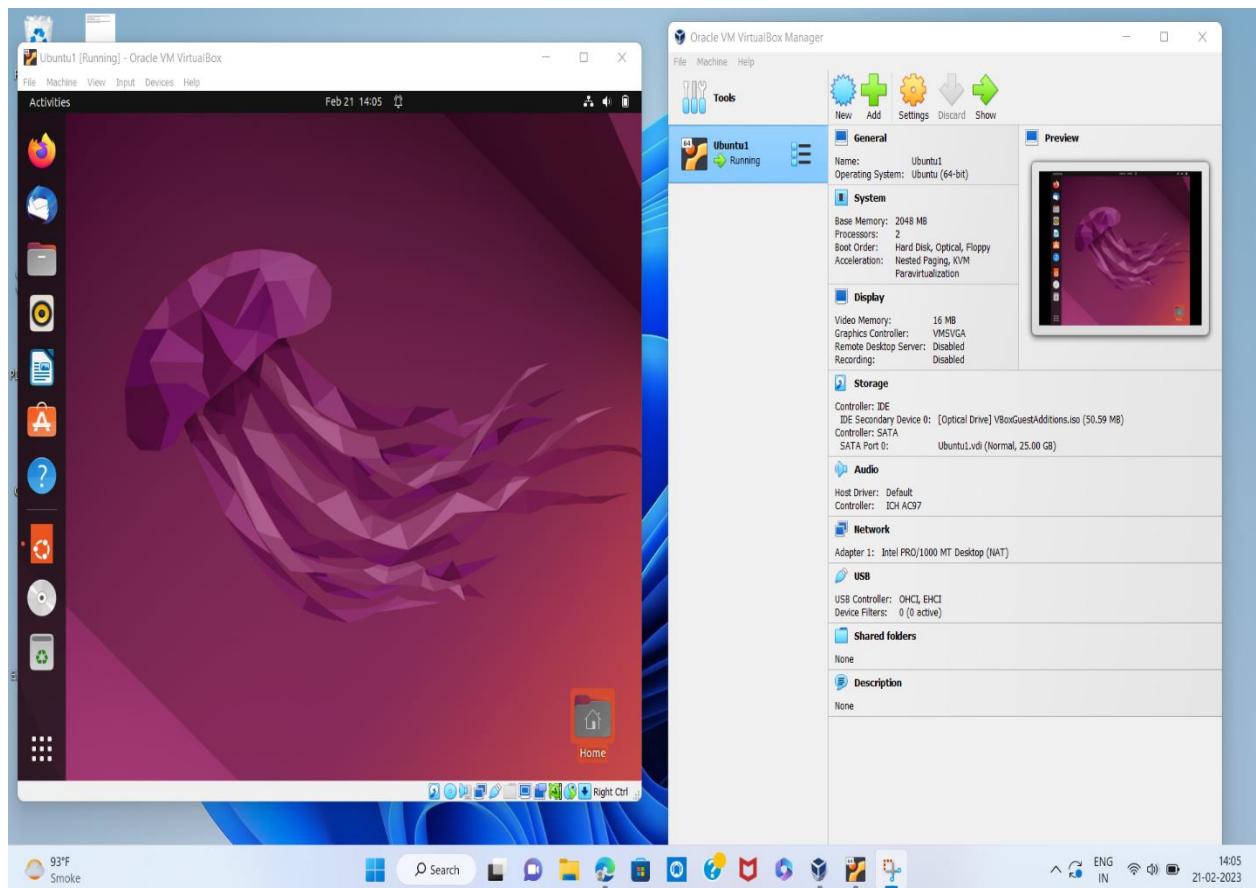


WEEK 11 ASSIGNMENT

Steps

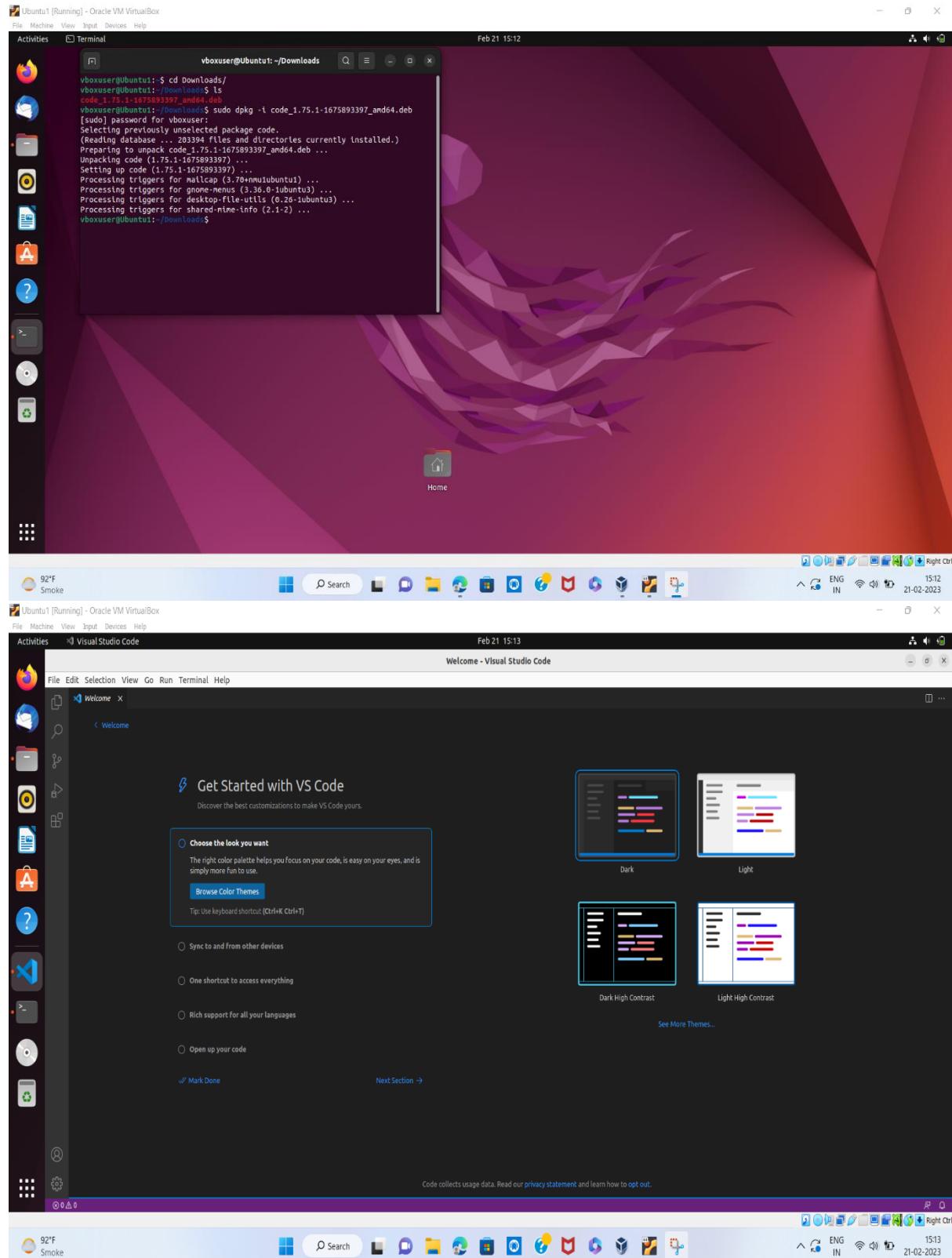
1. Host a Ubuntu Virtual Machine using Oracle VM Virtual Box.



