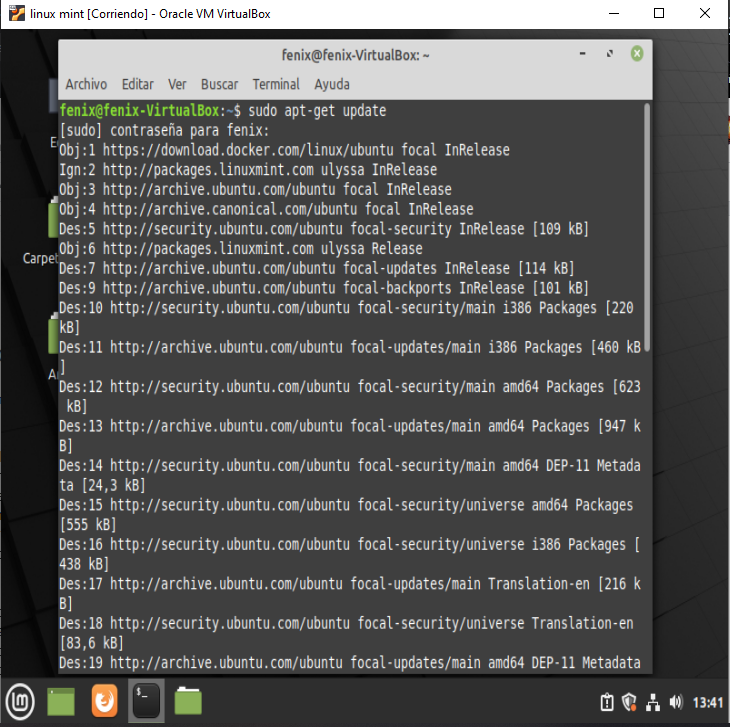
# **Taller 3 - Implementación de una Aplicación Multicontenedor en Docker-Compos**

Juan david forero acosta - 2220181022; Jorge Nikolas Pava Delgadillo - 2220201042; Nicolas Orlando Perez Cruz - 2220181014; Maria Paula Useche Carvajal - 2220211025

1. **Instalación del docker compose:**Pasos para la comprobación que se tiene instalado el docker:

Inicialmente, se procedió a actualizar el sistema para que sea más seguro y confiable para la instalación de Docker. Como se evidencia en la Imagen 1, se emplea el comando *sudo apt update* y este se actualiza la lista de paquetes disponibles y sus versiones, pero no se instalan o actualizan ningún paquete. Esta lista es seleccionada de los servidores con repositorios que tenemos definidos en el sources.list.

