

 Hecho

Apertura: jueves, 7 de marzo de 2024, 13:30

Cierre: jueves, 7 de marzo de 2024, 16:59

Primera parte: AWS

1. Cree una aplicación web, cree la imagen docker , suba la imagen a dockerhub
2. Acceda a la máquina virtual
3. Instale Docker

```
sudo yum update -y  
sudo yum install docker
```

4. Inicie el servicio de docker

```
sudo service docker start
```

5. Configure su usuario en el grupo de docker para no tener que ingresar “sudo” cada vez que invoca un comando

```
sudo usermod -a -G docker ec2-user
```

6. Desconectes de la máquina virtual e ingrese nuevamente para que la configuración de grupos de usuarios tenga efecto.
7. A partir de la imagen creada en Dockerhub cree una instancia de un contenedor docker independiente de la consola (opción “-d”) y con el puerto 6000 enlazado a un puerto físico de su máquina (opción -p):

```
docker run -d -p 42000:6000 --name firstdockerimageaws danielben/firstsprkwebapprepo
```



8. Abra los puertos de entrada del security group de la máxima virtual para acceder al servicio
9. Verifique que pueda acceder en una url similar a esta (la url específica depende de los valores de su maquina virtual EC2)

```
http://ec2-35-175-205-168.compute-1.amazonaws.com:42002/hello
```

Debería ver algo así:

Debería ver algo así:



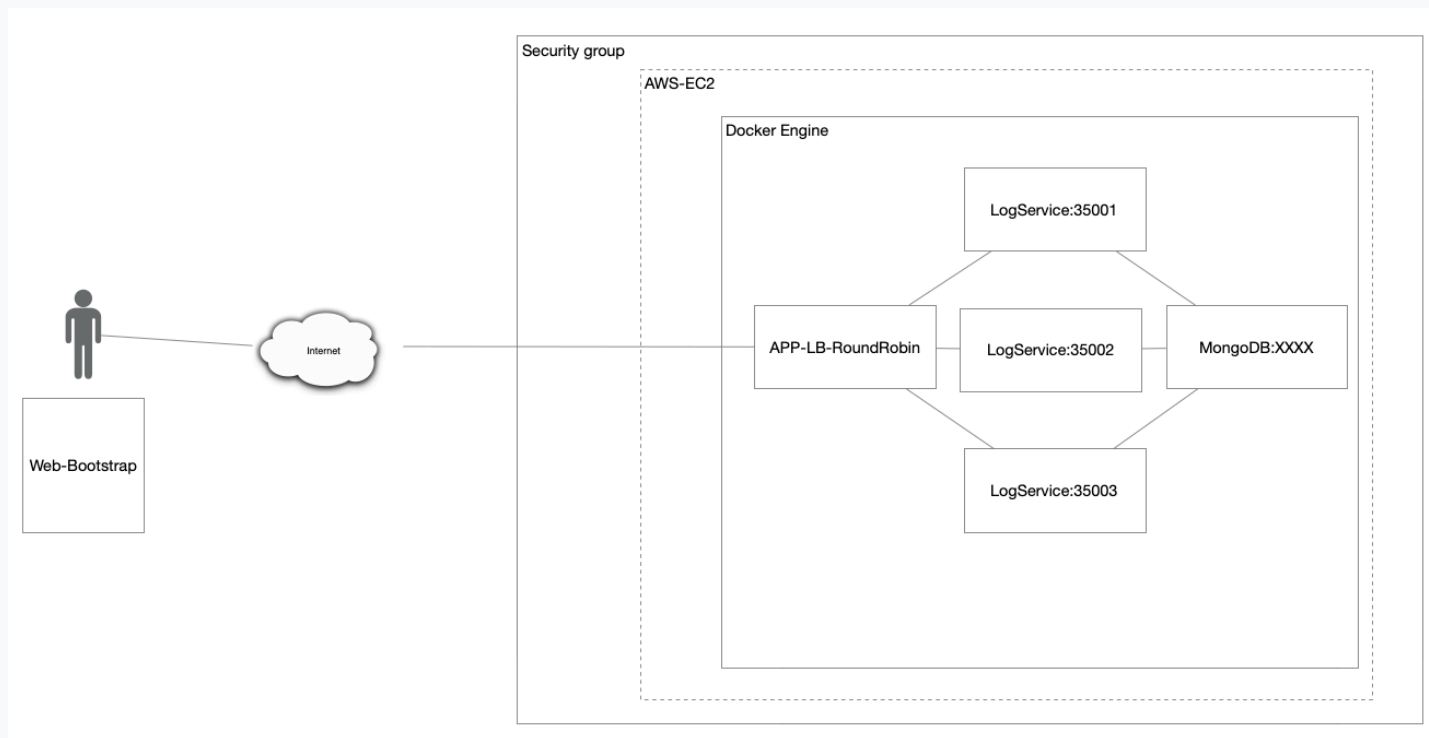
  ec2-35-175-205-168.compute-1.amazonaws.com:42002/hello

 Getting Started  Inbox (1) - danielben...  QuantumLibraries/C...  Escuela Colombiana...  Quantum theory, th...

Hello Docker!

TAREA

Para la tarea usted debe construir una aplicación con la arquitectura propuesta y desplegarla en AWS usando EC2 y Docker.



1. El servicio MongoDB es una instancia de MongoDB corriendo en un container de docker en una máquina virtual de EC2
2. LogService es un servicio REST que recibe una cadena, la almacena en la base de datos y responde en un objeto JSON con las 10 ultimas cadenas almacenadas en la base de datos y la fecha en que fueron almacenadas.
3. La aplicación web APP-LB-RoundRobin está compuesta por un cliente web y al menos un servicio REST. El cliente web tiene un campo y un botón y cada vez que el usuario envía un mensaje, este se lo envía al servicio REST y actualiza la pantalla con la información que este le regresa en formato JSON. El servicio REST recibe la cadena e implementa un algoritmo de balanceo de cargas de Round Robin, delegando el procesamiento del mensaje y el retorno de la respuesta a cada una de las tres instancias del servicio LogService.

Entregables:

1. El código del proyecto en un repositorio de GITHUB
2. Un README que explique un resumen del proyecto, la arquitectura, el diseño de clases y que muestre cómo generar las imágenes para desplegarlo. Además que muestre imágenes de cómo quedó desplegado cuando hicieron las pruebas.
3. Video con los despliegues funcionando.

Editar entrega

Borrar entrega

ESTADO DE LA ENTREGA

Estado de la entrega	Enviado para calificar
Estado de la calificación	Sin calificar
Tiempo restante	La tarea fue enviada 1 hora 13 minutos antes
Última modificación	jueves, 7 de marzo de 2024, 15:45
Texto en línea	+ https://github.com/ELS4NTA/AREP-LAB-o6.git
Comentarios de la entrega	► Comentarios (0)

ENLACES INSTITUCIONALES

Biblioteca

Investigación e innovación

Enlace - Académico

ENLACES DE INTERÉS

Ministerio de Educación Nacional

Colombia Aprende

Red Latinoamericana de Portales Educativos

Red Universitarias Metropolitana de Bogotá

CONTACTE CON NOSOTROS

📍 AK.45 No.205-59 (Autopista Norte).

 Teléfono: +57(1) 668 3600

Email: contactocc@escuelaing.edu.co

Escuela Colombiana de Ingeniería Julio Garavito