2. Set up Visual Studio code on Ubuntu VM.

Download a file “code-1.75.1-1675893397_amd64.deb”

Command : sudo dpkg -i code-1.75.1-1675893397_amd64.deb



3. Set up Python.

Command 1: sudo apt update

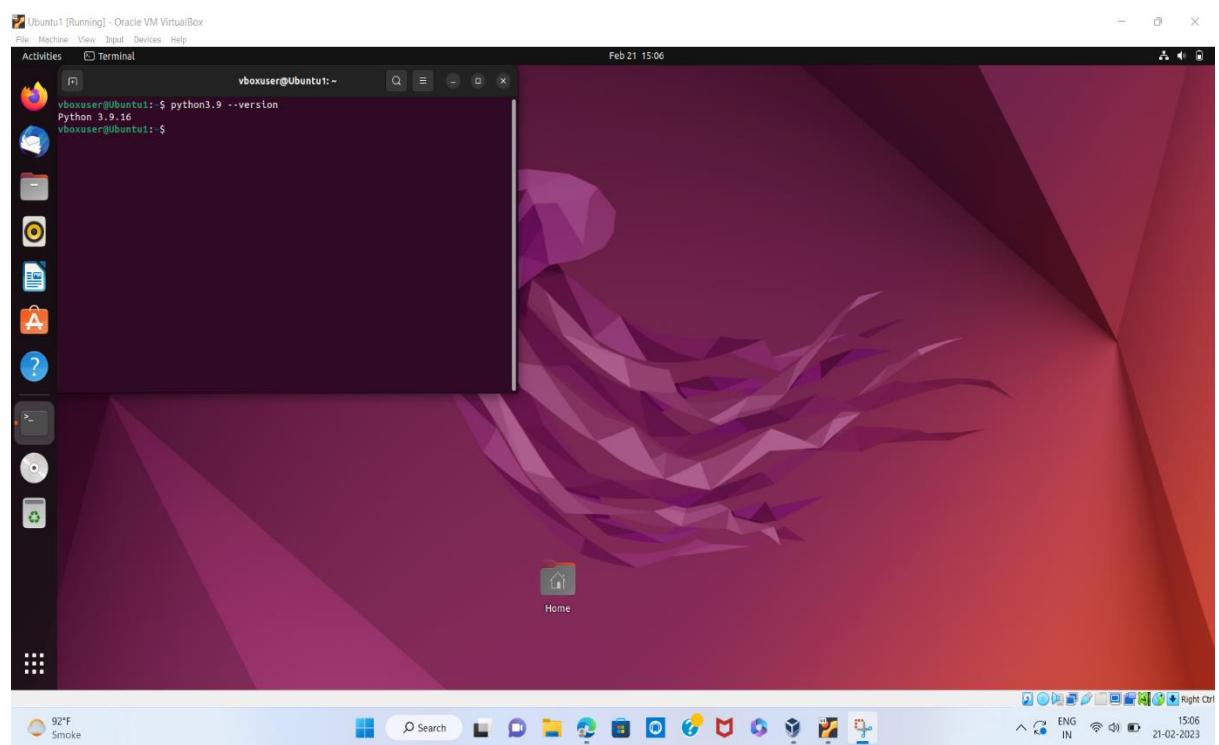
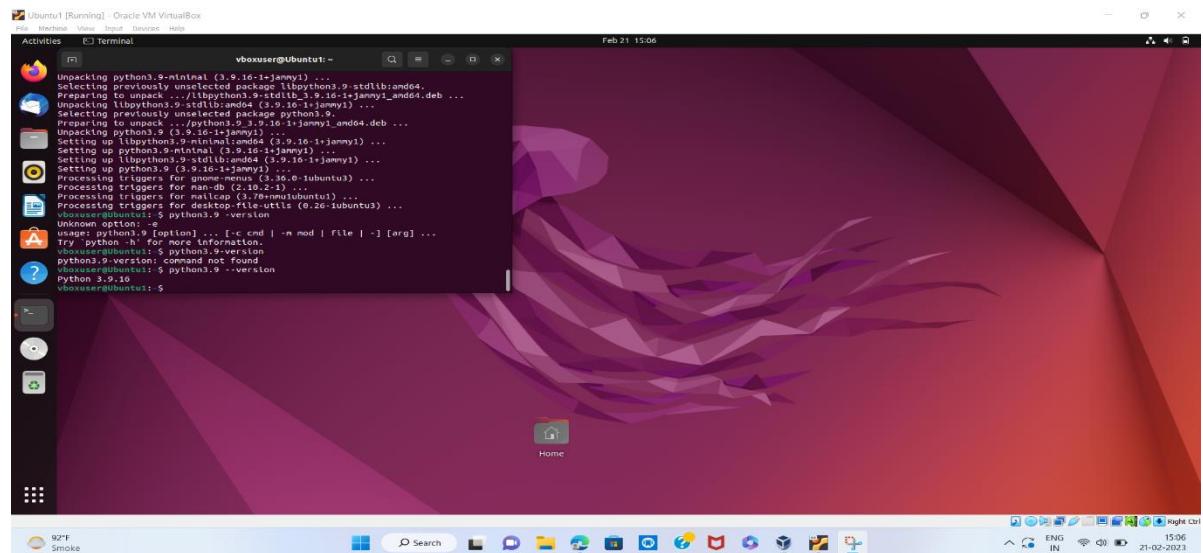
Command 2: sudo apt install software-properties-common

Command 3: sudo add-apt-repository ppa:deadsnakes/ppa

Command 4: sudo apt install python3.9

- Verification of the installation was successful.

Command: python3.9 --version



4.Clone this Github repository

<https://github.com/Vikas098766/Microservices.git>

Install git and then clone the specified repository

Command : git clone https://github.com/Vikas098766/Microservices.git

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window displays the following command and its execution:

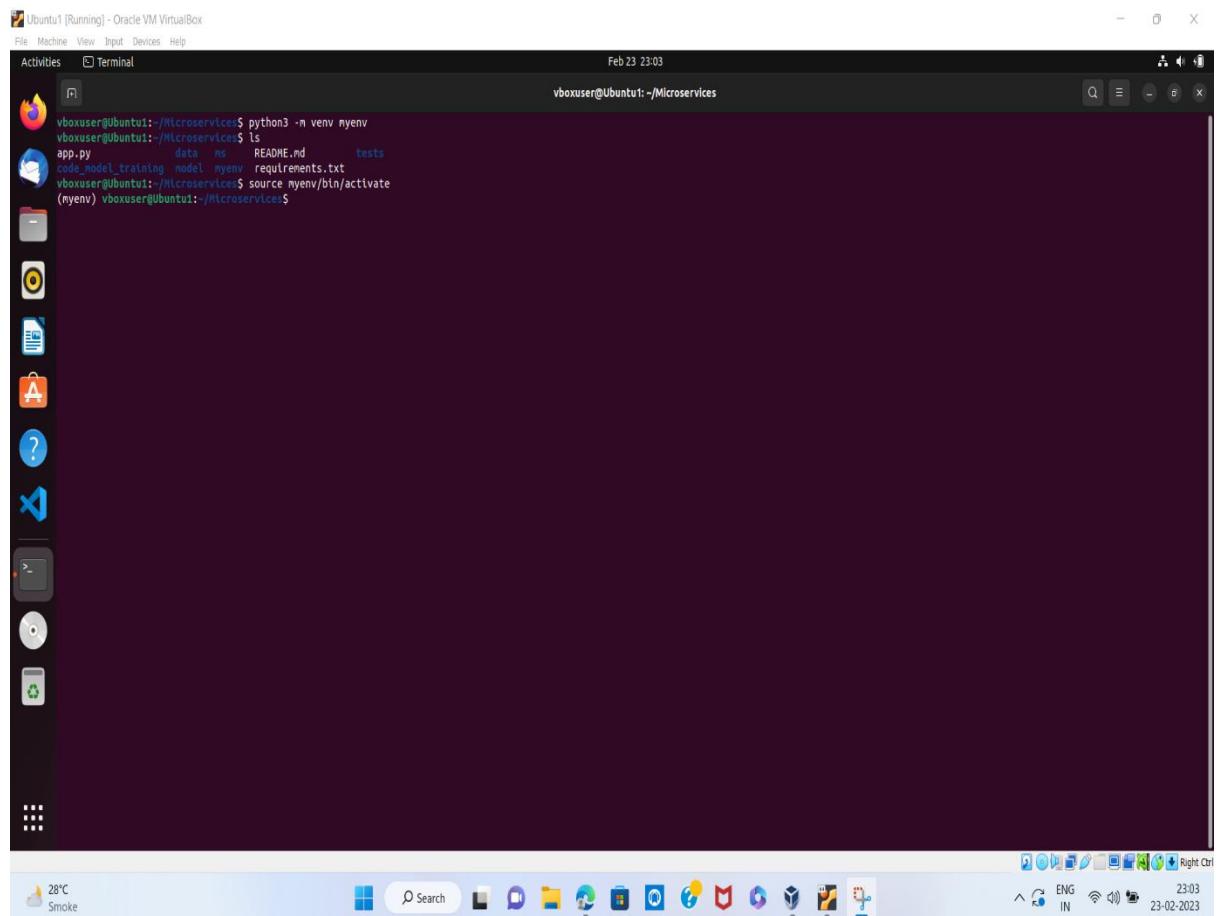
```
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libflashrom1 liblfrd1-2 liblflv13
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-email
  git-gui gtk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git-man liberror-perl
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
Need to get 4,121 kB of archives.
After this operation, 20,9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.8 [953 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.8 [3,141 kB]
Fetched 4,121 kB in 11s (388 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 204821 files and directories currently
installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1)
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.34.1-1ubuntu1.8_all.deb
...
Unpacking git-man (1:2.34.1-1ubuntu1.8) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.34.1-1ubuntu1.8_amd64.deb .
...
Unpacking git (1:2.34.1-1ubuntu1.8) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git-man (1:2.34.1-1ubuntu1.8) ...
Setting up git (1:2.34.1-1ubuntu1.8) ...
Processing triggers for man-db (2.10.2-1) ...
vboxuser@ubuntu:~$ git --version
vboxuser@ubuntu:~$ git clone https://github.com/Vikas098766/Microservices
Cloning into 'Microservices'...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (2/2), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 95 (delta 0), reused 0 (delta 0), pack-reused 93
Receiving objects: 100% (95/95), 96.43 KiB | 3.33 MiB/s, done.
Resolving deltas: 100% (27/27), done.
vboxuser@ubuntu:~$ ls
Desktop Documents Downloads Microservices Music Pictures Public snap Templates Videos
vboxuser@ubuntu:~$
```

