Contenido

[1. Autores del trabajo, planificación y entrega 3](#_Toc482052043)

[1.1 Autores 3](#_Toc482052044)

[1.2 Planificación 3](#_Toc482052045)

[1.3 Entrega 3](#_Toc482052046)

[2. Requisitos del prototipo a implementar 4](#_Toc482052047)

[2.1 Requisitos funcionales 4](#_Toc482052048)

[2.2 Otros requisitos 4](#_Toc482052049)

[3. Criterios de comparación en la implementación 5](#_Toc482052050)

[3.1 Criterio 1: *Facilidad de instalación.* 5](#_Toc482052051)

[3.2 Criterio 2: *Búsqueda de imágenes en un repositorio público.* 5](#_Toc482052052)

[3.3 Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.* 5](#_Toc482052053)

[3.4 Criterio 4: *Velocidad de ejecución de un contenedor.* 5](#_Toc482052054)

[3.5 Criterio 5: *Facilidad para la creación de contenedores.* 5](#_Toc482052055)

[3.6 Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.* 5](#_Toc482052056)

[3.7 Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.* 6](#_Toc482052057)

[3.8 Criterio 8: *Facilidad para ver los contenedores generados.* 6](#_Toc482052058)

[3.9 Criterio 9: *Facilidad para borrar un contenedor.* 6](#_Toc482052059)

[3.10 Criterio 10: *Facilidad para borrar una imagen.* 6](#_Toc482052060)

[4. Proyecto de implementación de un prototipo del sistema utilizando Docker 7](#_Toc482052061)

[Criterio 1: *Facilidad de instalación.* 7](#_Toc482052062)

[Criterio 2: *Búsqueda de imágenes en un repositorio público.* 9](#_Toc482052063)

[Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.* 9](#_Toc482052064)

[Criterio 4: *Velocidad de ejecución de un contenedor.* 9](#_Toc482052065)

[Criterio 5: *Facilidad para la creación de contenedores.* 10](#_Toc482052066)

[Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.* 10](#_Toc482052067)

[Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.* 11](#_Toc482052068)

[Criterio 8: *Facilidad para ver los contenedores generados.* 11](#_Toc482052069)

[Criterio 9: *Facilidad para borrar un contenedor.* 11](#_Toc482052070)

[Criterio 10: *Facilidad para borrar una imagen.* 11](#_Toc482052071)

[5. Proyecto de implementación de un prototipo del sistema utilizando la tecnología B 12](#_Toc482052072)

[Criterio 1: *Facilidad de instalación.* 12](#_Toc482052073)

[Criterio 2: *Búsqueda de imágenes en un repositorio público.* 13](#_Toc482052074)

[Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.* 13](#_Toc482052075)

[Criterio 4: *Velocidad de ejecución de un contenedor.* 13](#_Toc482052076)

[Criterio 5: *Facilidad para la creación de contenedores.* 13](#_Toc482052077)

[Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.* 13](#_Toc482052078)

[Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.* 14](#_Toc482052079)

[Criterio 8: *Facilidad para ver los contenedores generados.* 14](#_Toc482052080)

[Criterio 9: *Facilidad para borrar un contenedor.* 14](#_Toc482052081)

[Criterio 10: *Facilidad para borrar una imagen.* 14](#_Toc482052082)

[6. Comparación de las dos implementaciones 15](#_Toc482052083)

[6.1 Evaluación de los criterios en la implementación usando Docker 15](#_Toc482052084)

[6.2 Evaluación de los criterios en la implementación usando OpenVZ 15](#_Toc482052085)

[7. Comparación de la implementación de las tecnologías 16](#_Toc482052086)

[8. Conclusiones 19](#_Toc482052087)

# 1. Autores del trabajo, planificación y entrega

## 1.1 Autores

En este apartado se debe indicar el número de grupo y los nombres de los autores, poniendo en primer lugar al coordinador del grupo.

