Gestión de Contenedores Dockers

LAB Gestión de Contenedores con Docker en Windows 10 Pro

Objetivos

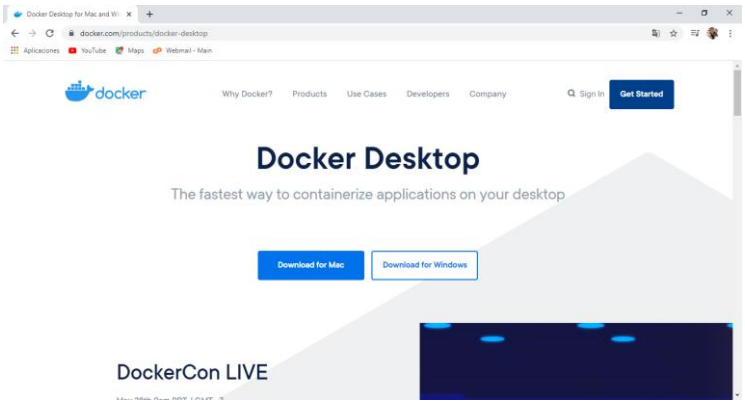
- Mostrar al participante el procedimiento para la instalación del software de gestión de contenedores Docker en Windows

Requisito

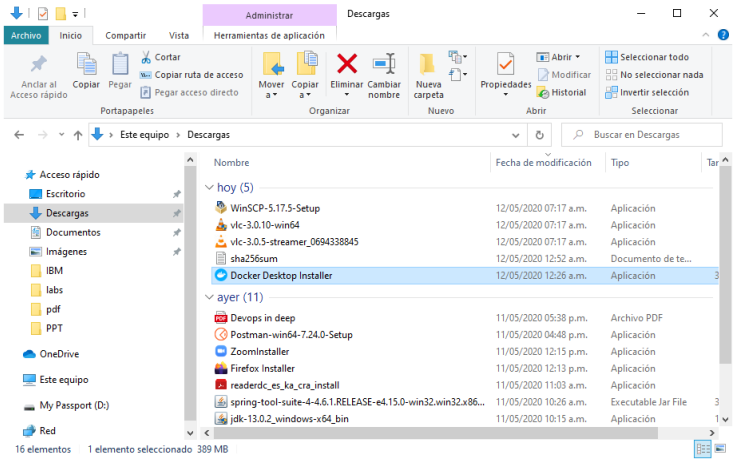
- Tener instalado el sistema operativo Windows 10 Pro/Home

Procedimiento

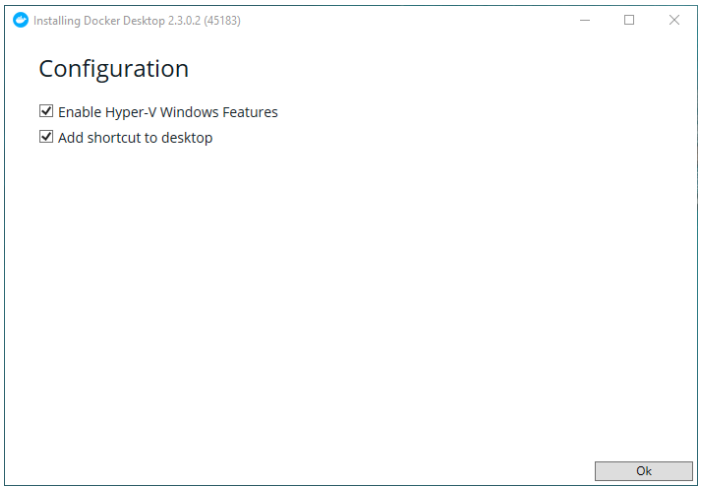
1. Descarga el software Docker para Windows desde https://www.docker.com/products/docker-desktop



2. Ejecuta el instalador "**Docker Desktop Installer**", como administrador.



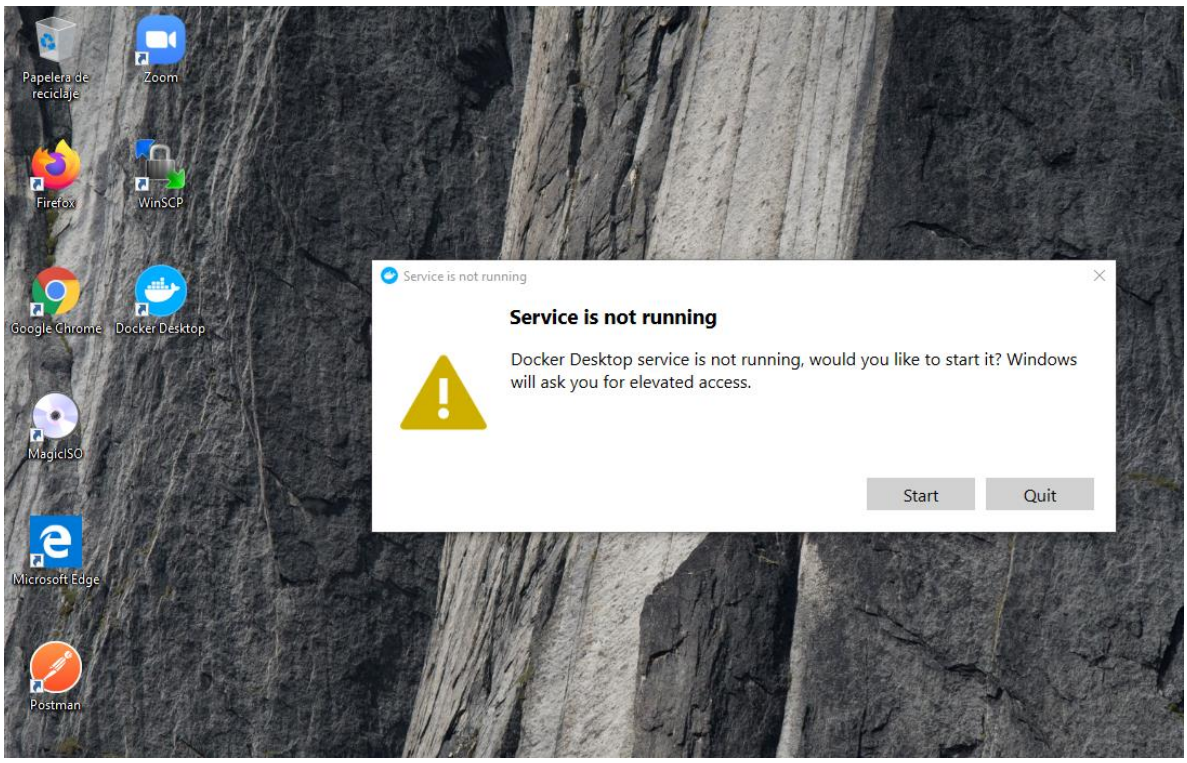
Sigues los pasos del asistente, acepta las opciones por defecto.