The desktop environment includes a dock with various icons, a system tray at the bottom right, and a file manager window showing a directory structure.

5.Create a Virtual Environment.

Command : python3 -m venv myenv

Command: source venv/bin/activate



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "Terminal" and the command prompt is "vboxuser@Ubuntu: ~/Microservices". The user has run the following commands:

```
vboxuser@Ubuntu:~/Microservices$ python3 -m venv myenv
vboxuser@Ubuntu:~/Microservices$ ls
app.py      data  ms  README.md  tests
code_model_training  model  myenv  requirements.txt
vboxuser@Ubuntu:~/Microservices$ source myenv/bin/activate
(myenv) vboxuser@Ubuntu:~/Microservices$
```

The desktop interface includes a dock with various icons (File Explorer, Terminal, etc.) and a system tray at the bottom showing weather information (28°C, Smoke), a search bar, and system status indicators.

6. Install the dependencies from requirements.txt file.

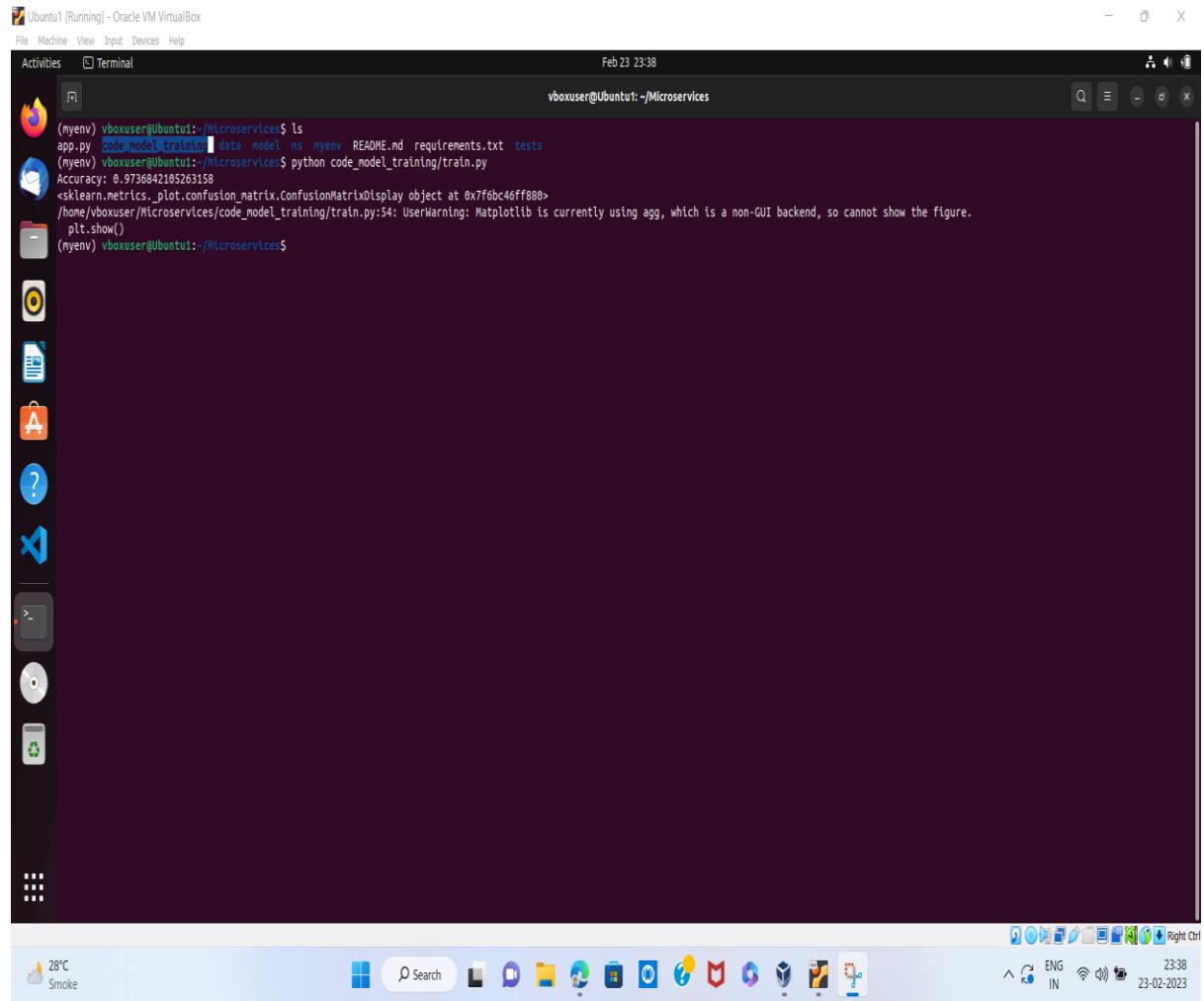
Command : pip install -r requirements.txt

```
vboxuser@Ubuntu1:~/Microservices$ python3 -m venv myenv
vboxuser@Ubuntu1:~/Microservices$ ls
app.py           data    README.md  tests
model_training.ipynb  requirements.txt
vboxuser@Ubuntu1:~/Microservices$ source myenv/bin/activate
(myenv) vboxuser@Ubuntu1:~/Microservices$ pip install -r requirements.txt
Collecting click==0.3
  Downloading click-0.3.0-py3-none-any.whl (97 kB)
Collecting cyclere==0.11.0
  Downloading cyclere-0.11.0-py3-none-any.whl (0.4 kB)
Collecting Flask==2.0.2
  Downloading Flask-2.0.2-py3-none-any.whl (95 kB)
Collecting fonttools==4.28.5
  Downloading fonttools-4.28.5-py3-none-any.whl (898 kB)
Collecting gunicorn==20.1.0
  Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
Collecting itsdangerous==2.0.1
  Downloading itsdangerous-2.0.1-py3-none-any.whl (18 kB)
Collecting Jinja2==3.0.3
  Downloading Jinja2-3.0.3-py3-none-any.whl (133 kB)
Collecting joblib==1.1.0
  Downloading joblib-1.1.0-py2-py3-none-any.whl (386 kB)
Collecting kiwisolver==1.3.2
  Downloading kiwisolver-1.3.2-cp310-cp310-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.6 MB)
Collecting MarkupSafe==2.0.1
  Downloading MarkupSafe-2.0.1-cp310-cp310-manylinux_2_12_x86_64.manylinuxx_86_64.manylinux_2_12_x86_64.manylinux2010_x86_64.whl (30 kB)
Collecting matplotlib==3.5.1
  Downloading matplotlib-3.5.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.9 MB)
Collecting numpy==1.22.0
  Downloading numpy-1.22.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.8 MB)
Collecting packaging==21.3
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
Collecting pandas==1.3.5
  Downloading pandas-1.3.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.5 MB)
```

```
vboxuser@Ubuntu1:~/Microservices$ python3 -m venv myenv
vboxuser@Ubuntu1:~/Microservices$ ls
app.py           data    README.md  tests
model_training.ipynb  requirements.txt
vboxuser@Ubuntu1:~/Microservices$ source myenv/bin/activate
(myenv) vboxuser@Ubuntu1:~/Microservices$ pip install -r requirements.txt
Collecting matplotlib==3.5.1
  Downloading matplotlib-3.5.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.9 MB)
Collecting numpy==1.22.0
  Downloading numpy-1.22.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.8 MB)
Collecting packaging==21.3
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
Collecting pandas==1.3.5
  Downloading pandas-1.3.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.5 MB)
Collecting Pillow==9.0.0
  Downloading Pillow-9.0.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.3 MB)
Collecting pyrsparser==3.0.6
  Downloading pyrsparser-3.0.6-py3-none-any.whl (97 kB)
Collecting python-dateutil==2.8.2
  Downloading python_dateutil-2.8.2-py3-none-any.whl (247 kB)
Collecting pytz==2021.3
  Downloading pytz-2021.3-py2-py3-none-any.whl (503 kB)
Collecting scikit-learn==1.0.2
  Downloading scikit_learn-1.0.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (26.5 MB)
Collecting scipy==1.7.3
  Downloading scipy-1.7.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (39.9 MB)
Collecting six==1.16.0
  Downloading six-1.16.0-py2-py3-none-any.whl (11 kB)
Collecting sklearn==0.0
  Downloading sklearn-0.0.tar.gz (1.1 kB)
  Preparing metadata (setup.py) ... done
Collecting threadpoolctl==3.0.0
  Downloading threadpoolctl-3.0.0-py3-none-any.whl (14 kB)
Collecting Werkzeug==2.0.2
  Downloading Werkzeug-2.0.2-py3-none-any.whl (288 kB)
Requirement already satisfied: setuptools==3.0 in ./myenv/lib/python3.10/site-packages (from gunicorn==20.1.0->-r requirements.txt (line 5)) (59.60)
Using legacy 'setup.py install' for sklearn, since package 'wheel' is not installed.
Installing collected packages: pytz, Werkzeug, threadpoolctl, six, pyrsparser, Pillow, numpy, MarkupSafe, kiwisolver, joblib, itsdangerous, gunicorn, fonttools, cyclere, click, scipy, python-dateutil, packaging, Jinja2, scikit-learn, pandas, matplotlib, Flask, sklearn
  Running setup.py install for sklearn ... done
Successfully installed Flask-2.0.2 Jinja2-3.0.3 MarkupSafe-2.0.1 Pillow-9.0.0 Werkzeug-2.0.2 click-0.11.0 fonttools-4.28.5 gunicorn-20.1.0 itsdangerous-2.0.1 joblib-1.1.0 kiwisolver-1.3.2 matplotlib-3.5.1 numpy-1.22.0 packaging-21.3 pandas-1.3.5 pyrsparser-3.0.6 python-dateutil-2.8.2 pytz-2021.3 scikit-learn-1.0.2 scipy-1.7.3 six-1.16.0 sklearn-0.0 threadpoolctl-3.0.0
(myenv) vboxuser@Ubuntu1:~/Microservices$
```

