|  |
| --- |
| portátil en mesa de oficina  Documentación Solución de prueba técnica Devops Engine  Presentado por: Eyvind Mendoza Banco CuscatlánFecha: 27/12/2021 |
| Página 1 |

A continuación se describe como se realizo la prueba tecnica y los diagramas de dicha solución Como primer punto se debia realizar:

**Levantar 2 clúster en kubernetes usando IaC, uno de deployment y otro de development, que permita desplegar un hello-world en java 16 almacenado en el sistema de versionamiento en la nube de su preferencia.**

1)Se creo Proyecto java para el microservicio hello world el cual se subio al repositorio git publico :

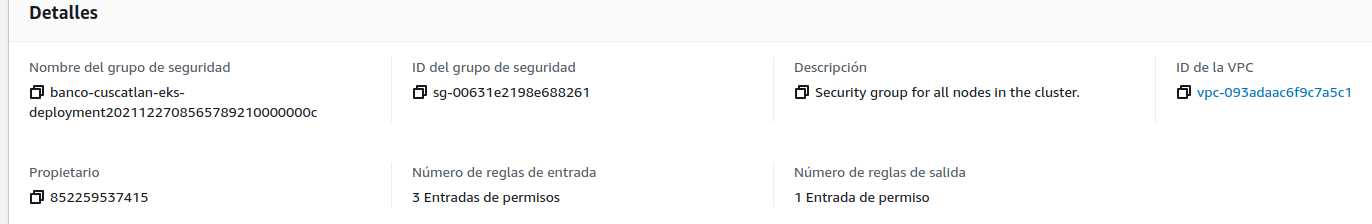
<https://github.com/eyvindx/java16-helloworld>

en este repositoriol se crearon ramas master, development, hotfix, realease y feature basado en la metodologia gitflow

la rama feature/bccicd02 es rama que usa el pipeline de Jenkins.

**Proveedor de nube AWS.**

2) Se creo una EC2 para instalar Docker, el servidor de Jenkins y sonarqube, aplicando medidas de seguridad creando security groups para poder acceder a estos recursos



La url de Jenkins es:

<http://ec2-100-26-99-124.compute-1.amazonaws.com:8080/>

credenciales

user: emendoza

password: gaton.,2

aqui se puede visualizer y ejecutar el pipeline llamado pipeline1

el cual realiza el CI y CD dejando una imagen de docker del microservicio helloword

**la ruta del docker hub donde se despliega la imagen del microservicio en java 16 que genera el pipeline Jenkins es :**

https://hub.docker.com/repository/docker/gaton/helloworld

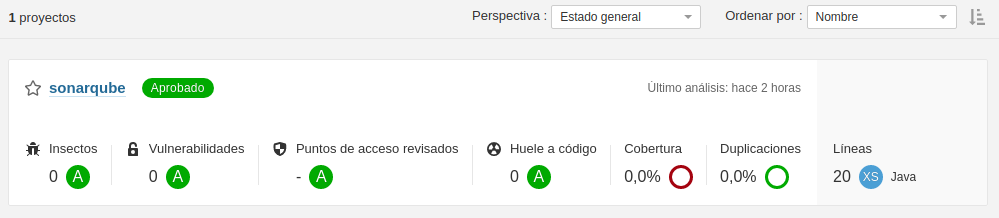
**La ruta de sonarqube es:**

<http://ec2-100-26-99-124.compute-1.amazonaws.com:9000/>

user: admin

password: gaton.,2

aqui se pueden visualizer los resultados del analisis



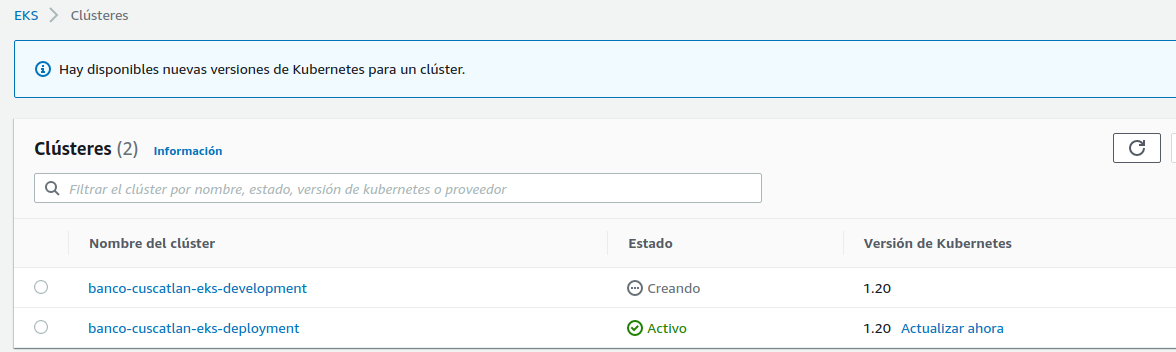
3) Para Iac se procedio a instalar el terraform cli para crear cada cluster (development y deployment)

Los archivos estan en

<https://github.com/eyvindx/documentacion-prueba-cuscatlan>

en los cuales se aplicaron buenas practicas, separando los archivos modularmente

estos procesos generaron los 2 cluster en aws



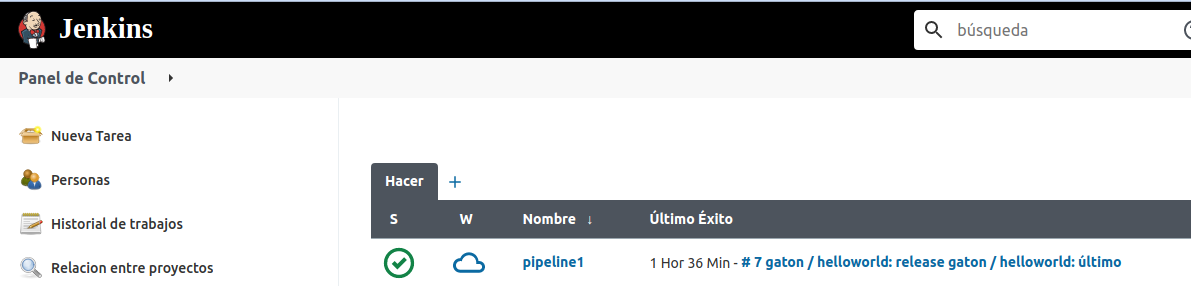
Cluster deployment con una instancia del microservicio

development con 2 instancias del microservicio y un balanceador de carga según lo solicitado en la guia. 

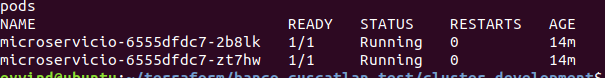
**Sobre el Pipeline jenkins**

El pipeline usa los plugin de sonarqube, docker, git, kubernetes

Al ejecutarse verifica el quality gates si este es ok continua con el build de maven y finalmente crea la imagen de docker que es tomada por el iac para desplegar y crear los pods en los cluster de kubernetes.



Microservicio con dos replicas



Un balanceador de carga



DIAGRAMA