## 1.2 Planificación

[Enlace planificación GanttPro](https://app.ganttpro.com/shared/token/816e68e5218bd8bde90bc277910b515ffe461a37930fd3bf9043e946b85c20d8/171184)

## 1.3 Entrega

[Enlace repositorio GitHub](https://github.com/DiegoUAH/TG3)

# 2. Requisitos del prototipo a implementar

## 2.1 Requisitos funcionales

|  |  |
| --- | --- |
| REQ. | DESCRIPCIÓN |
| RF01 | Instalación |
| RF02 | Buscar imágenes de contenedores en un repositorio |
| RF03 | Descargar imagen de contenedor de un repositorio |
| RF04 | Ejecutar un contenedor |
| RF05 | Crear un contenedor nuevo |
| RF06 | Instalar una aplicación en un contenedor nuevo |
| RF07 | Ver las imágenes del repositorio local |
| RF08 | Ver los contenedores del sistema |
| RF09 | Borrar un contenedor |
| RF10 | Borrar una imagen |

## 2.2 Otros requisitos

|  |  |
| --- | --- |
| REQ. | DESCRIPCIÓN |
| RSW | **Requisitos Software** |
| RSW01 | Ubuntu v.16.10 64bits |
| RHW | **Requisitos Hardware** |
| RHW01 | Arquitectura de 64 bits |
| RHW02 | Procesador de 1GHz |
| RHW03 | Espacio en disco 20GB |
| RHW04 | Memoria RAM 1GB |
| RHW05 | Acceso a internet |

# 3. Criterios de comparación en la implementación

## 3.1 Criterio 1: *Facilidad de instalación.*

*Nombre del criterio: Facilidad de instalación.*

*Descripción: Facilidad y rapidez de la instalación, incluyendo la facilidad para encontrar documentación para llevarla a cabo.*

*Tipo de valor: Numérico.*

## 3.2 Criterio 2: *Búsqueda de imágenes en un repositorio público.*

*Nombre del criterio: Búsqueda de imágenes en un repositorio público.*

*Descripción: Facilidad para encontrar imágenes compartidas en un repositorio público.*

*Tipo de valor: Numérico.*

## 3.3 Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.*

*Nombre del criterio: Tiempo de descarga de una imagen desde un repositorio.*

*Descripción: Valoración del tiempo que tarda en descargar una imagen en relación de su peso.*

*Tipo de valor: Numérico.*

## 3.4 Criterio 4: *Velocidad de ejecución de un contenedor.*

*Nombre del criterio: Velocidad de ejecución de un contenedor.*

*Descripción: Velocidad de ejecución de un contenedor (medido en segundos).*

*Tipo de valor: Numérico.*

## 3.5 Criterio 5: *Facilidad para la creación de contenedores.*

*Nombre del criterio: Facilidad para la creación de contenedores.*

*Descripción: Facilidad a la hora de crear un nuevo contenedor.*

*Tipo de valor: Numérico.*

## 3.6 Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.*

*Nombre del criterio: Facilidad de instalación de aplicaciones en un contenedor.*

*Descripción: Facilidad para instalar una aplicación en un contenedor y comprobar la independencia del contenedor con el sistema anfitrión.*

*Tipo de valor: Numérico.*

## 

## 3.7 Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.*

*Nombre del criterio: Facilidad para localizar las imágenes almacenadas en el repositorio local.*

*Descripción: Facilidad para encontrar todas las imágenes que tenemos almacenadas en el repositorio local.*

*Tipo de valor: Numérico.*

## 3.8 Criterio 8: *Facilidad para ver los contenedores generados.*

*Nombre del criterio: Facilidad para ver los contenedores generados.*

*Descripción: Facilidad para encontrar los contenedores que tenemos.*

*Tipo de valor: Numérico.*

## 3.9 Criterio 9: *Facilidad para borrar un contenedor.*

*Nombre del criterio: Facilidad para borrar un contenedor.*

*Descripción: Facilidad para borrar un contenedor.*

*Tipo de valor: Numérico.*

## 3.10 Criterio 10: *Facilidad para borrar una imagen.*

*Nombre del criterio: Facilidad para borrar una imagen.*

*Descripción: Facilidad para borrar una imagen.*

*Tipo de valor: Numérico.*

# 4. Proyecto de implementación de un prototipo del sistema utilizando Docker

## Criterio 1: *Facilidad de instalación.*

Para obtener la última versión, instalamos Docker desde el repositorio oficial. Para ello, seguiremos estos pasos:

Actualizar la base de datos de repositorios:

sudo apt-get update

Agregar la llave GPG del repositorio oficial de Docker a nuestro sistema:

sudo apt-key adv --keyserver hkp://p80.pool.sks-keyservers.net:80 --recv-keys 58118E89F3A912897C070ADBF76221572C52609D

Y agregar el repositorio de Docker:

echo "deb https://apt.dockerproject.org/repo ubuntu-xenial main" | sudo tee /etc/apt/sources.list.d/docker.list

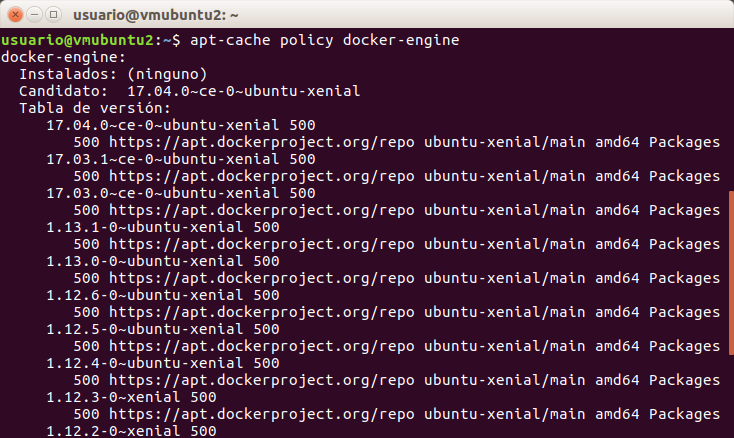
Actualizar la base de datos de repositorios una vez más:

sudo apt-get update

Por último nos aseguraremos de bajar Docker del repositorio oficial y no del de ubuntu:

apt-cache policy docker-engine

Tendríamos que ver algo similar a esto:



En la salida del comando vemos que docker-engine no está instalado pero en caso de hacerlo lo haremos desde el repositorio de Docker y no del de Ubuntu.

Finalmente instalar docker:

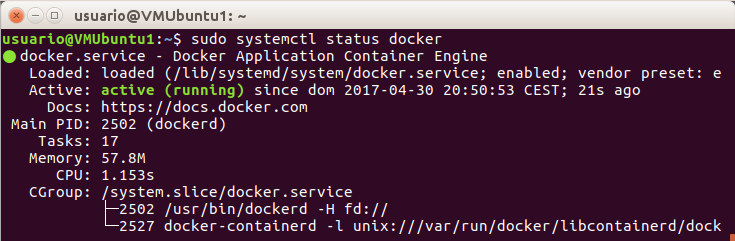
sudo apt-get install docker-engine

Docker ahora debe estar instalado, el daemon iniciado, y el proceso habilitado para iniciar en el arranque.

Comprobar que se está ejecutando:

sudo systemctl status docker

Lo que tendría que devolver algo similar a esto:

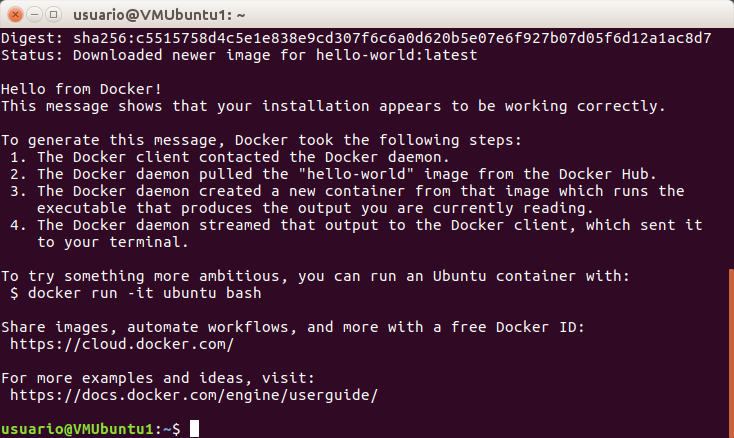


La instalación de Docker ofrece no sólo el servicio docker (daemon), sino también la utilidad de línea de comandos docker o el cliente docker.

Para confirmar que podamos acceder y descargar imágenes desde el Hub de Docker usaremos el comando:

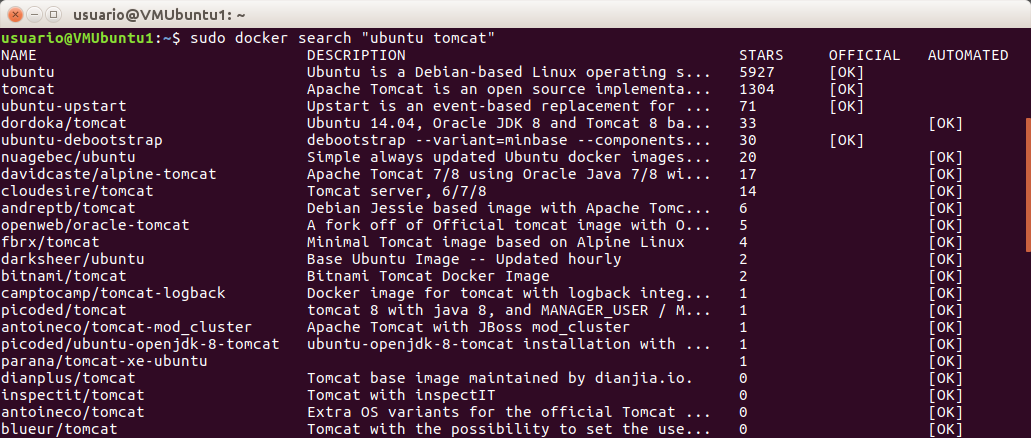
sudo docker run hello-world

Este comando descarga una imagen de prueba y ejecuta un contenedor con dicha imagen. Nos debería devolver algo similar a esto:



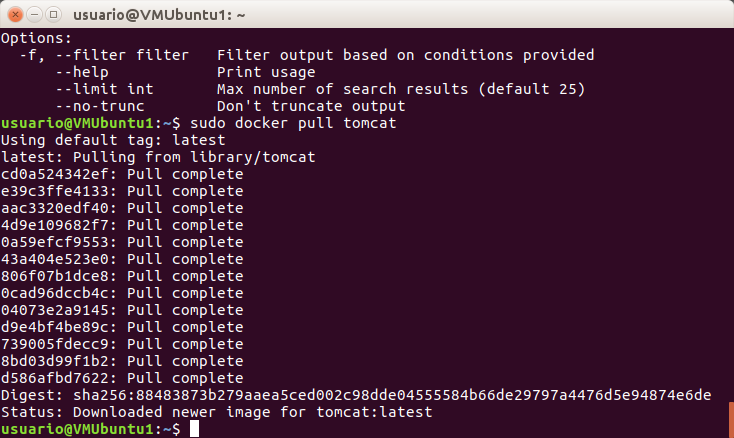
## Criterio 2: *Búsqueda de imágenes en un repositorio público.*

*Comando*: sudo docker search “nombre\_imagen”



## Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.*

*Comando*: sudo docker pull “nombre\_imagen”