7.Train and save the model.

Command : python code_model_training/train.py



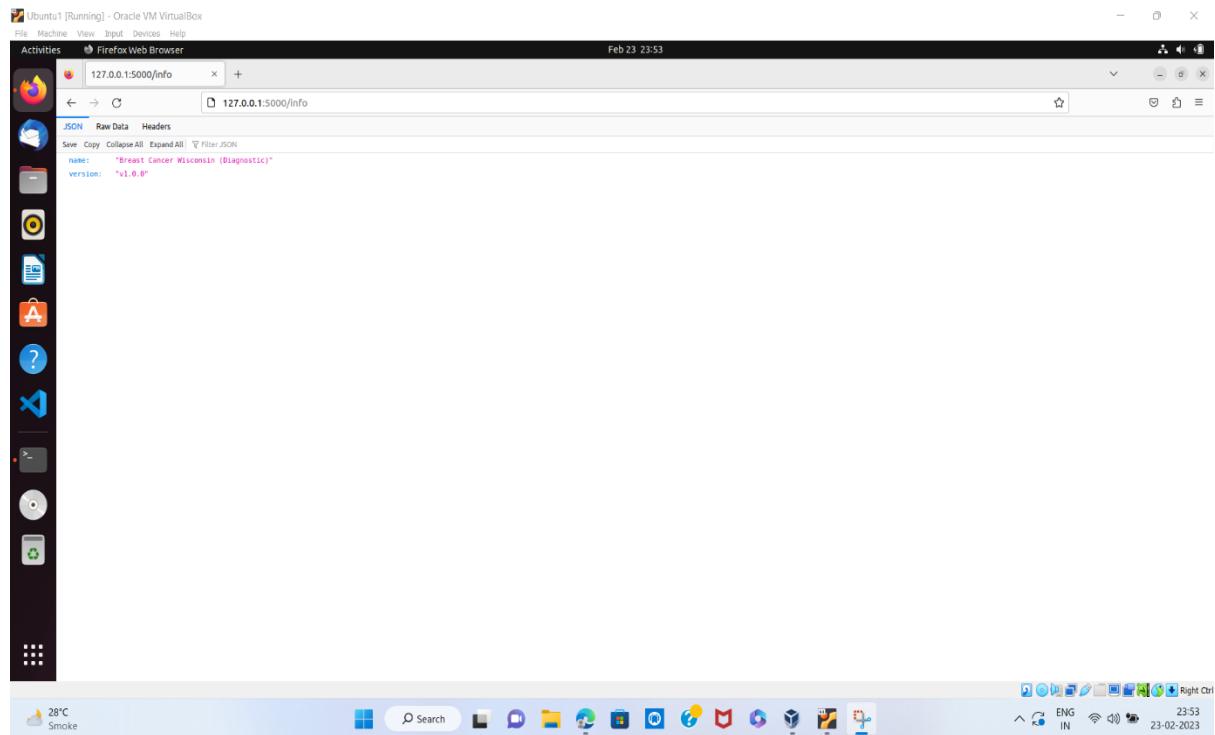
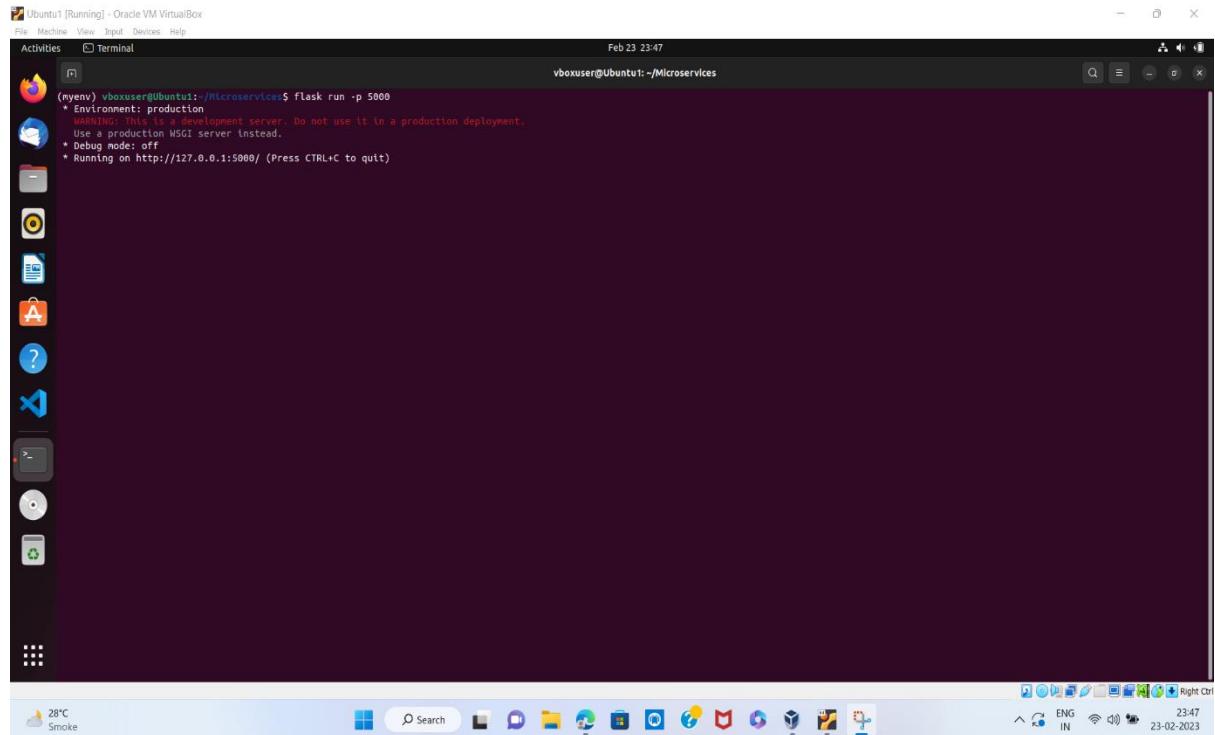
The screenshot shows a Linux desktop environment with a dark theme. A terminal window is open in the center, titled 'vboxuser@Ubuntu1: ~/Microservices'. The terminal displays the following command and its output:

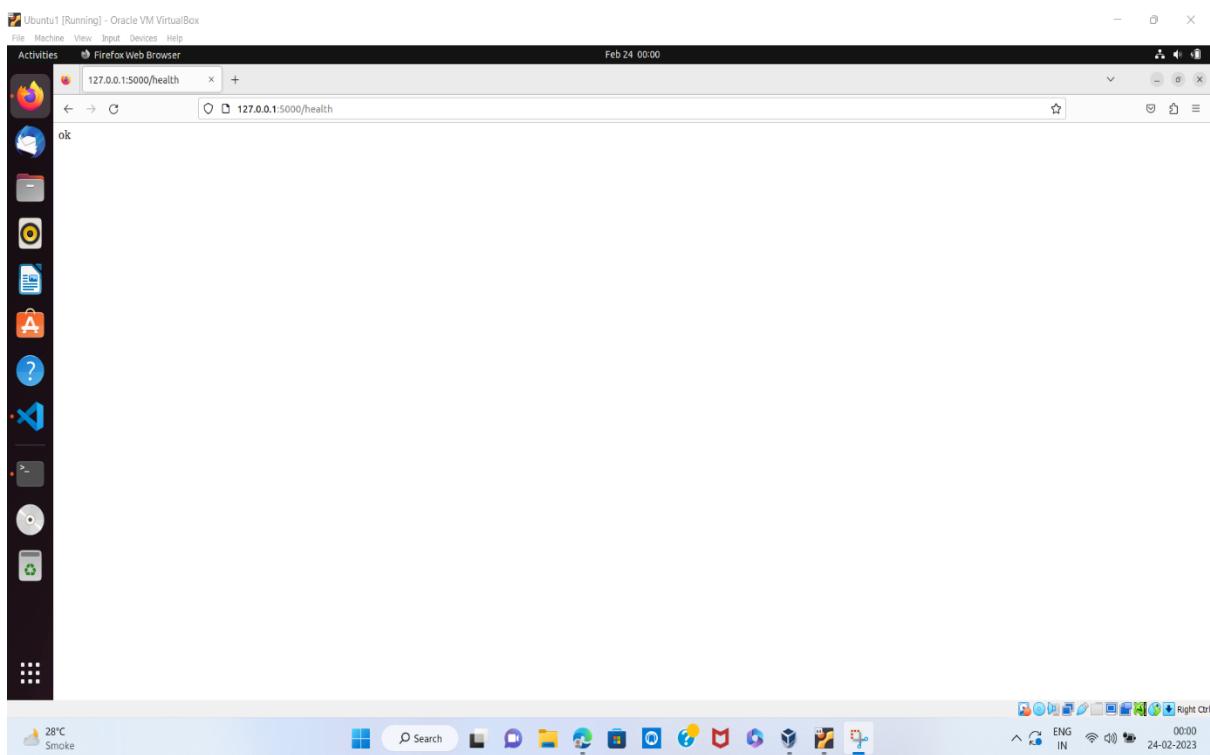
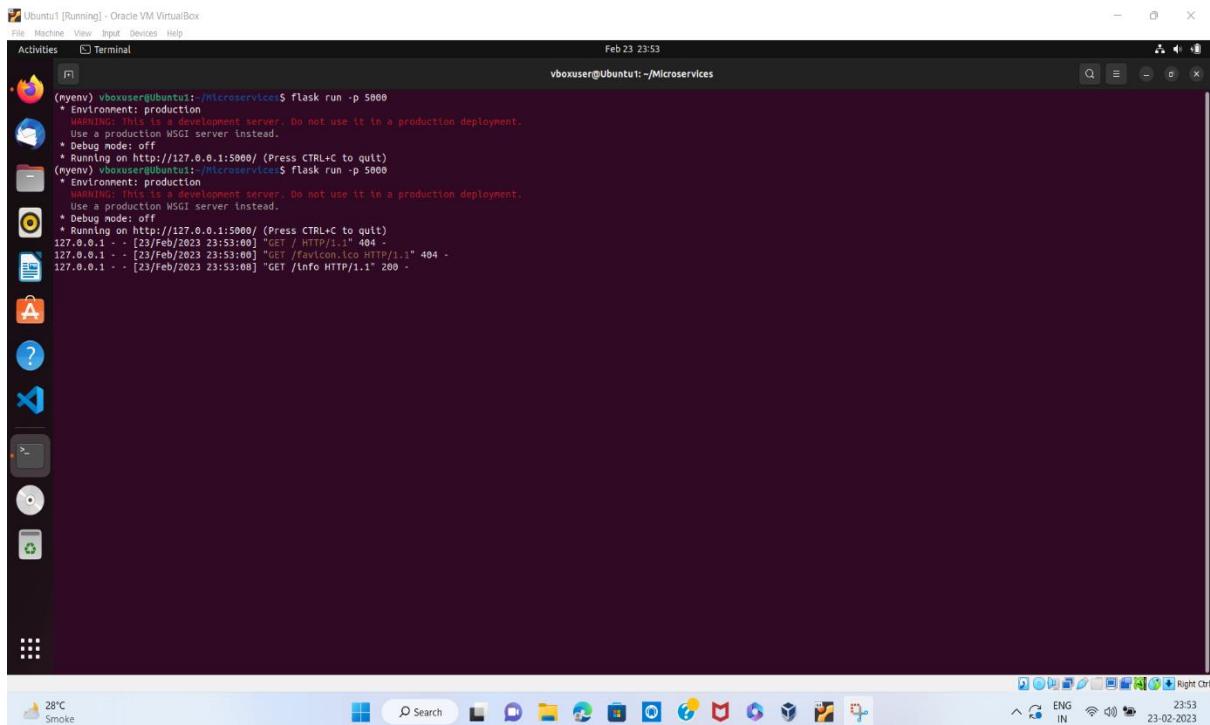
```
(myenv) vboxuser@Ubuntu1:~/Microservices$ ls
app.py code_model_training data model ms myenv README.md requirements.txt tests
(myenv) vboxuser@Ubuntu1:~/Microservices$ python code_model_training/train.py
Accuracy: 0.9736842105263158
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay object at 0x7f6bc46ff880>
/home/vboxuser/Microservices/code_model_training/train.py:54: UserWarning: Matplotlib is currently using agg, which is a non-GUI backend, so cannot show the figure.
plt.show()
(myenv) vboxuser@Ubuntu1:~/Microservices$
```

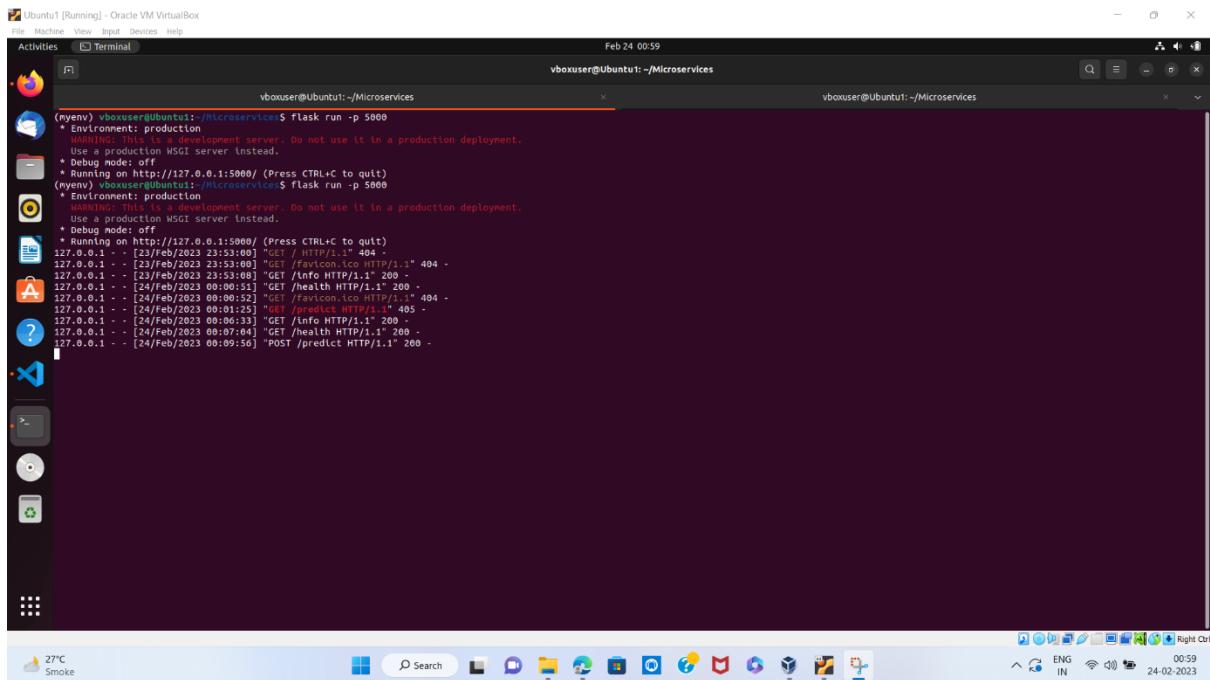
The desktop interface includes a docked application bar at the bottom with icons for various applications like File Explorer, Mail, and Terminal. The system tray shows the date (23-02-2023), time (23:38), battery level (28°C), and network status (ENG IN).

