] Building jar: C:\Users\mokanime\Desktop\Prueba Mercadolibre\xmen-ws\target\xmen-ws-0.0.1-SNAPSHOT.jar
[INFO] --- spring-boot-maven-plugin:2.5.0:repackage (repackage) @ xmen-ws ---
[INFO] Replacing main artifact with repackaged archive
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ xmen-ws ---
[INFO] Installing C:\Users\mokanime\Desktop\Prueba Mercadolibre\xmen-ws\target\xmen-ws-0.0.1-SNAPSHOT.jar to C:\Users\mokanime\.m2\repository\
[INFO] Installing C:\Users\mokanime\Desktop\Prueba Mercadolibre\xmen-ws\pom.xml to C:\Users\mokanime\.m2\repository\com\magonzalez\xmen-ws\0.
[INFO] BUILD SUCCESS
[INFO] Total time: 01:07 min
[INFO] Finished at: 2021-05-31T19:32:49-05:00
```

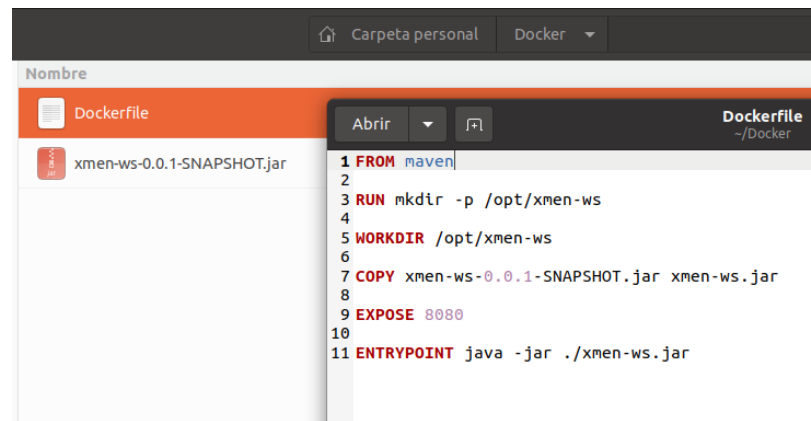
2. Si no se cuenta con una imagen docker con java 8, se puede realizar la respectiva búsqueda en el portal de Docker Hub y de allí ubicar la que mejor se ajuste. Para este caso descargo una imagen con maven (ejemplo):

https://hub.docker.com/_/maven



```
ubuntu@ubuntu-VirtualBox:~$ docker pull maven
Using default tag: latest
latest: Pulling from library/maven
daa797b1cf01: Pull complete
13e7f736a217: Pull complete
1d43932aad3f: Pull complete
d91082d8df2e: Pull complete
f4a5e10f765f: Pull complete
07ff428ecbf8: Pull complete
bae7c4834da0: Pull complete
Digest: sha256:68af4d9148e5a578e9042c3ec0f6871b861fb4132abf0683967816d6155780e6
Status: Downloaded newer image for maven:latest
docker.io/library/maven:latest
ubuntu@ubuntu-VirtualBox:~$
```

3. El jar generado lo almacenan en una ruta fácil de acceder para poderlo referenciarlo en el archivo "Dockerfile" sin extensión.
4. Crear el archivo Dockerfile con las siguientes instrucciones:



5. Se crea una nueva imagen con este archivo Dockerfile:

```
ubuntu@ubuntu-VirtualBox:~/Docker$ ls
Dockerfile  xmen-ws-0.0.1-SNAPSHOT.jar
ubuntu@ubuntu-VirtualBox:~/Docker$ docker build . -t xmen-ws-image
Sending build context to Docker daemon 38.97MB
Step 1/6 : FROM maven
--> 968d394ed243
Step 2/6 : RUN mkdir -p /opt/xmen-ws
--> Running in d97abe7502ea
Removing intermediate container d97abe7502ea
--> c009733dcf1e
Step 3/6 : WORKDIR /opt/xmen-ws
--> Running in e1a224ded170
Removing intermediate container e1a224ded170
--> 73ca61f6d27c
Step 4/6 : COPY xmen-ws-0.0.1-SNAPSHOT.jar xmen-ws.jar
--> d072a085e582
Step 5/6 : EXPOSE 8080
--> Running in 3079c36cd9d9
Removing intermediate container 3079c36cd9d9
--> 1afe9bf37572
Step 6/6 : ENTRYPOINT java -jar ./xmen-ws.jar
--> Running in 6b430c12d32e
Removing intermediate container 6b430c12d32e
--> ab49c9bc7a3d
Successfully built ab49c9bc7a3d
Successfully tagged xmen-ws-image:latest
ubuntu@ubuntu-VirtualBox:~/Docker$
```

6. Se listan las imágenes para validar que este:

```
ubuntu@ubuntu-VirtualBox:~/Docker$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
xmen-ws-image	latest	ab49c9bc7a3d	2 minutes ago	806MB
maven	latest	968d394ed243	3 days ago	767MB

7. Se crea un primer contenedor a partir de esa imagen con el comando "Ejemplo":

```
docker run -p 8585:8080 --name xmen-container xmen-ws-image
```

Donde:

- -p es el mapeo de puertos (el de la izquierda es el del host y el de la derecha el del container)

- ```


[+] Terminal ▾ 31 de mayo 2023
ubuntu@ubuntu-VirtualBox: ~/Docker

ubuntu@ubuntu-VrtualBox:~/Docker$ docker run -p 8585:8080 --name xmen-container xmen-ws-image

=====
:: Spring Boot ::
 (v2.5.0)

2021-06-01 01:21:35.848 INFO 1 --- main | com.magonazalezo.xmen.XmenApplication : Starting XmenApplication v0.0.1-SNAPSHOT using Java 16.0.1 on 217046524fe0 with PID
2021-06-01 01:21:35.878 INFO 1 --- main | org.springframework.boot.StartupInfoLogger : No active profile set, falling back to default profiles: default
2021-06-01 01:21:42.169 INFO 1 --- main | .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2021-06-01 01:21:42.598 INFO 1 --- main | .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 369 ms. Found 1 JPA repository interface
2021-06-01 01:21:48.087 INFO 1 --- main | o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started with port(s): 8080 (http)
2021-06-01 01:21:48.181 INFO 1 --- main | o.apache.catalina.core.StandardService : Initializing service [Tomcat]
2021-06-01 01:21:48.182 INFO 1 --- main | org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.46]
2021-06-01 01:21:48.685 INFO 1 --- main | o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2021-06-01 01:21:48.685 INFO 1 --- main | w.s.c.ServletWebApplicationInitializer : Root WebApplicationContext: initialization completed in 12328 ms
2021-06-01 01:21:52.465 INFO 1 --- main | o.h.bernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2021-06-01 01:21:53.112 INFO 1 --- main | o.h.bernate.annotations.common.Version : HHH0000412: Hibernate ORM version 5.4.3.Final
2021-06-01 01:21:55.063 INFO 1 --- main | com.zaxxer.hikari.HikariDataSource : HCANN000001: Hibernate Commons Annotations [5.1.2.Final]
2021-06-01 01:21:59.238 INFO 1 --- main | com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2021-06-01 01:21:59.367 INFO 1 --- main | com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2021-06-01 01:21:59.367 INFO 1 --- main | org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL57Dialect
2021-06-01 01:21:59.362 INFO 1 --- main | o.h.e.t.j.p.l.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta
Platform]
2021-06-01 01:22:03.550 INFO 1 --- main | j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-06-01 01:22:05.696 WARN 1 --- main | JpaBaseConfigurationJpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be p
ndering. Explicitly configure spring.jpa.open-in-view to disable this warning
2021-06-01 01:22:08.531 INFO 1 --- main | o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2021-06-01 01:22:08.622 INFO 1 --- main | com.magonazalezo.xmen.XmenApplication : Started XmenApplication in 36.321 seconds (JVM running for 39.981)
2021-06-01 01:22:08.642 INFO 1 --- main | o.s.b.a.ApplicationAvailabilityBean : Application availability state LiveState changed to CORRECT
2021-06-01 01:22:08.651 INFO 1 --- main | o.s.b.a.ApplicationAvailabilityBean : Application availability state ReadinessState changed to ACCEPTING_TRAFFIC