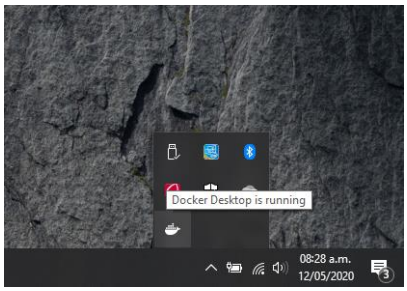
Nota: A diferencia de versiones anteriores al 2020, Docker en Windows ahora tiene soporte de virtualización nativa en Windows con **Hyper-V**.

Espera a que termine la instalación y luego reinicia la PC.

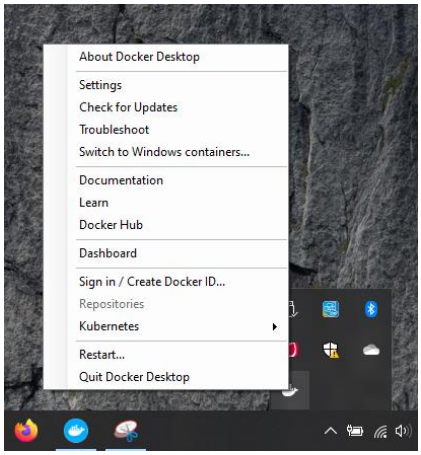
3. Iniciando el servicio Docker. Haz clic en icono del escritorio "**Docker Desktop**" e inicia el servicio de Docker.



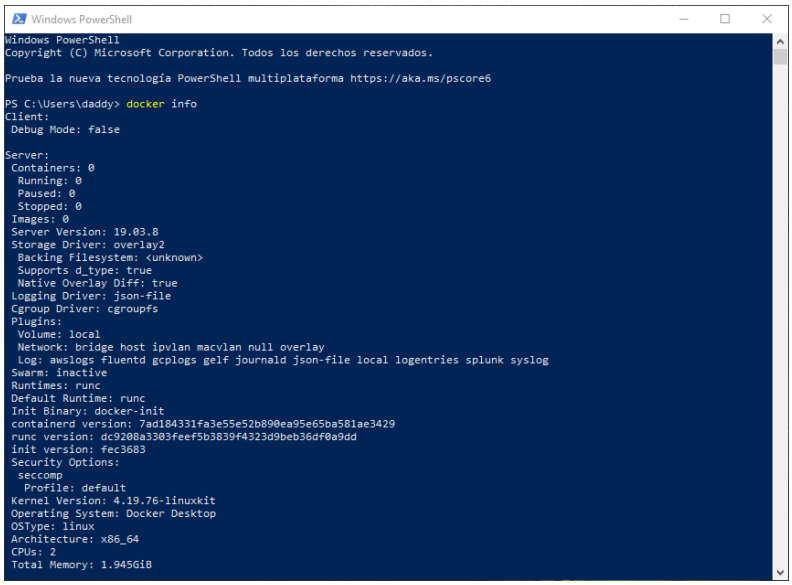
Haz clic en **Start**. Verifica el estado del pasando el puntero del mouse por sobre el icono de Docker.



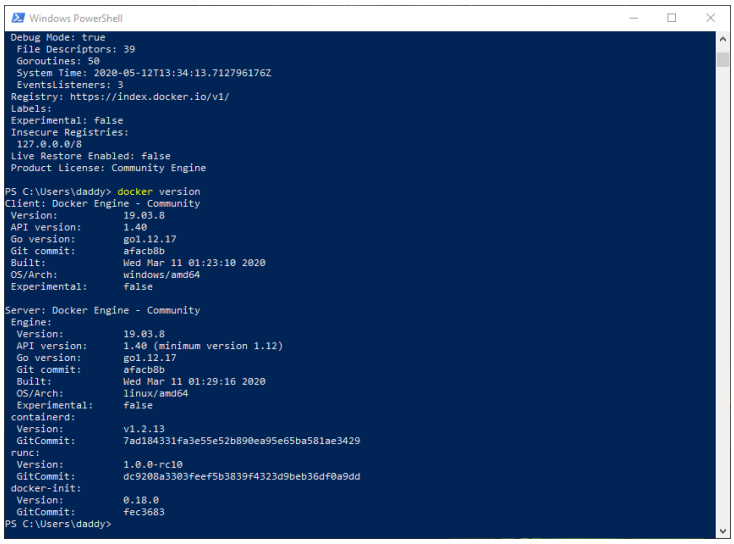
Puedes acceder a la configuración, haciendo clic en el icono Docker en la barra de Windows y seleccionando "**Settings**".



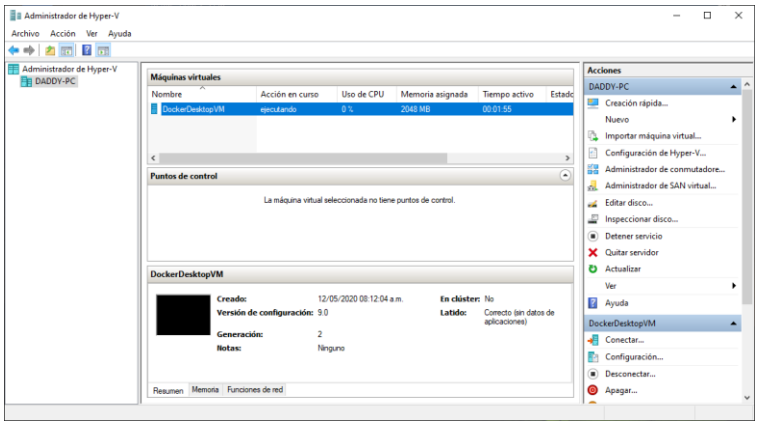
Usando una consola **PowerShell** también podemos conocer el estado de Docker. En la consola ejecuta: C:\> **docker** info