8.Test the Flask web application.

Command : flask run -p 5000



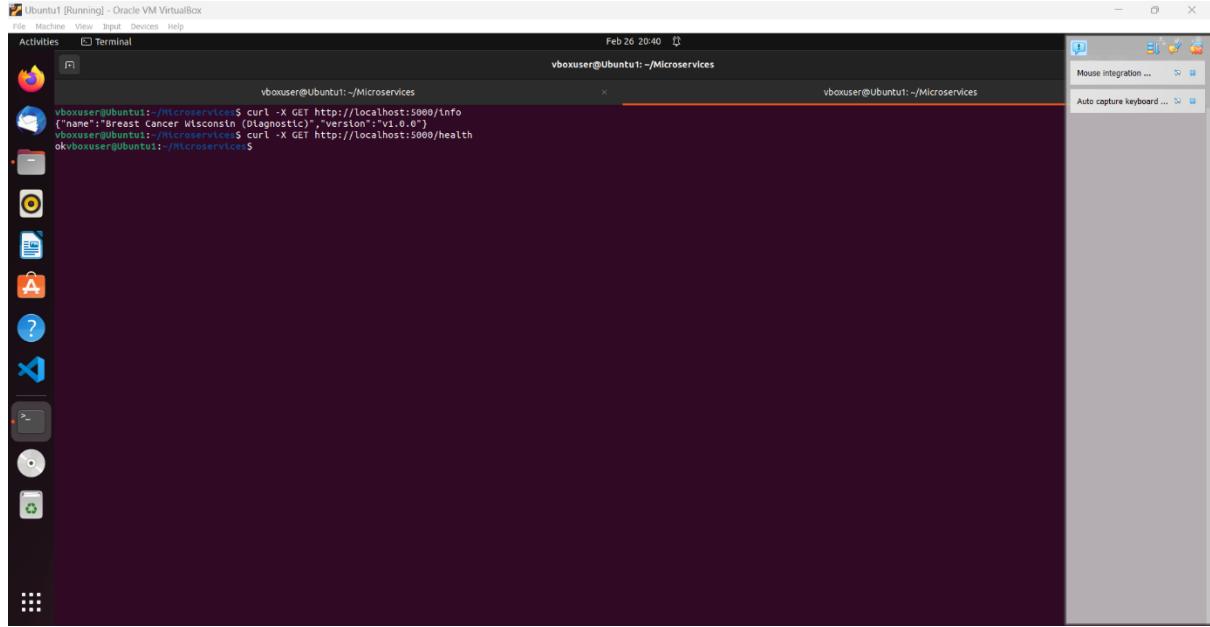




9.Test the application and make predictions using the example calls available in the folder/tests.

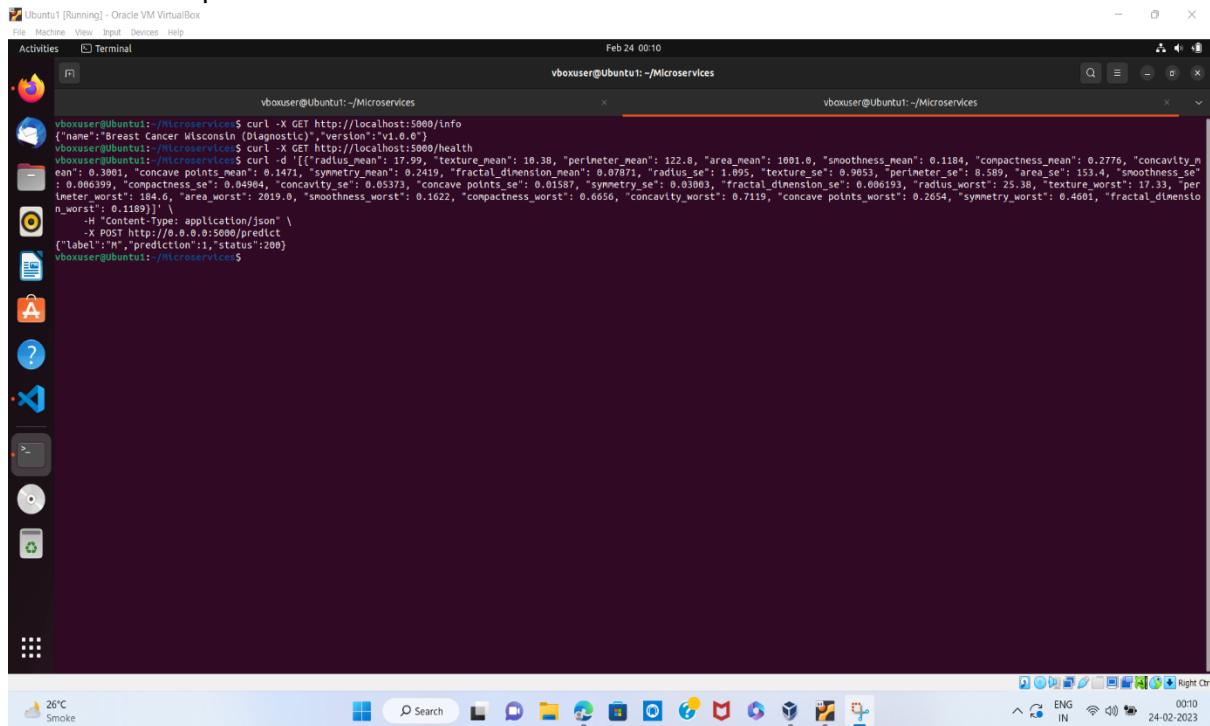
Command : curl -X GET http://localhost:5000/info

```
curl -X GET http://localhost:5000/health
```



```
vboxuser@Ubuntu1:~/Microservices$ curl -X GET http://localhost:5000/info
{"name":"Breast Cancer Wisconsin (Diagnostic)", "version":"v1.0.0"}
vboxuser@Ubuntu1:~/Microservices$ curl -X GET http://localhost:5000/health
ok
vboxuser@Ubuntu1:~/Microservices$
```

