



ARQUITECTURA EMPRESARIAL
LUIS DANIEL BENAVIDES NAVARRO

INFORME LABORATORIO 5 - TALLER MODULAR AWS Y DOCKER

Autores:
Guillermo Castro

Marzo 2021

Tabla de Contenido

1. Introducción	2
2. Objetivos	2
3. Estructura,Diseño y Aquitectura del Programa	3
3.1. Estructura	3
3.2. Diseño	3
3.3. Arquitectura	4
4. Pruebas	5
5. Conclusiones	7
6. Referencias	8

1. Introducción

La transformación digital en el mundo ha impulsado el crecimiento del uso de servicios cloud en las organizaciones, esto convierte las necesidades de computación sobre la nube en una oportunidad vital para publicar los servicios que se necesiten de manera rápida, práctica y segura. El objetivo de este informe consiste en crear una arquitectura que contiene un load balancer, 3 nodos que manejan requests REST y un nodo que contiene una base de datos mongo. Cada nodo representa un contenedor en Docker. La idea es trabajar con el microframework java Spark. Una vez que tengamos nuestra infraestructura procederemos a construir una aplicación web que reciba mensajes y los maneje adecuadamente para mantener la disponibilidad. Luego, crearemos un repositorio en DockerHub y subiremos nuestras imágenes. Finalmente, crearemos una máquina virtual AWS EC2, instalaremos Docker/Docker-compose y desplegaremos nuestra arquitectura en el clon. .

2. Objetivos

- Retomar conocimientos aprendidos de asignaturas pasadas, poniendo en práctica todos los sistemas complejos.
- Aprender y entender el funcionamiento o la construcción de los contenedores.
- Entender e implementar las peticiones web y el ciclo de una aplicación web desplegada
- Construir una arquitectura web Cloud con AWS y Docker
- Aprender a utilizar el hub de Docker.
- Identificar las ventajas del uso de contenedores.

3. Estructura,Diseño y Aquitectura del Programa

3.1. Estructura

Esta estructura la usamos para ver como se esta conformando el proyecto y lo podemos unir al readme creado donde vemos mas a fondo la descripcion del proyecto.



Figura 1: Estructura de arbol del proyecto creado

3.2. Diseño

Aquí presento el diagrama de clases de la arquitectura,se presentan dos clases que implementamos para dar solución al Laboratorio AREP 5.

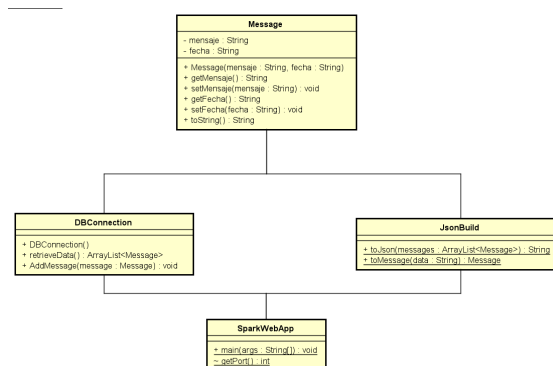


Figura 2: Diagrama de Clases creado en astah

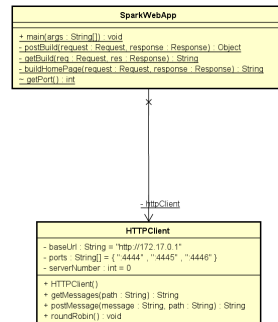


Figura 3: Diagrama de Clases creado en astah

3.3. Arquitectura

En el aplicativo tiene una arquitectura muy sencilla en cuanto a la parte web. Se realizó la implementación de el balancceador de carga y el servidor que atiende las peticiones en 2 proyectos en java y maven separados, a cada uno de estos proyectos se les agrego su respectivo Dockerfile, se crearon las imagenes y se subieron a dockerhub (Link servidor Web / Link Balanceador de carga). Una vez subidos a DockerHub, en la máquina EC2 de AWS se realizó la respetiva instalación de docker y se ejecutó el docker-compose (docker-compose up -d scale web=3) para crear los contenedores de las imagenes mencionadas anteriormente y crear o ejecutar la imagen y el contenedor del Motor Mongo DB.

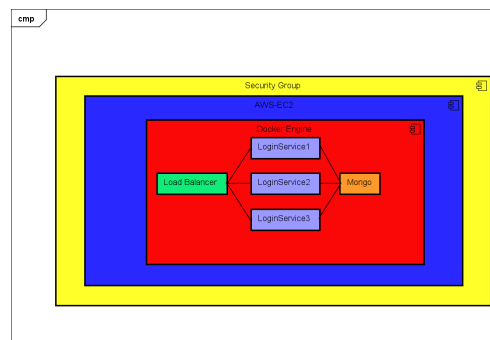


Figura 4: Diagrama de Componentes creado en astah

4. Pruebas

Se realizo la prueba de la instancia lanzada con la estructura verificada que se llevo acabo

