



UNIVERSIDAD DE GUADALAJARA
CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍAS
DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

Diferencias entre Docker, Kubernetes, Apache Mesos y OpenShift

Oscar Evanilson Gutiérrez Pérez

Carrera: Ingeniería en Computación

Código de estudiante: 219748308

2 de mayo de 2022

Diferencias:

Docker es la base de todas estas aplicaciones, ya que se necesitan los contenedores de Docker para utilizar las imágenes que se van a desplegar en las demás opciones, con Docker se empaqueta nuestra aplicación.

Kubernetes no puede crear imágenes, sino que es un orquestador de contenedores, por lo que sirve de buena manera en aplicaciones ya realizadas y empaquetadas por Docker.

Apache Mesos es un administrador de Cluster de código abierto basado en el kernel de Linux, a diferencia de que Mesos puede correr en cualquier plataforma ya sea Linux, Windows o OSX. Además, proporciona aplicaciones con API's para el manejo de recursos y planificación.

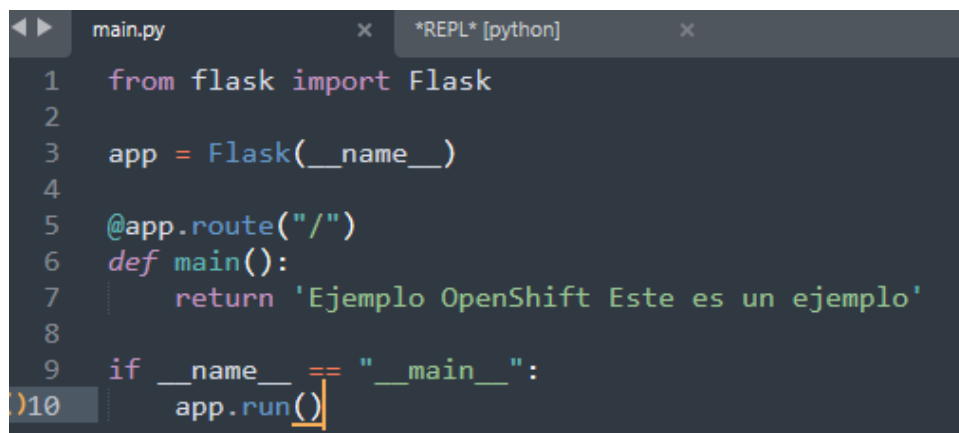
El mismo abstrae CPU, memoria, almacenamiento y otros recursos desde servidores dedicados o máquinas virtuales permitiendo sistemas elásticos de fácil construcción y que se puede correr de manera eficaz, destacando la tolerancia a fallos.

Mesos se utiliza para correr grandes aplicaciones especialmente aquellas que manejan grandes volúmenes de datos y se hace inviable correr en un único servidor. Mesos permite utilizar el cluster para correr los frameworks sobre los cuales se ejecuta la aplicación, distribuyendo la carga entre los distintos servidores y evitando así sobrecargas logrando un óptimo desempeño. Mesos se utiliza comúnmente para aplicaciones Java, Python, Scala y R.

Por último, OpenShift es una plataforma de desarrollo, con características de Cloud Computing (PaaS) que ofrece la empresa de Red Hat.

Esta plataforma se desarrolla a partir de contenedores kubernetes que los desarrolladores utilizan para desplegar apps en diferentes lenguajes de programación. Su principal objetivo es mejorar la productividad de los desarrolladores y promover la innovación.

Para el ejemplo de esta semana, utilice un tutorial muy sencillo de una aplicación y los pasos que se deben de realizar para utilizar open shift con esta misma. Es una aplicación muy sencilla en Python donde se utiliza flask para mostrar un mensaje, este código se presenta a continuación.