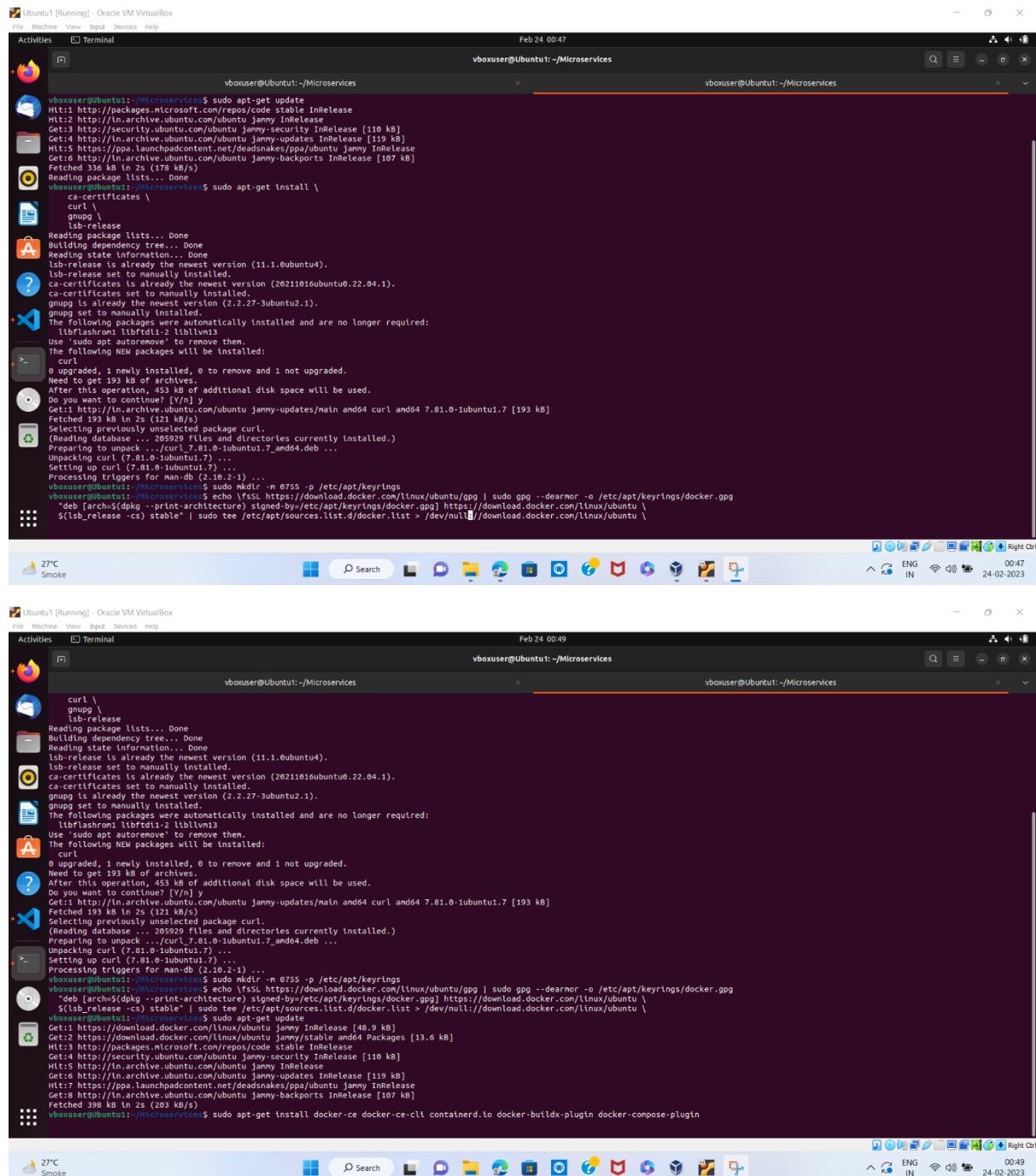
POST method predict :



```
vboxuser@Ubuntu1:~/Microservices$ curl -X GET http://localhost:5000/info
{"name":"Breast Cancer Wisconsin (Diagnostic)", "version":"v1.0.0"}
vboxuser@Ubuntu1:~/Microservices$ curl -X GET http://localhost:5000/health
vboxuser@Ubuntu1:~/Microservices$ curl -d [{"radius_mean": 17.99, "texture_mean": 10.38, "perimeter_mean": 122.8, "area_mean": 1001.0, "smoothness_mean": 0.1184, "compactness_mean": 0.2776, "concavity_mean": 0.3008, "concave points_mean": 0.1471, "symmetry_mean": 0.2419, "fractal_dimension_mean": 0.07871, "radius_se": 1.095, "texture_se": 0.9653, "perimeter_se": 8.589, "area_se": 153.4, "smoothness_se": 0.03009, "compactness_se": 0.04964, "concave points_se": 0.05373, "symmetry_se": 0.01587, "fractal_dimension_se": 0.000193, "radius_worst": 25.36, "texture_worst": 17.33, "perimeter_worst": 18.49, "area_worst": 2039.0, "smoothness_worst": 0.1622, "compactness_worst": 0.656, "concave points_worst": 0.2654, "symmetry_worst": 0.4601, "fractal_dimension_worst": 0.1189}]" \
-H "Content-Type: application/json" \
-X POST http://0.0.0.5000/predict
{"label": "M", "prediction": 1, "status": 200}
vboxuser@Ubuntu1:~/Microservices$
```

10.Create a docker image containing everything needed to run the application.

