

"Docker: una solución al transporte de aplicaciones"

Mauricio Rigoberto Martínez Romero

-Bachiller-

rigobertomauriciomartinez@gmail.com

**Facultad Multidisciplinaria Oriental,
Universidad de El Salvador, El Salvador**

Indice

OBJETIVOS.....	3
REQUISITOS MÍNIMOS DE HARDWARE.....	3
REQUISITOS DE SOFTWARE.....	3
CONOCIMIENTOS PREVIOS.....	3
INTRODUCCIÓN.....	3
DESARROLLO TEÓRICO.....	3
DESARROLLO PRÁCTICO.....	3
RESULTADOS ESPERADOS.....	3
MATERIAL DE APOYO.....	3
BIBLIOGRAFÍA.....	3

OBJETIVOS

- **Conocer Docker y todos los componentes básicos del mismo. Junto con sus beneficios.**
- **Implementar los conocimientos obtenidos dentro del taller, para la elaboración de una pequeña aplicación PHP y su implementación con Docker.**

REQUISITOS MÍNIMOS DE HARDWARE

- **Procesador:** procesador de doble núcleo con una frecuencia de reloj de 2 GHz o superior
- **RAM:** 4 GB o más.
- **Requisitos adicionales:** 2GB de Swap

REQUISITOS DE SOFTWARE

- **SO:** Distribución GNU/LINUX debian de 32-64 bits otro distributivo compatible con la versión del núcleo 3.10 o posterior
- **Requisitos adicionales:**
 - **Docker:** versión **1.4.1** o superior

CONOCIMIENTOS PREVIOS

- **Uso de la terminal de GNU/Linux y comandos básicos, para la instalación de lo necesario para realizar el taller.**
- **Haber realizado algún proyecto de desarrollo en cualquier lenguaje junto con el manejo de base de datos.**

INTRODUCCIÓN

En el presente taller se hará uso del Software Docker, el cual funciona en un entorno GNU/LINUX y en entorno Windows con las configuraciones necesarias, Docker nos ayuda para poder resolver el problema de transferencia de aplicaciones entre sistemas operativos, o mejor dicho en nuestros entornos de trabajo, ya que no todos los entornos están configurados de la misma forma, o las mismas versiones de librerías necesarias, lo que muchas veces ocasiona conflictos en un grupo de desarrollo en distintos entornos, Docker es un sistema de virtualización, pero no el que todos conocemos, que es creando una maquina virtual e instalando otro sistema operativo, si no que es uno que hace uso del kernel, haciendo así mas eficiente la implementación de la aplicación en otro entorno. Tiene un gran potencial, y muchos profesionales aseguran que usar Docker es el futuro en el desarrollo de aplicaciones. Docker puede hacer uso de la nube, y las grandes empresas como Amazon están trabajando con este tipo de tecnología, para hacerse una idea del porvenir de esta gran herramienta.

DESARROLLO TEÓRICO

¿Qué es Docker?

La idea detrás de Docker es crear contenedores ligeros y portables para las aplicaciones software que puedan ejecutarse en cualquier máquina con Docker instalado, independientemente del sistema operativo que la máquina tenga por debajo, facilitando así también los despliegues.

Algo auto contenido en sí, que se puede llevar de un lado a otro de forma independiente, es portable.

Ahora, volviendo al software, para que podamos acceder como usuarios normales a una aplicación, dicha aplicación software necesita estar ejecutándose en una máquina, en un ordenador. Pero además, dependiendo del tipo de aplicación, dicho ordenador también necesita tener instaladas una serie de cosas para que la aplicación se ejecute correctamente: cierta versión de Java instalado, un servidor de aplicaciones (p.e tomcat, que es el software que realmente estará ejecutando mi aplicación y haciendo que pueda interactuar con ella).

Docker, me permite meter en un contenedor (“una caja”, algo auto contenido, cerrado) todas aquellas cosas que mi aplicación necesita para ser ejecutada (java, Maven, tomcat...) y la propia aplicación. Así yo me puedo llevar ese contenedor a cualquier máquina que tenga instalado Docker y ejecutar la aplicación sin tener que hacer nada más, ni preocuparme de qué versiones de software tiene instalada esa máquina, de si tiene los elementos necesarios para que funcione mi aplicación, de si son compatibles...

DESARROLLO PRÁCTICO

El taller consistirá en llevar de la mano la teoría con lo práctico, es decir, dar conceptos teóricos de los elementos de Docker y posteriormente llevarlo a la práctica, para el taller se estima el siguiente cronograma de contenido, considerando que es el adecuado y suficiente para dar una introducción a Docker, para incentivar al participante del taller a que investigue más por su cuenta y le resulte más motivante y divertido aprender más sobre esta herramienta;

- **¿Qué es Docker?**
- **Contenedores vs Máquinas Virtuales**
- **Ventajas de trabajar con Docker**
- **Arquitectura de Docker**
- **Instalación de Docker**
- **Componentes de Docker**
- **Principales comandos de Docker**
- **Primeros Pasos**
 - La primera imagen
 - Comandos Básicos
 - Construir imágenes desde Dockerfiles
 - Docker-Machine
 - Docker-Compose
- **Docker y el cloud computing**
- **Introducción a DockerHub**
 - Uso de imágenes existentes
- **Ejercicio usando los conocimientos adquiridos.**

RESULTADOS ESPERADOS

En los resultados esperados es utilizar los conocimientos adquiridos dentro del taller, para poder realizar proyectos utilizando Docker para realizar pruebas, en distintos entornos de desarrollo, sin que estos afecten el funcionamiento de nuestra aplicación. Y tener mas opciones para la virtualización.

MATERIAL DE APOYO

Existen grandes cantidades de información en Internet para Docker, asi que no te costará obtener mas información a cerca de esta interesante herramienta.

BIBLIOGRAFÍA

- <https://docs.docker.com/>
- <https://www.docker.com/>