Conociendo la versión de Docker, ejecuta c:\> **docker** version

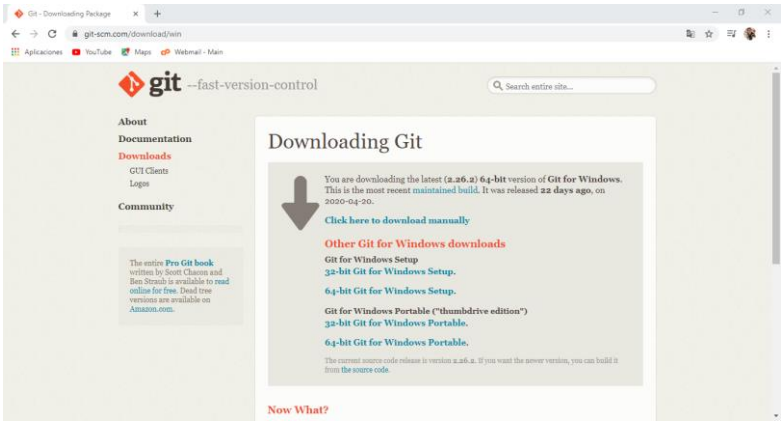


4. Verifica la maquina virtual creada en **Hyper-V**.

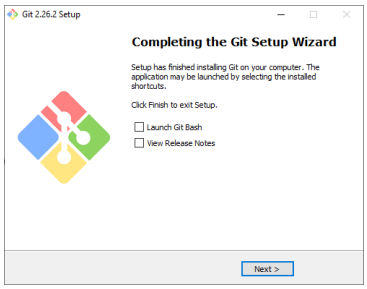


Nota: Durante la instalación se ha creado la máquina virtual "**DockerDesktopVM**" para la ejecución de Docker.

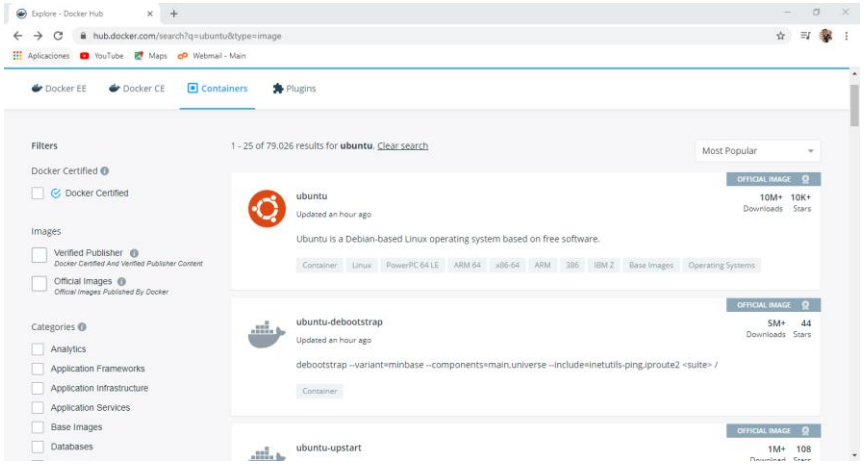
5. Instala Git en Windows. Descarga Git desde https://git-scm.com/download/win



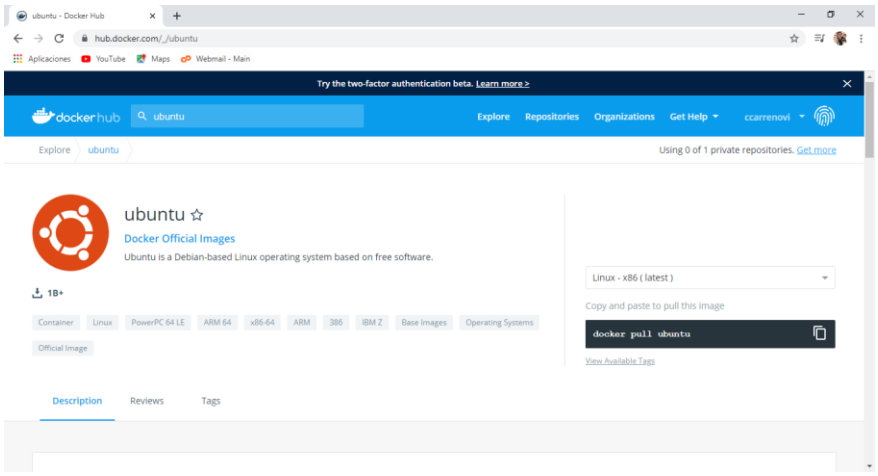
Ejecuta el instalador de Git y acepta todas las opciones por defecto.



6. Gestión de imágenes de Docker. Abre https://hub.docker.com/ y busca la imagen de Ubuntu ubica la imagen oficial.

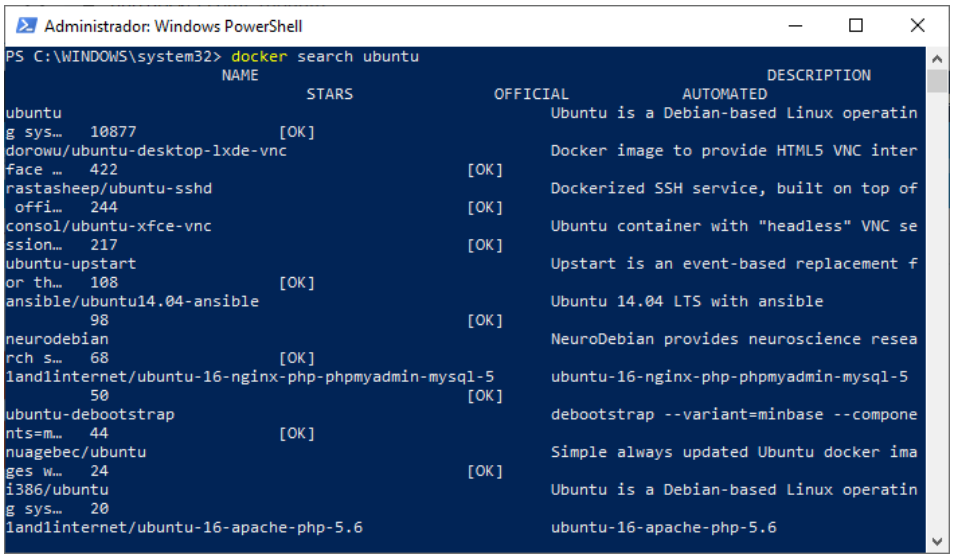


Ingresa al ítem Ubuntu “OFICIAL IMAGE”.