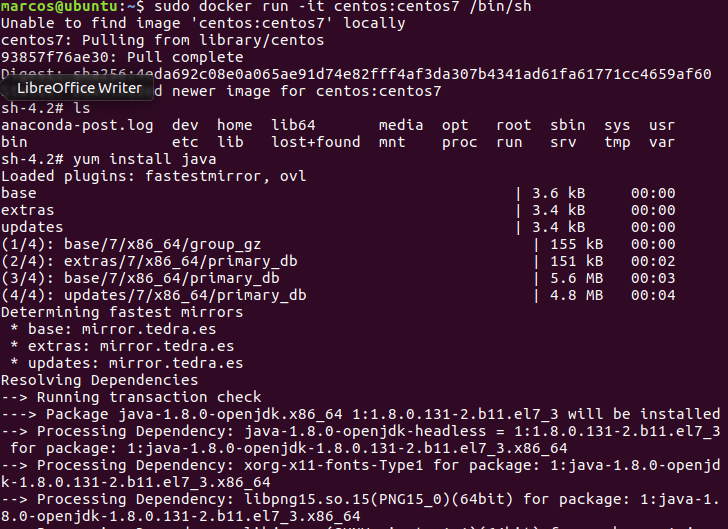
## Criterio 4: *Velocidad de ejecución de un contenedor.*