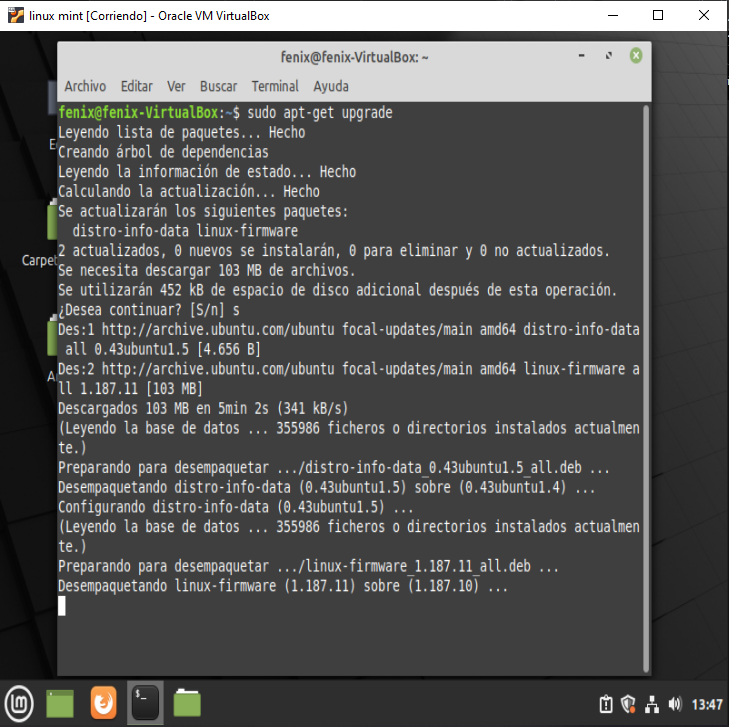
**sudo apt update**



*Imagen 1. Actualización del sistema, fuente propia (2021)*

Con el comando anteriormente mencionado haya descargado la lista de software disponible y la versión en la que se encuentra, se puede actualizar dichos paquetes usando el comando *apt-get upgrade* (ver Imagen 2). Con este comando se instalarán las nuevas versiones respetando la configuración del software cuando sea posible.

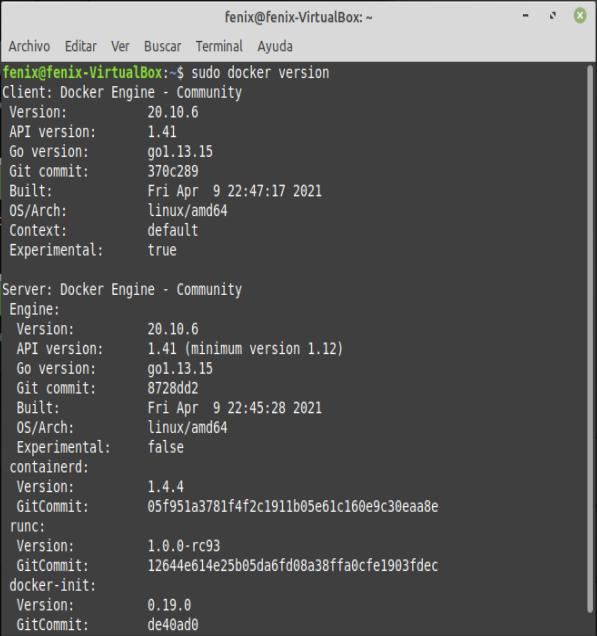
**sudo apt-get upgrade**



*Imagen 2. Descarga de versiones disponibles del sistema, fuente propia (2021)*

Adicional a esto, se confirmó que sí se tenga instalado el docker y luego, como se muestra en la Imagen 3, se procede a ejecutar el siguiente comando *sudo docker version.*

**sudo docker version**



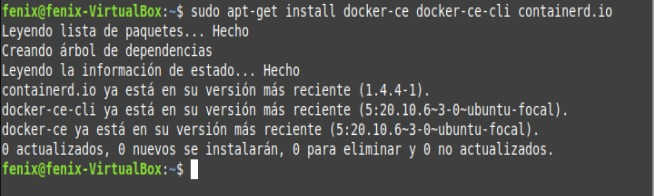
*Imagen 3. Comprobación de la versión del docker, fuente propia (2021)*

En caso de no contar con un docker, el siguiente [link](https://drive.google.com/file/d/1vh42LMxdUMeGJbxetqeRXpOwMcLSpfj0/view?usp=sharing) lleva a un instalador para completar esta tarea de forma rápida y sencilla.

Después de confirmar que se cuenta con docker, se procede a la instalación de docker compose, pero para eso se debe tener en cuenta los siguientes requisitos previos:

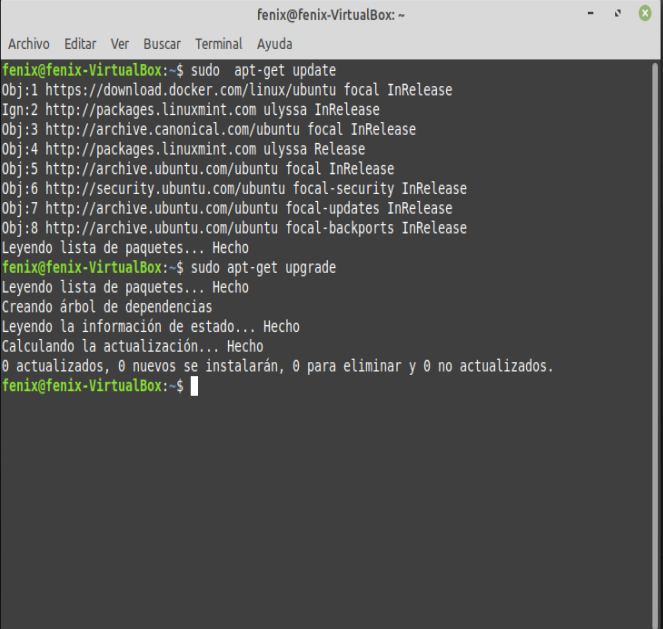
* Docker Compose se basa en Docker Engine para cualquier trabajo significativo, así que se debe asegurar tener Docker Engine instalado de forma local o remota, según su configuración.
* En sistemas de escritorio como Docker Desktop para Mac y Windows, Docker Compose se incluye como parte de esas instalaciones de escritorio.
* En sistemas Linux, primero instalar Docker Engine para su sistema operativo, para esto se puede ingresar al siguiente [link](https://docs.docker.com/engine/install/ubuntu/).

Como resultado debe quedar lo siguiente:



*Imagen 4. Instalación del docker Engine , fuente propia (2021).*

Algo importante que se debe tener en cuenta es que los paquetes deben estar actualizados, esto con el fin de prevenir errores durante la instalación.



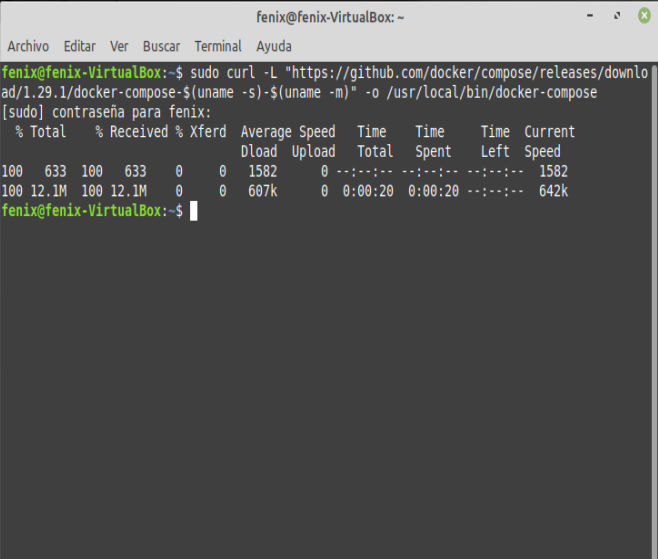
*Imagen 5. Comprobación de todas las actualizaciones, fuente propia (2021).*

**Nota:** Para ejecutar Compose como usuario no root, se puede consultar Administrar Docker como usuario no root .

Después de comprobar que se cumpla todos los requisitos pasamos a instalar ahora si el docker compose, como se muestra en la Imagen 6, se procede a ejecutar el siguiente comando para descargar la versión estable actual de Docker Compose:

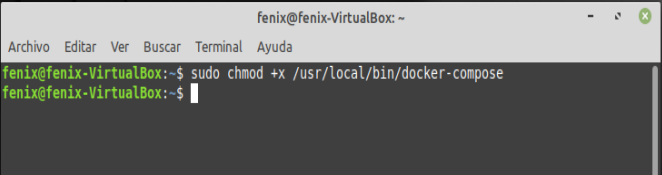
sudo curl -L “https://github.com/docker/compose/releases/download/1.29.1/docker-co

mpose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose



*Imagen 6. Instalación de la versión más estable del docker compose , fuente propia (2021).*

Finalmente para terminar con la instalación se procede a darle los correspondientes permisos ejecutables al binario con el comando *sudo chmod +x /usr/local/bin/docker-compose.*