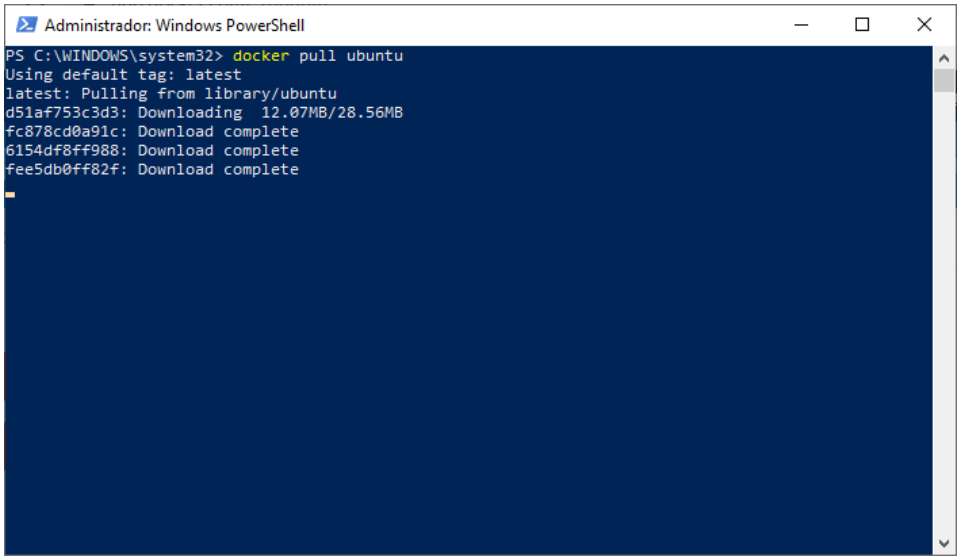


Anota el comando para la descarga de la imagen: **docker pull** ubuntu

7. Abre una consola PowerShell y ejecuta el comando para la descarga de la imagen de Ubuntu. Primero buscamos la imagen con el comando: docker search ubuntu



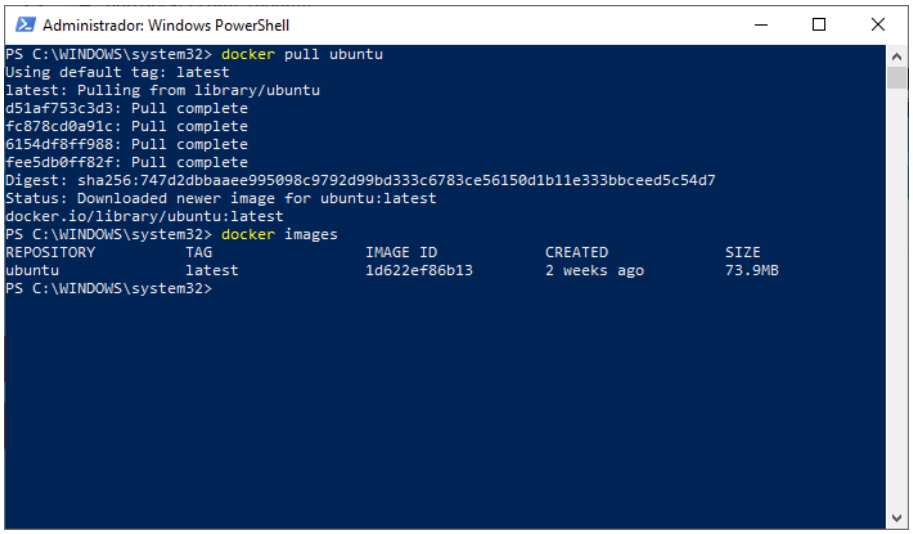
Luego ejecuta: docker pull ubuntu



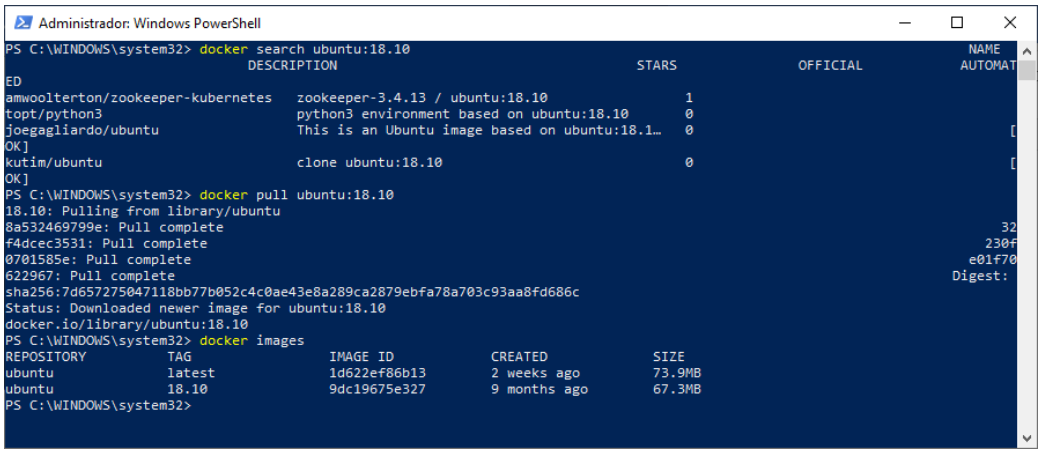
Espera a que descargue la imagen.

8. Verifica las imágenes descargadas. Para listar las imágenes en el sistema ejecuta: docker images

Nota: Observa que cada imagen tiene un identificador llamado “**IMAGE ID**”, el cual contiene un valor hexadecimal único. También el campo “**tag**” es importante en este caso es la ultima version de la imagen “latest”.