*Comando*: sudo docker start –i nombre\_imagen

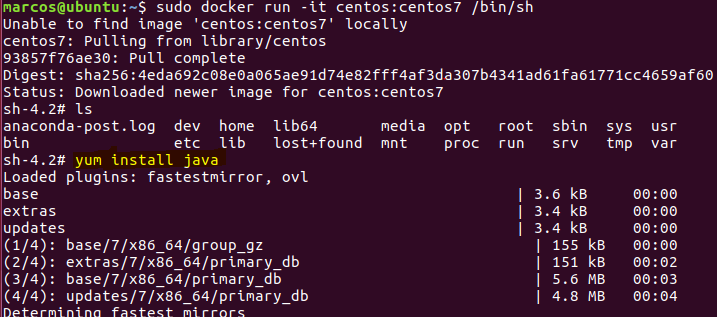


## Criterio 5: *Facilidad para la creación de contenedores.*

*Comando*: sudo docker run –it nombre\_imagen

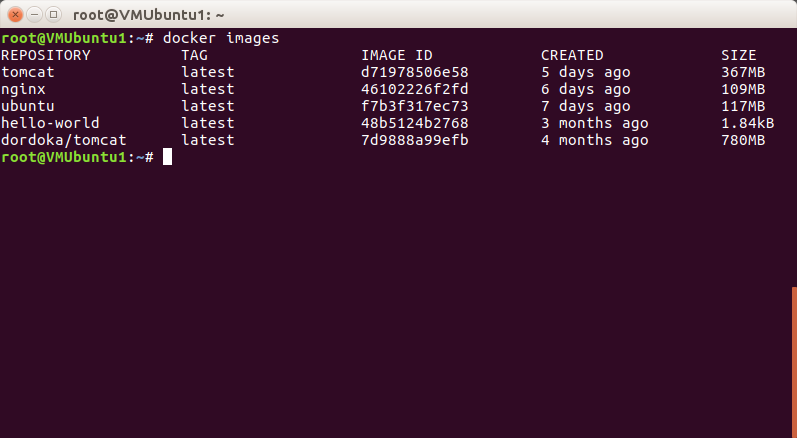


## Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.*



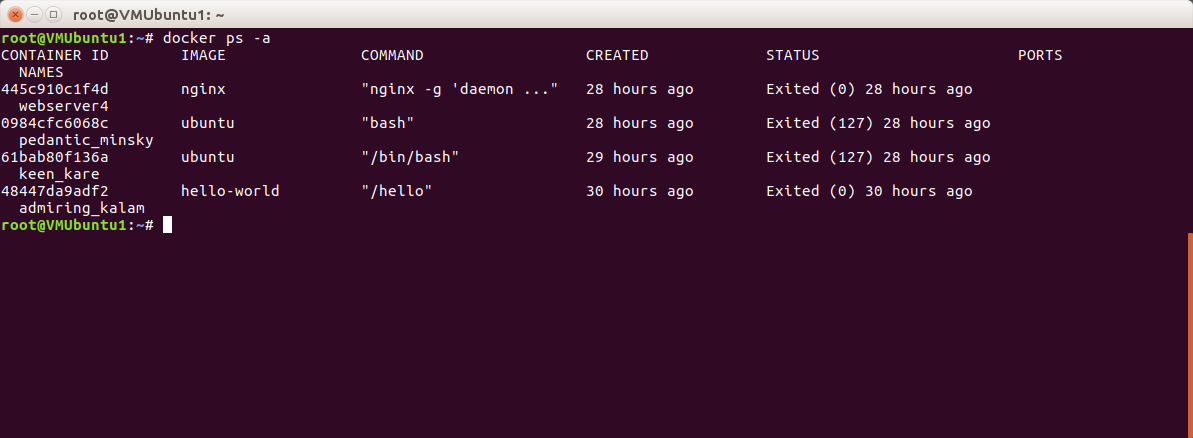
## Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.*

*Comando*: sudo docker images



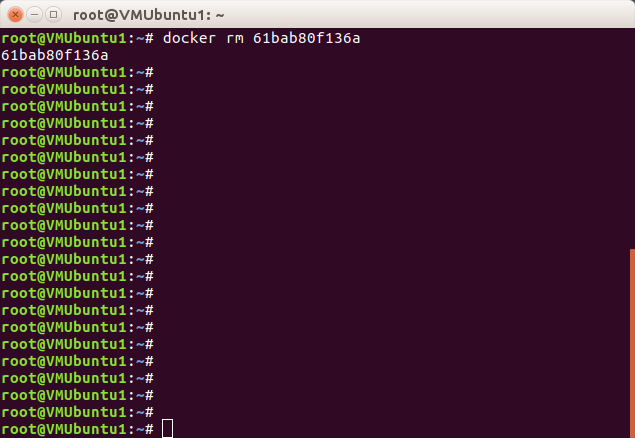
## Criterio 8: *Facilidad para ver los contenedores generados.*

*Comando*: sudo docker ps -a



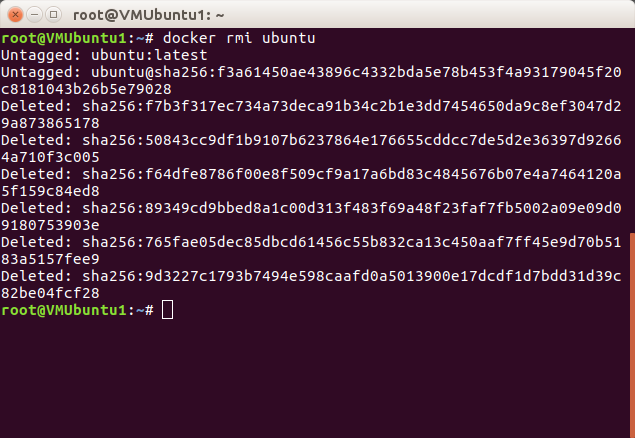
## Criterio 9: *Facilidad para borrar un contenedor.*

*Comando*: sudo docker rm id\_contenedor (o nombre contenedor)



## Criterio 10: *Facilidad para borrar una imagen.*

*Comando*: sudo docker rmi id\_imagen (o nombre imagen)



# 5. Proyecto de implementación de un prototipo del sistema utilizando la tecnología B

## Criterio 1: *Facilidad de instalación.*

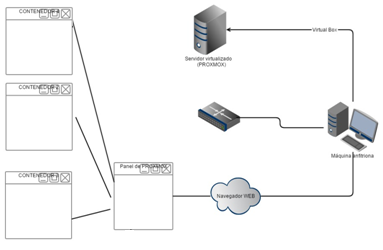
Para la instalación se realizan los siguientes pasos:

Instalación de Oracle Virtual Box, herramienta gratuita de virtualización, cabe reseñar que bien es cierto que la instalación de OpenVZ se realiza sobre Linux, pudiendo instalar en el disco duro principal y como máquina anfitriona la propia, pero en este caso optamos por la virtualización.