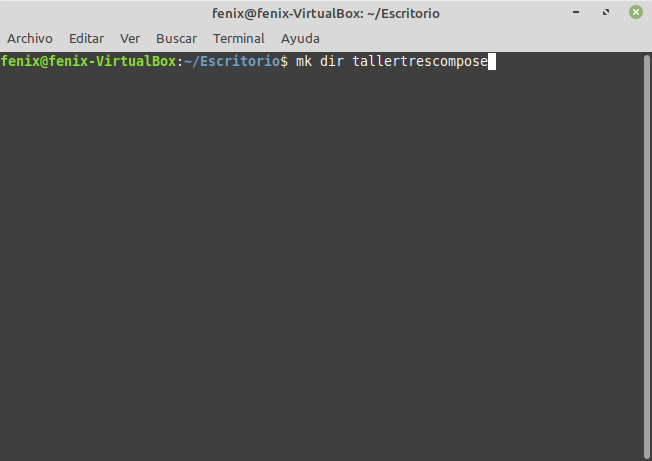
**sudo chmod +x /usr/local/bin/docker-compose**

*Imagen 7. Instalación de la versión más estable del docker compose , fuente propia (2021)*.

1. **Construcción del proyecto en docker compose**

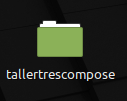
El primer paso es proceder a la correspondiente configuración, donde se define las dependencias de la aplicación, por ende se crea un directorio para el proyecto de la siguiente manera:

**mkdir tallertrescompose**



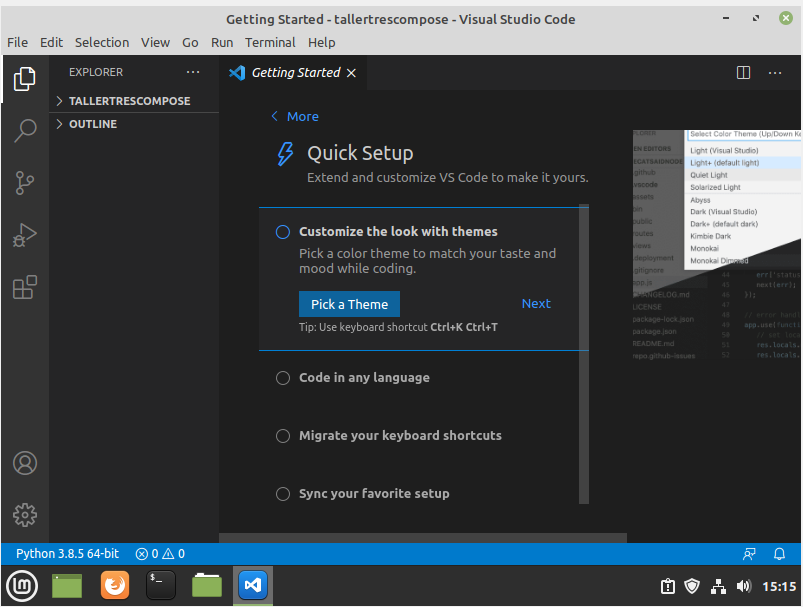
*Imagen 8. Creación de directorio del proyecto , fuente propia (2021).*

Para continuar con el procedimiento de la creación del proyecto en Python se recomienda trabajar con visual code, con sus diversos plugins. Una vez obtenidos todos los recursos se procede a crear una carpeta en el escritorio la cual se llamará *tallertrescompose* como se observa en la Imagen 9.