9. Usando tags para descargar una version especifica de la imagen. Descarguemos la imagen de ubuntu en version 18.10



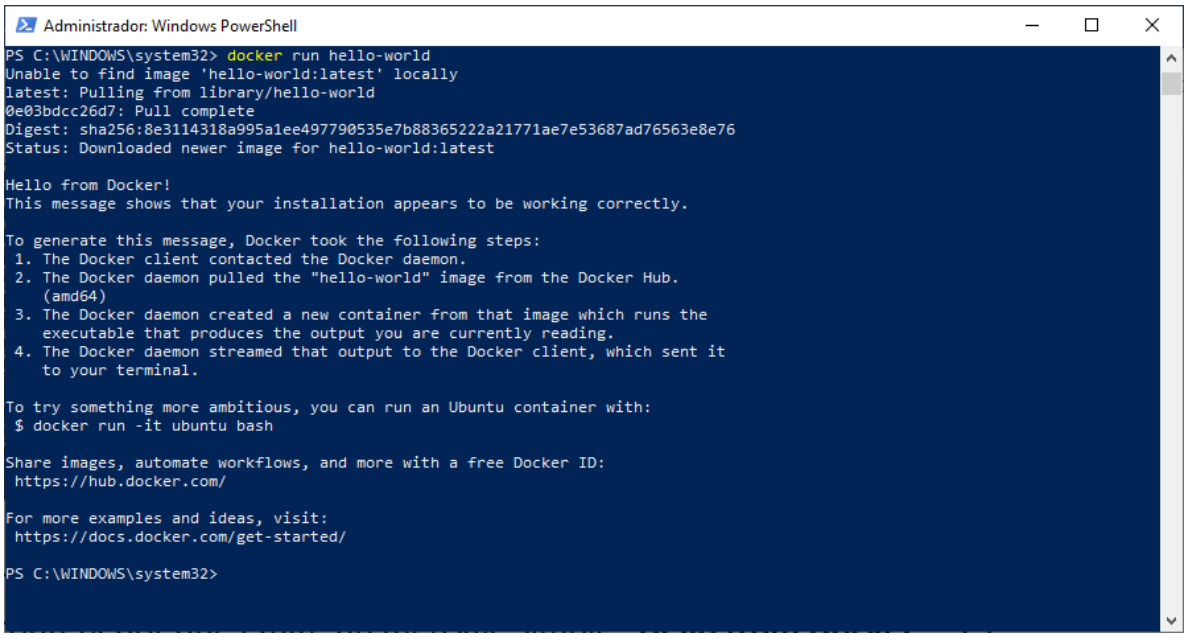
10. Creando nuestro primer contenedor “for Dummy”, será nuestro contenedor hello-world. Crearemos un contenedor que emite un mensaje simple “**Hello from Docker!**”.

    Nota: Es necesario recordar que cuando Docker ejecuta un contenedor y no existe la imagen, el controlador Docker intentara descargar la imagen apropiada.

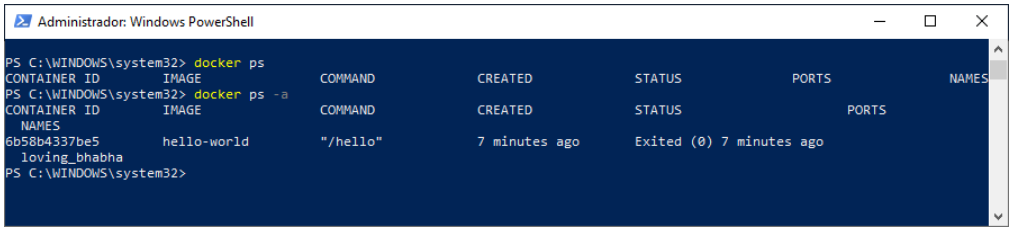
    Para iniciar un contenedor ejecuta el comando C:\> docker run <contenedor-name> .

    Si deseas listar los contenedores que se están corriendo en el sistema, debes ejecutar el comando: C:\> docker ps

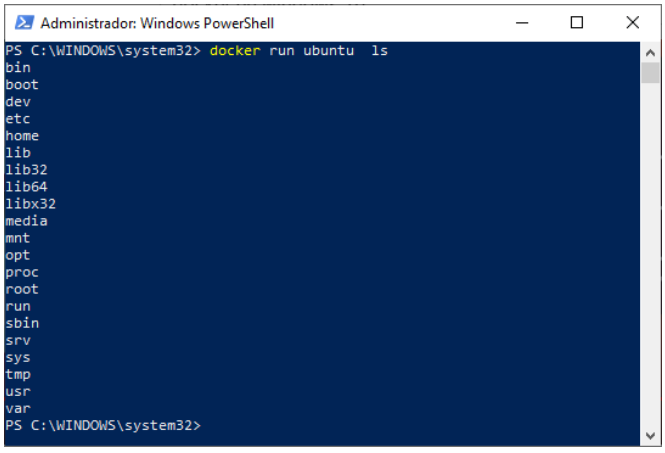
    Algunos contenedores ejecutan una instrucción especifica y terminan el proceso, para listar los procesos de los contenedores incluyendo los que han finalizado ejecuta el comando: C:\> docker ps -a



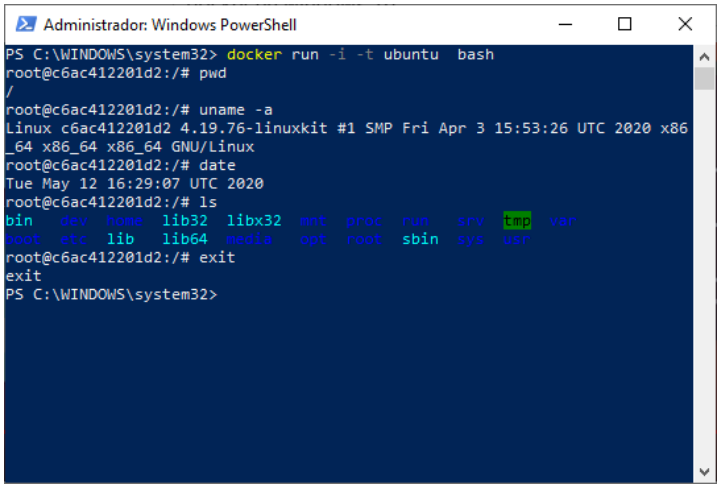
Conociendo el estado de los procesos de los contenedores.



11. Ejecutar comando en contenedores. Para ejecutar comandos en el contenedor debes iniciar un Shell o también puedes ejecutar un comando especifico. Crea un contenedor usando la imagen de ubuntu.