Una vez instalado, con el *wizard* de instalación, procederemos a instalar y montar una imagen ISO de PROXMOX 4.4, esta imagen es una distribución de Linux capaz de construir contenedores OpenVZ y además de crear máquinas virtuales a nivel de kernel. Para instalar PROXMOX, en la barra de herramientas de virtual box seleccionamos crear nueva máquina, ajustamos los parámetros de RAM (en nuestro caso 3072 MB 🡪 3GB) y el espacio de disco duro(50 GB en nuestro caso), una vez creada la máquina virtual seleccionamos en su configuración en la pestaña de almacenamiento y seleccionamos en el panel de la derecha disco vacío para procedes a buscar la ruta de la imagen previamente descargada de PROXMOX 4.4, una vez indicada la ruta procedemos, seleccionamos la opción de red en el menú de configuración, cambiamos el tipo de NAT a Puentey ya podemos arrancar la máquina virtual de PROXMOX.

El siguiente paso será configurar la instalación de PROXMOX, iniciamos la instalación en el menú de dando enter a install, configuramos región, zona horaria e idioma, posteriormente las credenciales de usuario (el usuario por defecto será root), y acto seguido pasaremos a la pantalla de configuración de la red, por defecto coge los datos del anfitrión, pero podemos configurarla a nuestra exigencias de la red., continuamos y esperamos a que finalice la instalación, posteriormente reiniciamos la máquina

Al reiniciar la máquina accederemos al panel de la terminal de PROXMOX, iniciaremos sesión con las credenciales antes configuradas, cuando el servicio este activo solamente accederemos a través del navegador web de nuestra máquina anfitriona la dirección IP que muestra en la pantalla junto con el puerto, siempre precedido de la conexión seguro https



## Criterio 2: *Búsqueda de imágenes en un repositorio público.*

## Criterio 3: *Tiempo de descarga de una imagen desde un repositorio.*

## Criterio 4: *Velocidad de ejecución de un contenedor.*



* Darle al botón Iniciar.

## Criterio 5: *Facilidad para la creación de contenedores.*

## 

Introducimos una contraseña y, exceptuando Plantilla en la que tienes que descargar una previamente, lo demás se configura solo y no tienes que modificar ningún parámetro para que funcione.

## Criterio 6: *Facilidad de instalación de aplicaciones en un contenedor.*

## Criterio 7: *Facilidad para localizar las imágenes almacenadas en el repositorio local.*

## Criterio 8: *Facilidad para ver los contenedores generados.*

## 

## Criterio 9: *Facilidad para borrar un contenedor.*



* Darle al botón Eliminar e introducir la ID.

## Criterio 10: *Facilidad para borrar una imagen.*

# 6. Comparación de las dos implementaciones

## 6.1 Evaluación de los criterios en la implementación usando Docker

|  |  |
| --- | --- |
| Criterio | Evaluación |
| 1. Facilidad de instalación | 5 |
| 2. Búsqueda de imágenes en un repositorio público | 5 |
| 3. Tiempo de descarga de una imagen desde un repositorio | 3 |
| 4. Velocidad de ejecución de un contenedor | <3 seg. |
| 5. Facilidad para la creación de contenedores | 4 |
| 6. Facilidad de instalación de aplicaciones en un contenedor | 5 |
| 7. Facilidad para localizar las imágenes almacenadas en el repositorio local | 5 |
| 8. Facilidad para ver los contenedores generados | 4 |
| 9. Facilidad para borrar un contenedor | 5 |
| 10. Facilidad para borrar una imagen | 5 |