*Imagen 9. Creación de la carpeta del proyecto , fuente propia (2021).*

Después de haber creado la carpeta se abrirá en visual studio code, dando como resultado lo siguiente:



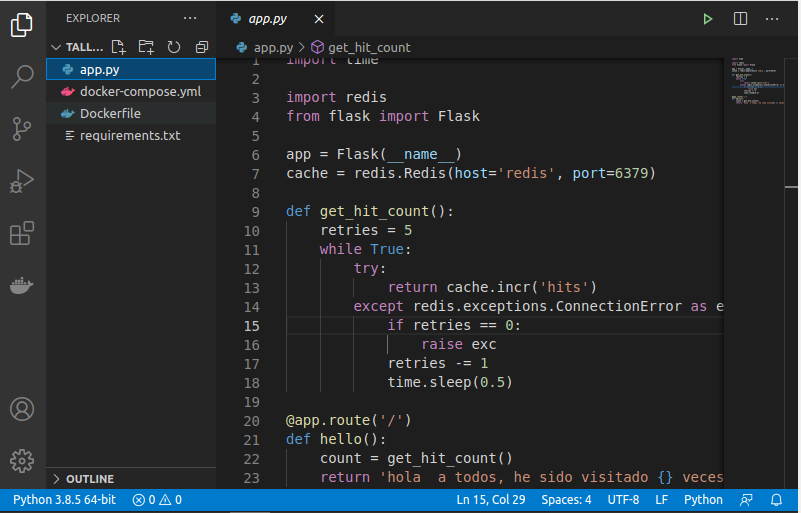
*Imagen 10.Insertar la carpeta del proyecto en visual code, fuente propia (2021).*

Ahora en la carpeta del proyecto que se ve en la imagen 10, se procede a crear un archivo llamado app.py, cabe resalta que py es para definir que es un proyecto Python, como se ve en la Imagen 11:



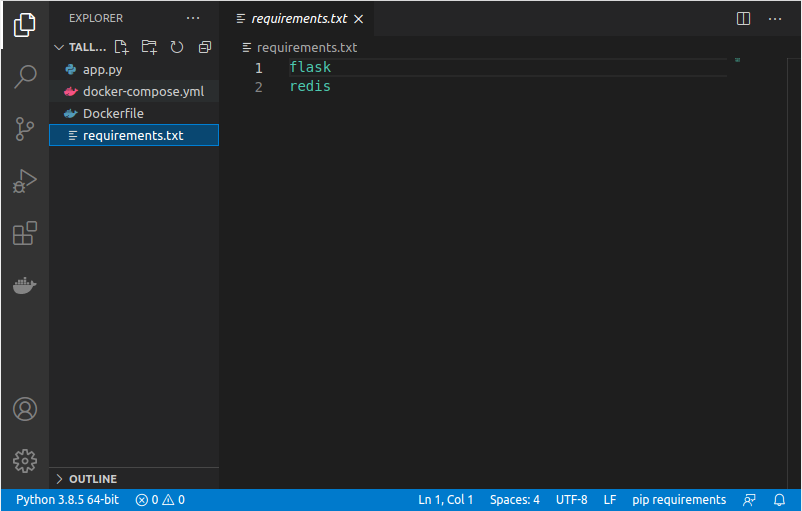
*Imagen 11. Creación del archivo donde estará el código fuente del proyecto, fuente propia (2021).*

Ahora bien, dentro del App.py introduccimos el correspondiente código de la aplicación, como se observa en la siguiente imagen:



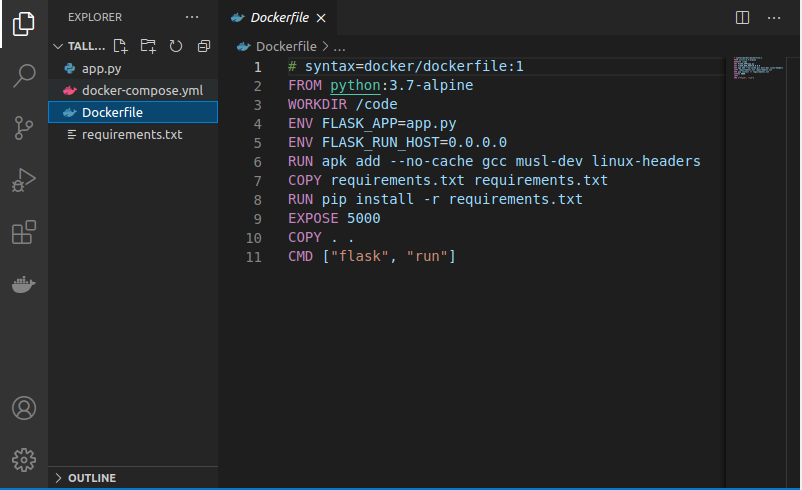
*Imagen 12. código fuente del proyecto, fuente propia (2021)*