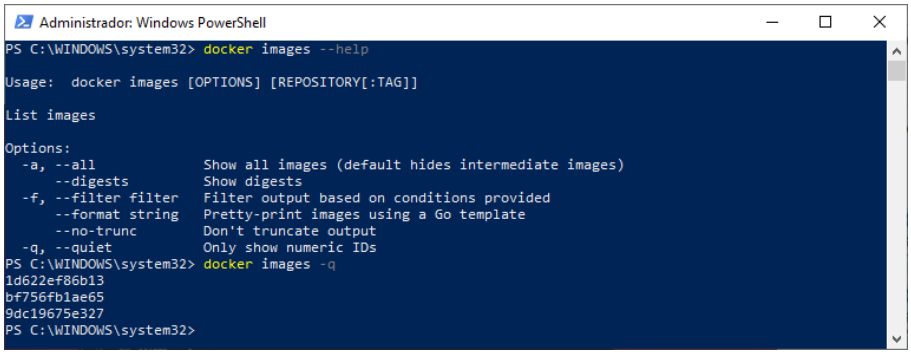
Este contenedor ejecuta el comando ls (muestra el contenido de los archivos y directorios) y termina.



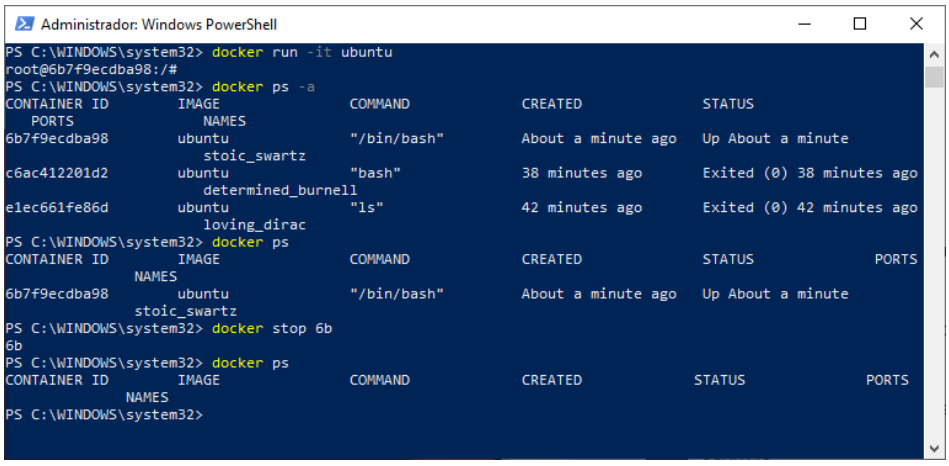
En este caso creamos un contenedor abriendo un terminal y ejecutando el Shell **bash** de ubuntu.

12. Eliminando un contenedor. Para eliminar un contenedor ejecuta el comando c:\> docker rm <id contenedor>

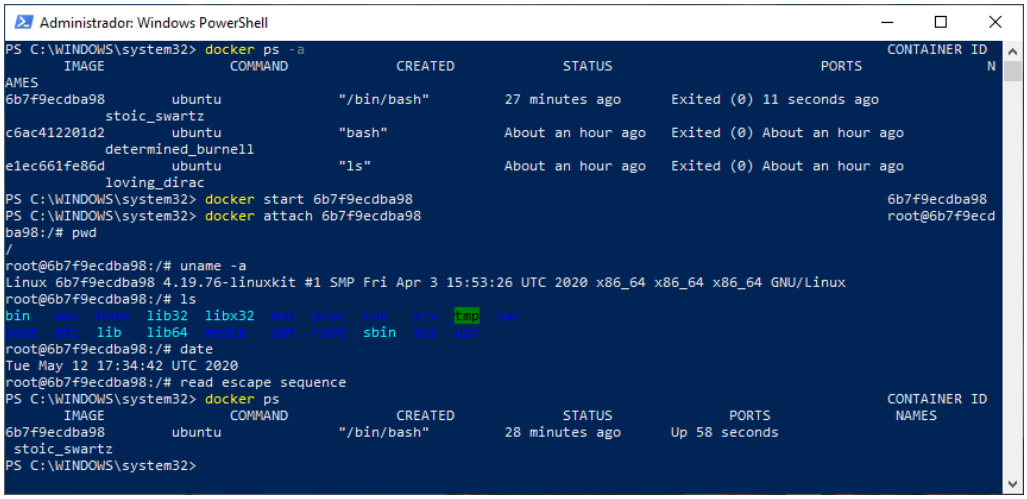
13. Mostrando la ayuda de los comandos en Docker. Para mostrar la ayuda de los comandos ejecuta el comando con la opción **--help**



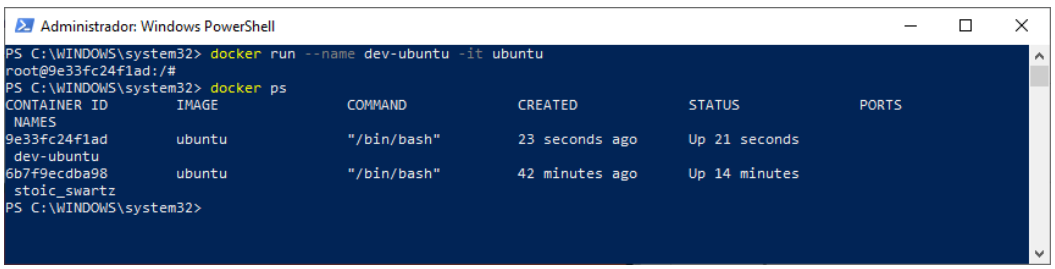
14. Salir del contenedor sin terminar la ejecución. Inicia el contenedor con la opción -it (interactive y un seudo terminal) y ejecuta la combinación de teclas [Ctrl] + [p] + [q]



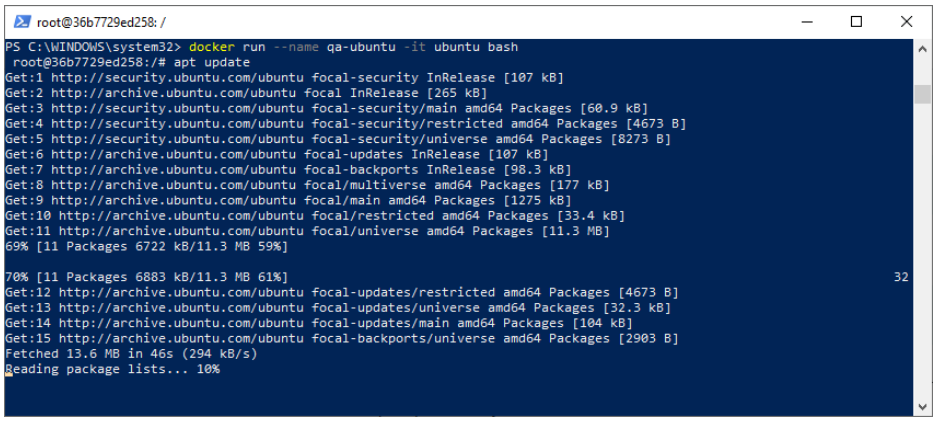
15. Contenedores interactivos. Si el contenedor fue creado con la opción **-it**, podemos interactuar con el contenedor, podemos iniciarlo con la opción **start**, podemos recuperar el "focus" con la opción **attach**.