```



The screenshot shows a web browser window with the address bar displaying 'localhost:8585/stats'. The browser's address bar and the REST client's response are visible. The REST client shows a JSON response with the following data:

```

{
 "count_mutant_dna": 1,
 "count_human_dna": 2,
 "ratio": 0.5
}

```

```
Terminal 31 de may 20:29
ubuntu@ubuntu-VirtualBox: ~/Docker

ubuntu@ubuntu-VirtualBox:~/Docker$ docker run -p 8585:8080 --name xmen-container xmen-ws-image

ubuntu@ubuntu-VirtualBox: ~$ docker run -p 8787:8080 --name xmen-container2 xmen-ws-image

=====
:: Spring Boot ::
=====

2021-06-01 01:21:11.111:INFO:main:com.magonzalez.xmen.XmenApplication: Starting XmenApplication
2021-06-01 01:21:11.111:INFO:main:com.magonzalez.xmen.XmenApplication: No active profile found, using default: dev
2021-06-01 01:21:11.111:INFO:main:s.d.r.c.RepositryConfigurationDelegate: Bootstrapping
2021-06-01 01:21:11.111:INFO:main:s.d.r.c.RepositryConfigurationDelegate: Finished Spring
2021-06-01 01:21:11.111:INFO:main:org.springframework.boot.web.embedded.tomcat.TomcatWebServer: Tomcat initial
2021-06-01 01:21:11.111:INFO:main:org.apache.catalina.core.StandardService: Starting service
2021-06-01 01:21:11.111:INFO:main:org.apache.catalina.core.StandardEngine: Starting Servlet
2021-06-01 01:21:11.111:INFO:main:org.apache.catalina.core.StandardContext: Initializing S
2021-06-01 01:21:11.111:INFO:main:w.s.c.ServletWebServerApplicationContext: Root WebApplic
2021-06-01 01:21:11.111:INFO:main:org.hibernate.jpa.internal.util.LogHelper: HHH000204: Pro
2021-06-01 01:21:11.111:INFO:main:org.hibernate.Version: HHH000412: Hib
2021-06-01 01:21:11.111:INFO:main:org.hibernate.annotations.common.Version: HCANN000001: H
2021-06-01 01:21:11.111:INFO:main:jpaBaseConfiguration$JpaWebConfiguration: spring.jpa.open-in-view is enable
2021-06-01 01:21:11.111:INFO:main:jpaBaseConfiguration$JpaWebConfiguration: to disable this warning
2021-06-01 01:22:08.622:INFO:main:org.springframework.boot.web.embedded.tomcat.TomcatWebServer: Tomcat started on port(s): 8080
2021-06-01 01:22:08.622:INFO:main:com.magonzalez.xmen.XmenApplication: Started XmenApplication in 36.32
2021-06-01 01:22:08.622:INFO:main:org.springframework.boot.autoconfigure.condition.ConditionalOnAvailableBean: Application availability sta
```

des Navegador web Firefox

localhost:8585/stats localhost:8787/stats

localhost:8787/stats

JSON Datos sin procesar Cabeceras

Guardar Copiar Contraer todo Expandir todo Filtrar JSON

|                   |     |
|-------------------|-----|
| count_mutant_dna: | 1   |
| count_human_dna:  | 2   |
| ratio:            | 0.5 |