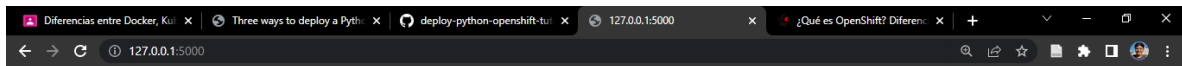
A screenshot of a code editor with two tabs: 'main.py' and '*REPL* [python]'. The 'main.py' tab is active and shows a Python script. The script imports Flask, creates an app, defines a route for '/' that returns a message, and runs the app if the script is executed directly. The code is as follows:

```
1 from flask import Flask
2
3 app = Flask(__name__)
4
5 @app.route("/")
6 def main():
7     return 'Ejemplo OpenShift Este es un ejemplo'
8
9 if __name__ == "__main__":
10    app.run()
```

En esta primera parte del ejemplo vemos el programa de Python, este archivo main tiene la función de crear un servidor local para nuestra aplicación, donde se verá escrito una frase, para esto utilicé la librería de Python llamada flask, esta librería es buena para desarrollo de aplicaciones o páginas web.

```
* Serving Flask app "main" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```

Al correr el programa, se nos arroja esta información de que nuestra página web se está corriendo correctamente en el servidor local 5000.



Ejemplo OpenShift Este es un ejemplo

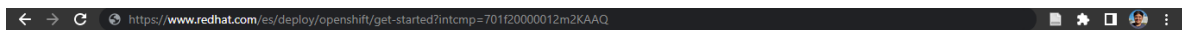


Ahora debemos utilizar Docker para crear una imagen con nuestra aplicación, haciendo los pasos que hemos realizado en las tareas anteriores donde necesitábamos una imagen de Docker también.

```
1 from alpine:latest
2 RUN apk add --no-cache py3-pip \
3     && pip3 install --upgrade pip
4
5 WORKDIR /app
6 COPY . /app
7
8 RUN pip3
9
10 EXPOSE 5000
11
12 ENTRYPOINT ["python3"]
13 CMD ["main.py"]
```

Con estos pasos ya realizados, debemos ingresar a openshift donde indicaremos en la terminal de openshift la imagen de Docker a desplegar, en este caso utilizaremos la nuestra, para esto debemos crear una instancia.

Al final tendremos el mismo resultado, con nuestra aplicación desplegada:

A screenshot of a web browser window. The address bar shows the URL: <https://www.redhat.com/es/deploy/openshift/get-started?intcmp=701f20000012m2KAAQ>. The page content is mostly blank, suggesting the main content area is not fully loaded or is a placeholder.

Ejemplo OpenShift Este es un ejemplo



Link de supositorio en GitHub para revisar el código:

Conclusiones

Este programa me resultó muy similar al que realizamos en la clase anterior, y gracias a eso pude comprender de mejor manera lo que debía de realizar en esta tarea asignada.

Me pareció muy interesante realizar la tarea de esta semana ya que en mi caso lo que hice fue correr una aplicación con una nueva herramienta y que es muy similar a lo que hemos estado probando semana tras semana.

Creo que es muy similar el resultado final que el usuario puede llegar a observar al momento de utilizar alguna de estas formas para correr una aplicación, mínimo de manera local, pero creo que por lo que vi en internet, utilizar alguna de estas herramientas es mejor, ya que administra de mejor manera nuestra aplicación y algunos errores que pueden llegar a suceder.

Pienso que al utilizar todas estas herramientas y probar una por una nos ayuda a conocer mejor cada una de estas, además de poderlas comparar entre ellas y ver las ventajas que nos pueden ofrecer cada una.

Por último, espero que se avance el desarrollo de todas estas herramientas para que en un futuro sea más fácil de utilizar y de comprender, además de que se puedan hacer más cosas con estas y que pueda ayudar a muchos programadores en proyectos importantes o incluso a estudiantes como nosotros en sus tareas y proyectos.

Referencias

IBM Developer. (s. f.). IBM. Recuperado 28 de abril de 2022, de <https://developer.ibm.com/tutorials/deploy-python-app-to-openshift-cluster-source-to-image/>