16. Creando contenedores con nombre. Podemos asignar un nombre al contenedor con el parámetro --name



17. Personalización de Contenedores. Crearemos un contenedor basado en ubuntu y realizaremos la instalación de algunas aplicaciones clásicas.



En el contenedor ejecuta los comandos:

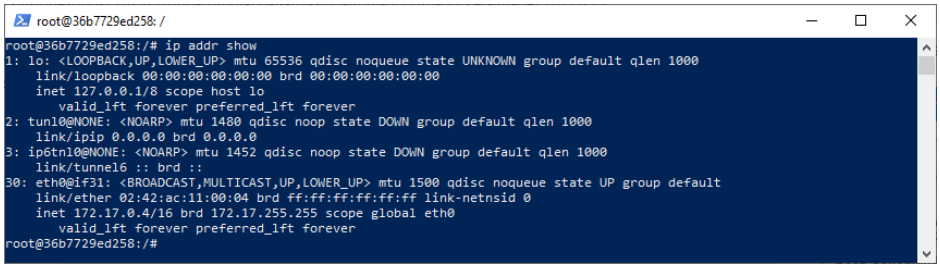
$ apt install nano

Luego:

$apt install git

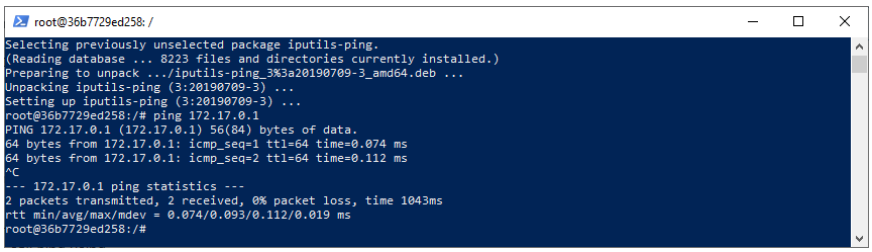
$apt install iproute2

Ahora podemos ejecutar el comando: ip addr show.

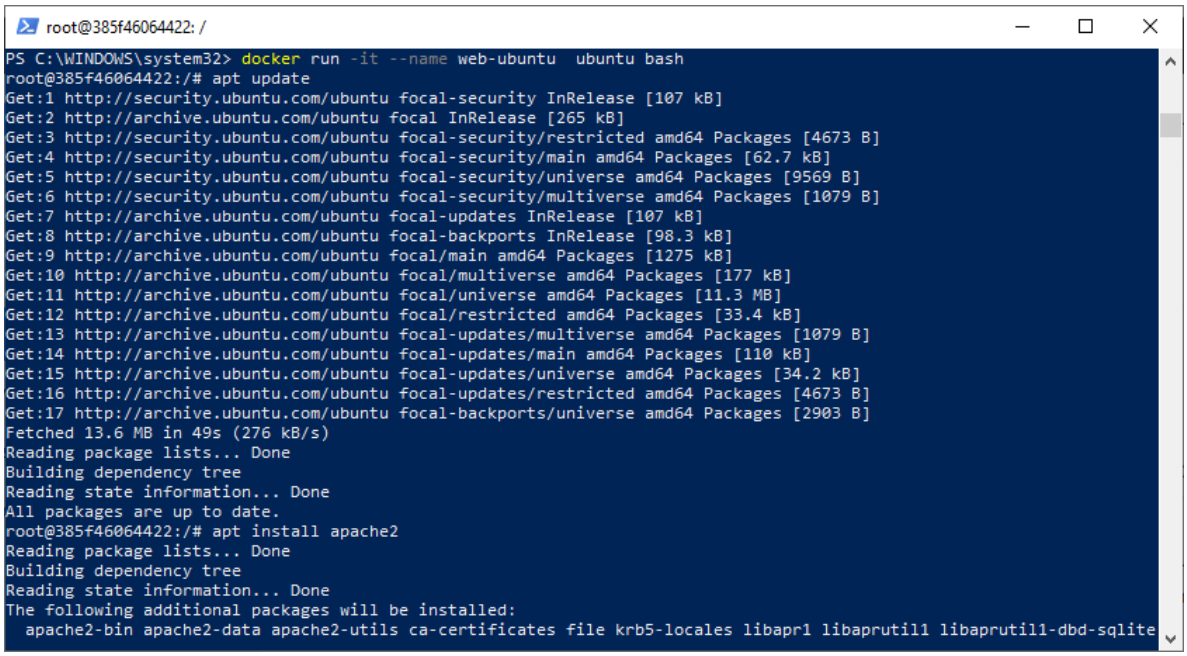


Instala las herramientas para ejecutar comando de networking.

apt install iputils-ping



18. Creando imágenes a partir de contenedores. Crearemos un container iterativo llamado web-ubuntu basado en la distribución de ubuntu, luego instalaremos apache2.



