

Workshop 3

Nombre Estudiante: Emmanuel Bolaños Ocampo

Código Estudiante: A00309828

Repositorio: <https://github.com/emmanuel8664/sd-workshop3/blob/master/Informe.md>

Universidad ICESI

Curso: Sistemas Distribuidos

Docente: Daniel Barragán C.

Tema: Introducción a Docker

Correo: daniel.barragan at correo.icesi.edu.co

Objetivos

Realizar de forma autónoma el aprovisionamiento automático de infraestructura Diagnosticar y ejecutar de forma autónoma las acciones necesarias para lograr infraestructuras estables

Prerrequisitos

Docker Imagen de Python 3.6 de dockerhub Repositorio de paquetes local de pip Descripción

Deberá realizar el aprovisionamiento de un ambiente compuesto por los siguientes elementos: un servidor web (puede emplear apache+php o crear un servicio web con el lenguaje de su preferencia) y un servidor de base de datos (postgresql, mysql ó mariadb). Se debe probar el funcionamiento de la conexión entre los servicios web y base de datos a través de una aplicación web que realice consultas a la base de datos

Actividades

Realice el despliegue de un repositorio de paquetes de pip

Creación de volumen que almacenará los paquetes:

```
docker volume create pypi_packages
```

Se llenan los paquetes de este volumen con un contenedor de centos auxiliar

```
docker run -it --name=centos -v pypi_packages:/lib/python2.7/site-packages/centos bash
```

Puede crear un repositorio de paquetes local usando la imagen en el siguiente enlace: <https://hub.docker.com/r/janlo/pypi-mirror-nginx/>

```
docker run -d -p 80:80 \
-v pypi_packages:/web \
-e PYPI_SERVER_NAME=pypi-mirror \
```

```
janlo/pypi-mirror-nginx
```

Para instalar mas paquetes debe ingresar al contenedor centos usando el comando "docker exec -it centos bash" y ejecutar los siguientes comandos:

```
curl "https://bootstrap.pypa.io/get-pip.py" -o "get-pip.py"
```

```
python get-pip.py
```

```
pip install redis
```

```
pip install flask
```

Escriba un archivo Dockerfile donde realice el despliegue de la aplicación proporcionada en clase, incluya comentarios donde explique las líneas del archivo Dockerfile

Servidor Base de datos.

```
# Comando para utilizar imagen publica de mysql
```

```
FROM mysql
```

```
# Asignación de la variable de ambiente que hace referencia a la clave del usuario root
```

```
ENV MYSQL_ROOT_PASSWORD my-secret-pw
```

```
# En la creación del esquema de datos para mysql, se debe pasar el archivo .sql y guardarlo en la carpeta "docker-entrypoint-initdb.d"
```

```
ADD ./conf/schema.sql /docker-entrypoint-initdb.d/schema.sql
```

Servidor Web

```
# Comando para utilizar imagen publica de php, versión 7.0-apache
```

```
FROM php:7.0-apache
```

```
# Instalación de pdo_mysql para conectarse remotamente con paquetes
```

```
RUN docker-php-ext-install pdo pdo_mysql
```

```
# Se agrega el archivo "index.php"
```

```
ADD ./pages /var/www/html
```

```
# Se expone el puerto 80
```

```
EXPOSE 80
```

```
# Se inicia el servicio de apache2 para acceder a la pagina web desde un navegador
```

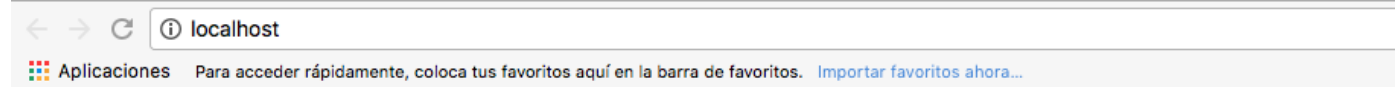
```
CMD service apache2 start && tail -f /var/log/apache2/access.log
```

Incluya los comandos para:

Construir la imagen del contenedor docker build -t apache-php .

Ejecutar el contenedor. docker run -d -p 80:80 --name=apache-php apache-php

Incluya evidencias gráficas que muestren el funcionamiento de lo solicitado



CONSULTA DE PRUEBA