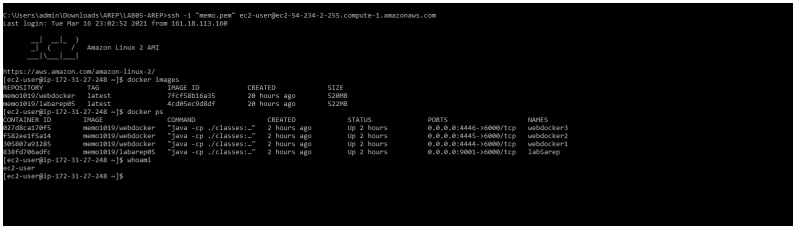


Figura 5: Pruebas

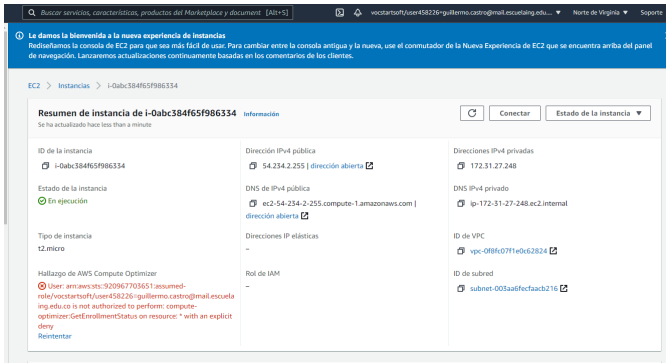


Figura 6: Pruebas

Se realizaron las pruebas con el despliegue en Docker llevado acabo y la aplicacion funcionando de manera optima

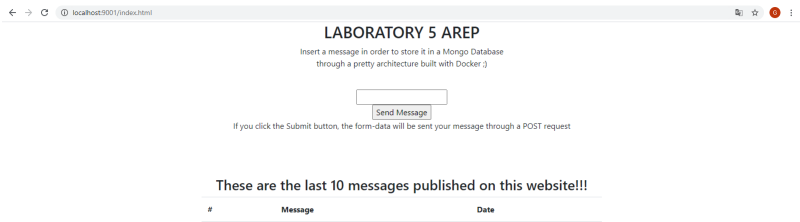


Figura 7: Prueba de despliegue en docker

y finalmente se hizo prueba del servidor web creado desplegado en Ec2 con aws en el docker

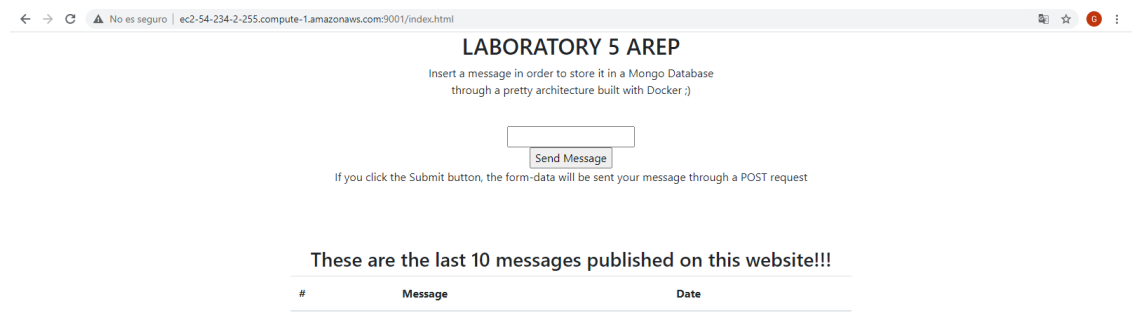


Figura 8: Prueba de aws

5. Conclusiones

- Esta actividad permitió conocer una arquitectura implementada con docker y desplegada en una maquina virtual de AWS, se entendió el funcionamiento, comportamiento y metodología de las imágenes y contenedores con Docker y tambien se fortalecieron los conocimientos de despliegue y configuración de máquinas virtuales en la nube.
- Se realizo la implementacion de los metodos solicitados para realizar el servidor web de manera optima,logrando ver que es posible la construccion de un servidor web mediante java.
- Es necesario tener conocimientos sobre los encabezados http y como se envian ya que estos son la base de como podemos tratar la información recilida y de como debemos enviarla para que el navrgador poeda entender.
- Mediante las imagenes creadas en DockerHub, se puede desplegar el aplicativo web en una máquina de AWS en menos de 5 minutos, haciendo que esta tecnología cobre un gran valor y nos de alta disponibilidad en caso de fallo.
- He comprendido la importancia de saber utilizar docker y los beneficios de esta nueva tecnología, realmente corresponde a una solución tecnológica impresionante que sirve para el desarrollo de aplicaciones, ya sea en la nube u on-premise
- La implementación del servidor web cumple con su proposito de recibir peticiones get y post, asi como de renderizar archivos estáticos como JavaScript, Css, Html e imagenes.

6. Referencias

- Socket, <https://www.speedcheck.org/es/wiki/socket/fn1>, Accessed on 2020-09-03.
- What is a web server? (2020-09-02.) Servidor Web, <https://blog.infranetworking.com/servidor-web/>, Acces- sed on 2020-09-03.
- HTTP Request / Response,<https://sites.google.com/site/conceptoprogramacic/request-response>, Accessed on 2020-09-03.
- ¿Qué son Spring framework y Spring Boot? Tu primer programa Java con este framework?,(2020-03-31) de <https://www.campusmvp.es/recursos/post/que-son-spring-framework-y-spring-boot-tu-primer-programa-java-con-este-framework.aspx>
- POST, [https://developer.mozilla.org/es/docs/Web/HTTP/Methods/ POST](https://developer.mozilla.org/es/docs/Web/HTTP/Methods/POST), Accessed on 2020-09-03.
- ¿Qué es DOCKER?, [https://www.redhat.com/es/topics/containers/ what-is-docker](https://www.redhat.com/es/topics/containers/what-is-docker). Accessed on 2020-09-21.