Dentro del contenedor, ejecuta el comando

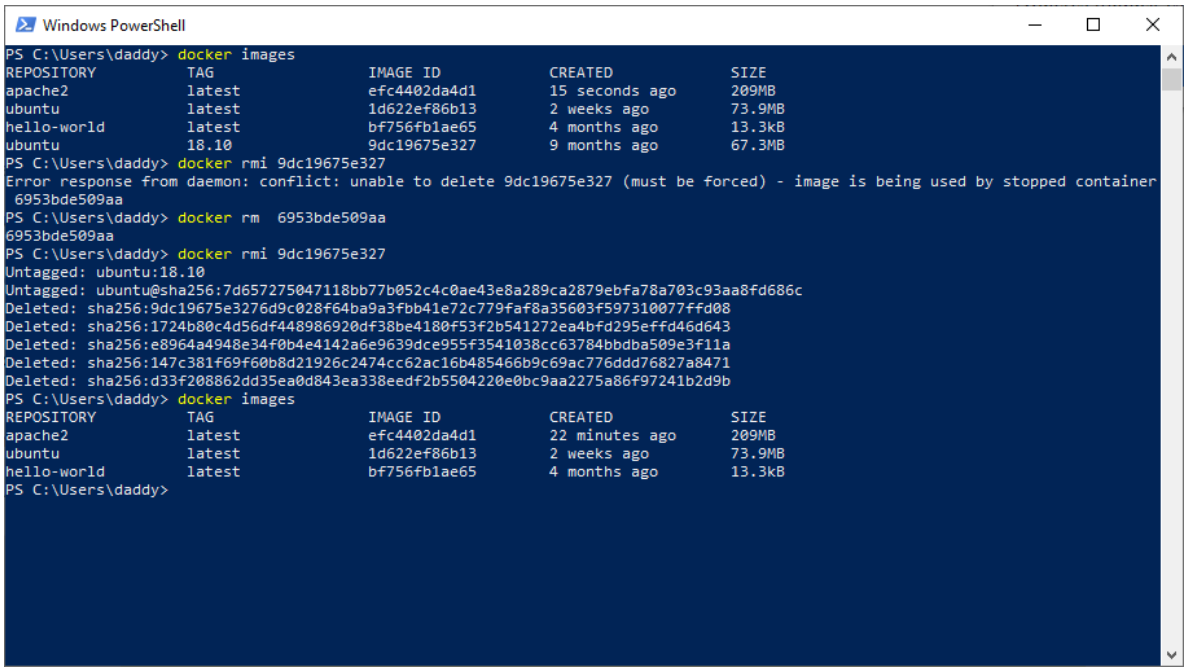
apt update

Luego:

apt install apache2

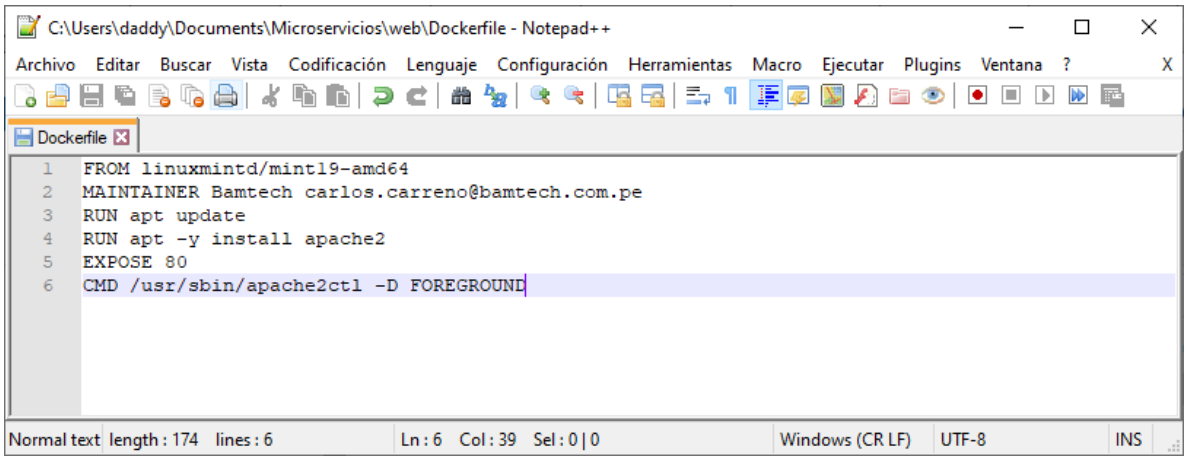
En otro terminal ubica el id del container **web-ubuntu** y crea la nueva imagen con nombre **apache2**.

19. Eliminando imágenes. Para eliminar imágenes utiliza la acción **rmi** (remove image).



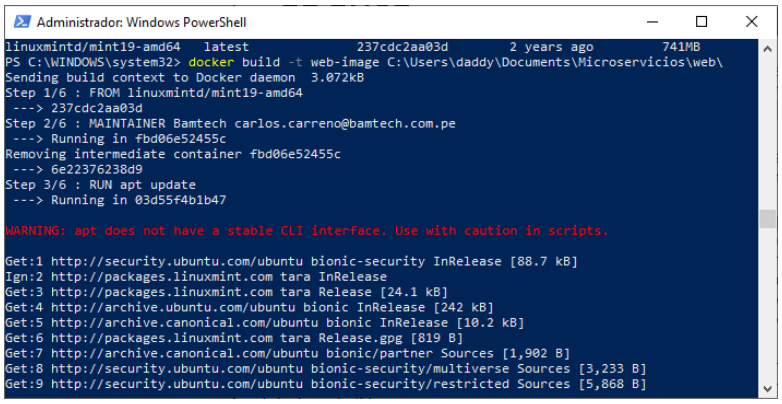
Nota: Observa que, si la imagen esta siendo utilizada por algún contenedor no dejara eliminarla de manera normal, primero tienes que eliminar los contenedores.

20. Creación de una imagen desde un Dockerfile. Crea el archivo de configuración de la imagen, llama al archivo **Dockerfile**.

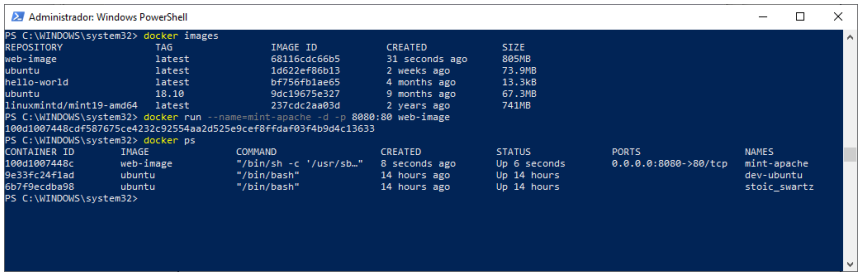


Luego ejecuta el comando:

docker build -t web-image C:\Users\daddy\Documents\Microservicios\web\



21. Creación del contenedor y exposición de puertos locales.



22. Acceso al software de middleware o servicio (servicio, servidor web, servidor de aplicaciones). Para acceder al apache web server del contenedor **mint-apache**, abre un browser y acceder a la url: http://localhost:8080