Para este trabajo el redises es el nombre de host del contenedor redis en la red de la aplicación. Se utilizó el puerto predeterminado para Redis, es decir el puerto [ 6379 ]. Ahora, se crea un archivo con el nombre requirements.txt para insertar los datos que se ven en la Imagen 13, en este caso se insertó los datos, flask y redis.



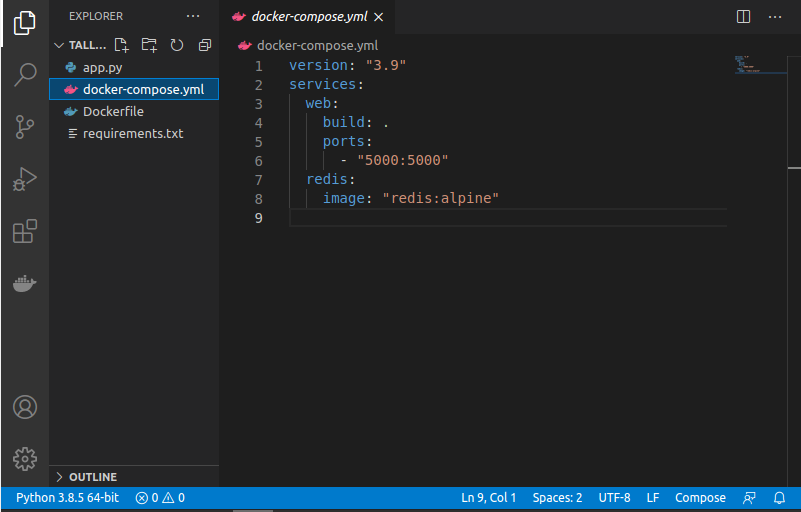
*Imagen 13. Creación de la carpeta requirements.txt , fuente propia (2021)*

Después se procedió a crear el correspondiente Dockerfile, dentro de este se crea una imagen de Docker. Dicha imagen contiene todas las dependencias que requiere la aplicación Python, incluido el propio Python, con el cual se trabajó en el proyecto. Como se evidencia en la Imagen 14, se creó una imagen comenzando con la imagen de Python 3.7, se establece el directorio de trabajo en /code al igual que las variables de entorno que utiliza el flash comando, se instaló gcc y otras dependencias, se estableció el requirements.txt y se instaló las dependencias de Python, se agregaron los metadatos a la imagen para describir que el contenedor está escuchando en el puerto 5000, se copio el directorio actual del proyecto en el directorio de trabajo de la imagen y finalmente se estableció el comando predeterminado para el contenedor en flask run.



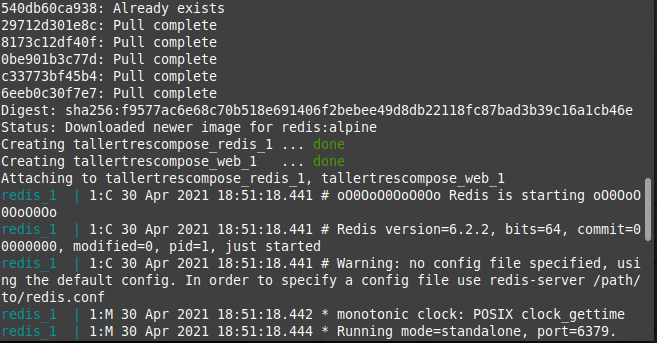
*Imagen 14. Creación del Dockerfile, fuente propia (2021)*

Por otro lado, se procedió a definir los correspondientes servicios que tendrá el archivo, para esto, se creó docker-compose.yml en el directorio del proyecto base y se definió el servicio web y redis. Específicamente para el servicio web se utiliza una imagen creada a partir del Dockerfile Directorio actual. Esta, se une al recipiente y la máquina host al puerto expuesto (5000). Este servicio de ejemplo utiliza el puerto predeterminado para el servidor web Frasco (5000), se puede afirmar que otro local Host común es el 8080. Por otro lado para el redis servicio utiliza una imagen pública de Redis extraída del registro de Docker Hub.