## 6.2 Evaluación de los criterios en la implementación usando OpenVZ

|  |  |
| --- | --- |
| Criterio | Evaluación |
| 1. Facilidad de instalación | 4 |
| 2. Búsqueda de imágenes en un repositorio público | 5 |
| 3. Tiempo de descarga de una imagen desde un repositorio | 4 |
| 4. Velocidad de ejecución de un contenedor | <3seg. |
| 5. Facilidad para la creación de contenedores | 4 |
| 6. Facilidad de instalación de aplicaciones en un contenedor | 3 |
| 7. Facilidad para localizar las imágenes almacenadas en el repositorio local | 5 |
| 8. Facilidad para ver los contenedores generados | 5 |
| 9. Facilidad para borrar un contenedor | 5 |
| 10. Facilidad para borrar una imagen | 5 |

# 7. Comparación de la implementación de las tecnologías

|  |  |  |  |
| --- | --- | --- | --- |
| Criterios | Docker | OpenVZ | Comentarios |
| 1. Facilidad de instalación | 5 | 4 | La instalación de ambas tecnologías es bastante sencilla, además es fácil encontrar tutoriales que describen paso a paso la instalación. |
| 2. Búsqueda de imágenes en un repositorio público | 5 | 5 | Docker dispone del repositorio Docker Hub donde hay gran cantidad de imágenes disponibles.  La wiki de OpenVZ ofrece un repositorio de imágenes y de plantillas. |
| 3. Tiempo de descarga de una imagen desde un repositorio | 3 | 4 | Los repositorios de ambas tecnologías ofrecen una buena capacidad de descarga, además las imágenes son bastante ligeras lo que facilita una descarga rápida. |
| 4. Velocidad de ejecución de un contenedor | <3 seg. | <3 seg. | La ejecución es casi inmediata en ambas tecnologías. |
| 5. Facilidad para la creación de contenedores | 4 | 4 | Desde la línea de comandos de Docker es bastante sencillo y rápido crear un contenedor.  En el panel de PROXMOX podemos crear contendores de OpenVZ con un wizard de ayuda. |
| 6. Facilidad de instalación de aplicaciones en un contenedor | 5 | 3 | Sencillo y rápido en ambas tecnologías, ya que es como instalar una aplicación en un Ubuntu normal. La aplicación queda aislada del sistema operativo anfitrión. |
| 7. Facilidad para localizar las imágenes almacenadas en el repositorio local | 5 | 5 | Desde la línea de comandos de Docker es bastante sencillo y rápido acceder a un listado con las imágenes almacenadas.  El panel de PROXMOX está muy bien organizado e intuitivo, apareciendo un listado de los contenedores y su estado. |
| 8. Facilidad para ver los contenedores generados | 4 | 5 | Desde la línea de comandos de Docker es bastante sencillo y rápido acceder a un listado con los contenedores generados, además permite separar los que están en ejecución de los que no añadiendo un atributo al comando.  En el panel gráfico aparecen todos los contendores OpenVZ generados. |
| 9. Facilidad para borrar un contenedor | 5 | 5 | Desde la línea de comandos de Docker es bastante sencillo y rápido borrar un contenedor.  Desde el panel grafico se pueden borrar contenedores OpenVZ en un click. |
| 10. Facilidad para borrar una imagen | 5 | 5 | Desde la línea de comandos de Docker es bastante sencillo y rápido borrar una imagen. Solo se pueden borrar las imágenes que no están siendo referenciadas por ningún contenedor, ya estén en ejecución o apagados.  Desde el panel grafico se pueden borrar imágenes de forma sencilla y rápida.. |

# 8. Conclusiones

A partir de la información incluida en el apartado 7 y de la experiencia al realizar el trabajo, el grupo debe estar en condiciones de manifestar su opinión sobre la implementación del sistema utilizando ambas tecnologías, y debe plasmarla en este apartado, indicando las ventajas e inconvenientes más relevantes de utilizar una u otra tecnología para implementar el sistema.

---------------------------

(Hay que cumplir la estructura básica indicada de secciones. Pero si se desea se pueden añadir otras secciones como anexos. Por ejemplo, alguna encuesta de opinión realizada sobre las tecnologías, etc.)