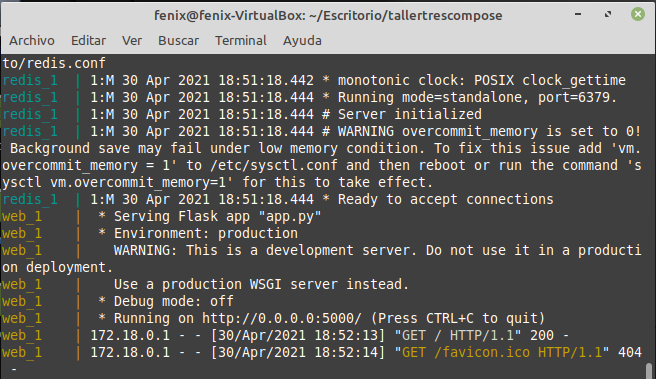


*Imagen 15. Asignación de servicios del proyecto, fuente propia (2021)*

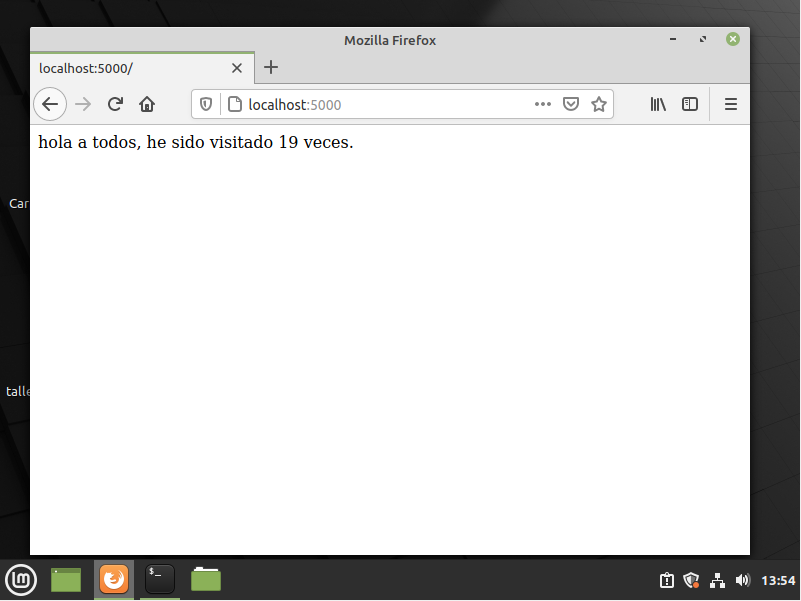
Seguido a esto, en el directorio actual, se inició la aplicación (Imagen 15 y 16). Luego, Compose crea una imagen para su código extrayendo la imagen de Redis, se crea una imagen para el código y este inicia los servicios que se definieron anteriormente.



*Imagen 16. Inicialización de la aplicación, fuente propia (2021)*



*Imagen 17. Inicialización de la aplicación, fuente propia (2021)*

Finalmente, se ingresa al link <http://localhost:5000/> en el navegador para ver la ejecución de la aplicación. En este caso, como se empleó Docker en Linux, la aplicación web está conectada al puerto 5000.Como se evidencia en la Imagen 18, al abrir la aplicación en el navegador web se encuentra el mensaje, en este caso “Hola a todos, he sido visitado 19 veces”*.* El valor que toma el número de visitas es debido a que se actualizó la página 19 veces. 

*Imagen 18. Visualización de ejecución de la aplicación, fuente propia (2021)*