Docker Installation and creating a docker image



```
vboxuser@Ubuntu1:~/Microservices$ sudo apt-get update
Hit:1 http://packages.microsoft.com/repos/code stable InRelease
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Fetched 338 kB in 2s (178 kB/s)
Reading package lists... Done
vboxuser@Ubuntu1:~/Microservices$ sudo apt-get install \
curl \
gpg \
libcurl \
lsb-release
Reading package lists... Done
Building dependency tree... Done
The following packages were already installed and are now being held:
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016ubuntu0.22.04.1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
libflashrom1 libfdt1:2 liblwn13
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
curl
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 193 kB of archives.
After this operation, 453 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.7 [193 kB]
Fetched 193 kB in 2s (121 kB/s)
Selecting previously unselected package curl.
(Reading database ... 205929 files and directories currently installed.)
Preparing to unpack .../curl_7.81.0-1ubuntu1.7_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.7) ...
Setting up curl (7.81.0-1ubuntu1.7) ...
Processing triggers for man-db (2.16.2-1) ...
vboxuser@Ubuntu1:~/Microservices$ sudo mkdir -p /etc/apt/keyrings
vboxuser@Ubuntu1:~/Microservices$ echo |fssl https://download.docker.com/linux/ubuntu \
$deb [arch=$dpkg --print-architecture] signed-by=/etc/apt/keyrings/docker.gpg https://download.docker.com/linux/ubuntu \
$S[lsb_release -cs] stable | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null || download.docker.com/linux/ubuntu \
27°C Smoke 24-02-2023 Right Ctrl
vboxuser@Ubuntu1:~/Microservices$ curl \
gpg \
libcurl \
lsb-release
Reading package lists... Done
Building dependency tree... Done
The following packages were already installed and are now being held:
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016ubuntu0.22.04.1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
libflashrom1 libfdt1:2 liblwn13
Use 'sudo apt autoremove' to remove them.
The following NEW package will be installed:
curl
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 193 kB of archives.
After this operation, 453 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.7 [193 kB]
Fetched 193 kB in 2s (121 kB/s)
Selecting previously unselected package curl.
(Reading database ... 205929 files and directories currently installed.)
Preparing to unpack .../curl_7.81.0-1ubuntu1.7_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.7) ...
Setting up curl (7.81.0-1ubuntu1.7) ...
Processing triggers for man-db (2.16.2-1) ...
vboxuser@Ubuntu1:~/Microservices$ sudo mkdir -p /etc/apt/keyrings
vboxuser@Ubuntu1:~/Microservices$ echo |fssl https://download.docker.com/linux/ubuntu \
$deb [arch=$dpkg --print-architecture] signed-by=/etc/apt/keyrings/docker.gpg https://download.docker.com/linux/ubuntu \
$S[lsb_release -cs] stable | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null || download.docker.com/linux/ubuntu \
Get:1 https://download.docker.com/linux/ubuntu jammy InRelease [48.9 kB]
Get:2 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [13.6 kB]
Hit:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:4 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:6 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy InRelease
Get:7 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Fetched 399 kB in 2s (203 kB/s)
vboxuser@Ubuntu1:~/Microservices$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
27°C Smoke 24-02-2023 Right Ctrl
```

```

Ubuntu1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Feb 24 00:50
vboxuser@Ubuntu: ~/Microservices
vboxuser@Ubuntu: ~/Microservices

Unpacking slirp4netns (1.0.1-2) ...
Setting up docker-scan-plugin (0.23.0-ubuntu-jammy) ...
Setting up docker-buildx-plugin (0.10.2-1-ubuntu.22.04-jammy) ...
Setting up docker-containernetworking (2.18.0-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker-compose-plugin (2.26.0-1-ubuntu.22.04-jammy) ...
Setting up docker-ce-cll (5.23.0.1-1-ubuntu.22.04-jammy) ...
Setting up liblprng (4.6.1-1build1) ...
Setting up liblprng-extras (5:23.0.1-1-ubuntu.22.04-jammy) ...
Setting up slirpnetns (1.0.1-2) ...
Setting up docker-ce (5:23.0.1-1-ubuntu.22.04-jammy) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-ubuntu1.1) ...
vboxuser@Ubuntu:~/Microservices$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db297f1813e: Pull complete
Digest: sha256:6e8b6f026e0b9c419eadf02d3905dd0952adfeea67543f525c73a0a790ffefb
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (andrea)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
vboxuser@Ubuntu:~/Microservices$ 

27°C Smoke Search ENG IN 00:50 24-02-2023 Right Ctrl

Ubuntu1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Feb 24 00:51
vboxuser@Ubuntu: ~/Microservices
vboxuser@Ubuntu: ~/Microservices

Setting up docker-buildx-plugin (0.10.2-1-ubuntu.22.04-jammy) ...
Setting up containerd.io (1.0.18-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker-scan-plugin (0.23.0-ubuntu-jammy) ...
Setting up docker-ce-cll (5:23.0.1-1-ubuntu.22.04-jammy) ...
Setting up liblprng (4.6.1-1build1) ...
Setting up liblprng-extras (5:23.0.1-1-ubuntu.22.04-jammy) ...
Setting up slirpnetns (1.0.1-2) ...
Setting up docker-ce (5:23.0.1-1-ubuntu.22.04-jammy) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for man-db (2.10.2-1) ...
vboxuser@Ubuntu:~/Microservices$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db297f1813e: Pull complete
Digest: sha256:6e8b6f026e0b9c419eadf02d3905dd0952adfeea67543f525c73a0a790ffefb
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (andrea)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

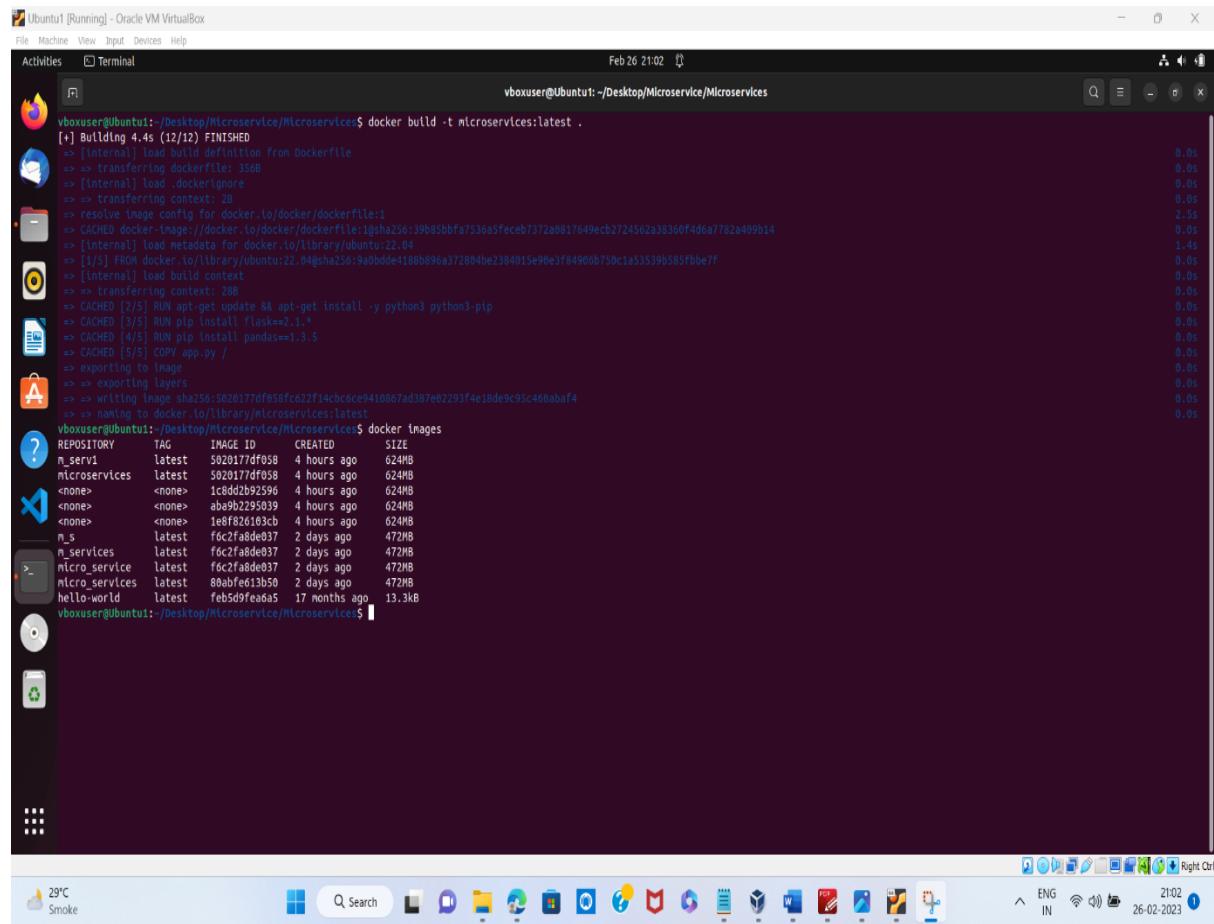
For more examples and ideas, visit:
https://docs.docker.com/get-started/
vboxuser@Ubuntu:~/Microservices$ docker --version
Docker version 23.0.1, build a5e5b1
vboxuser@Ubuntu:~/Microservices$ 

27°C Smoke Search ENG IN 00:51 24-02-2023 Right Ctrl

```

Creating a docker image

Command : docker build -t microservices:latest .

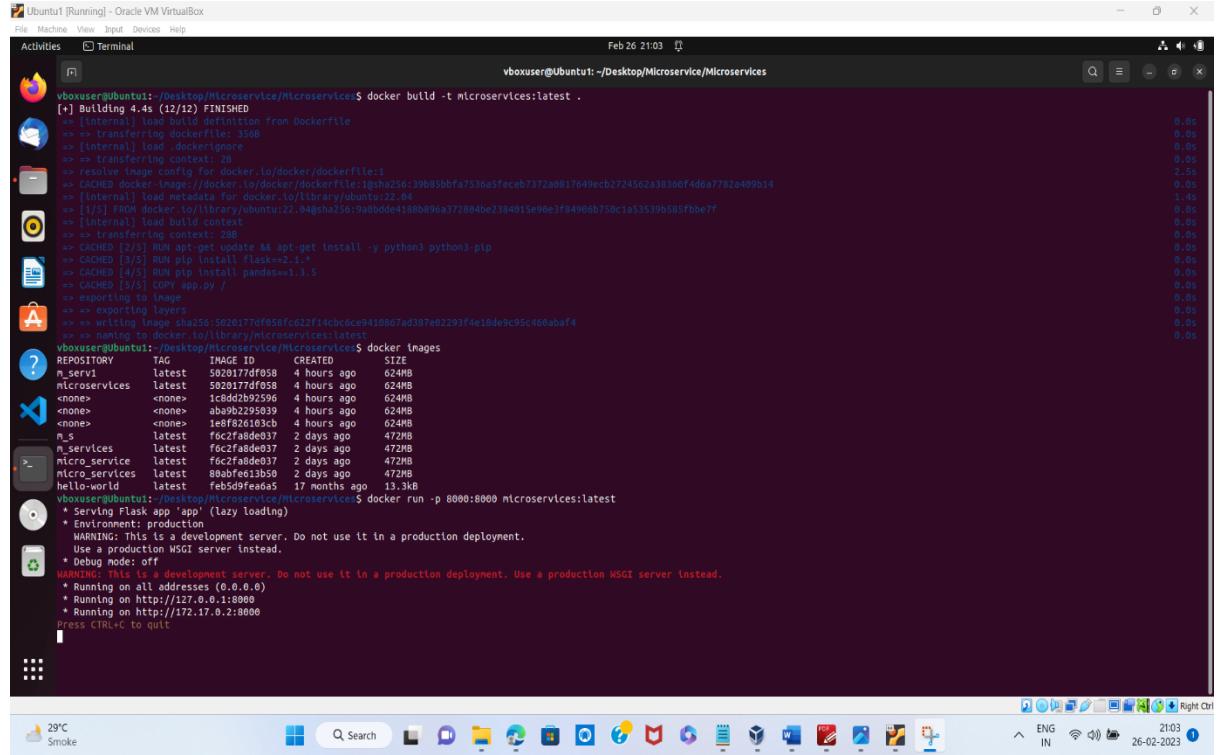


The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices". The terminal content shows the execution of the command "docker build -t microservices:latest .". The output of the command is displayed, showing the progress of the build process, including the creation of a Docker image from a Dockerfile. After the build is completed, the user runs the command "docker images" to list the available Docker images. The terminal window is part of a desktop interface with a dock at the bottom containing various application icons.

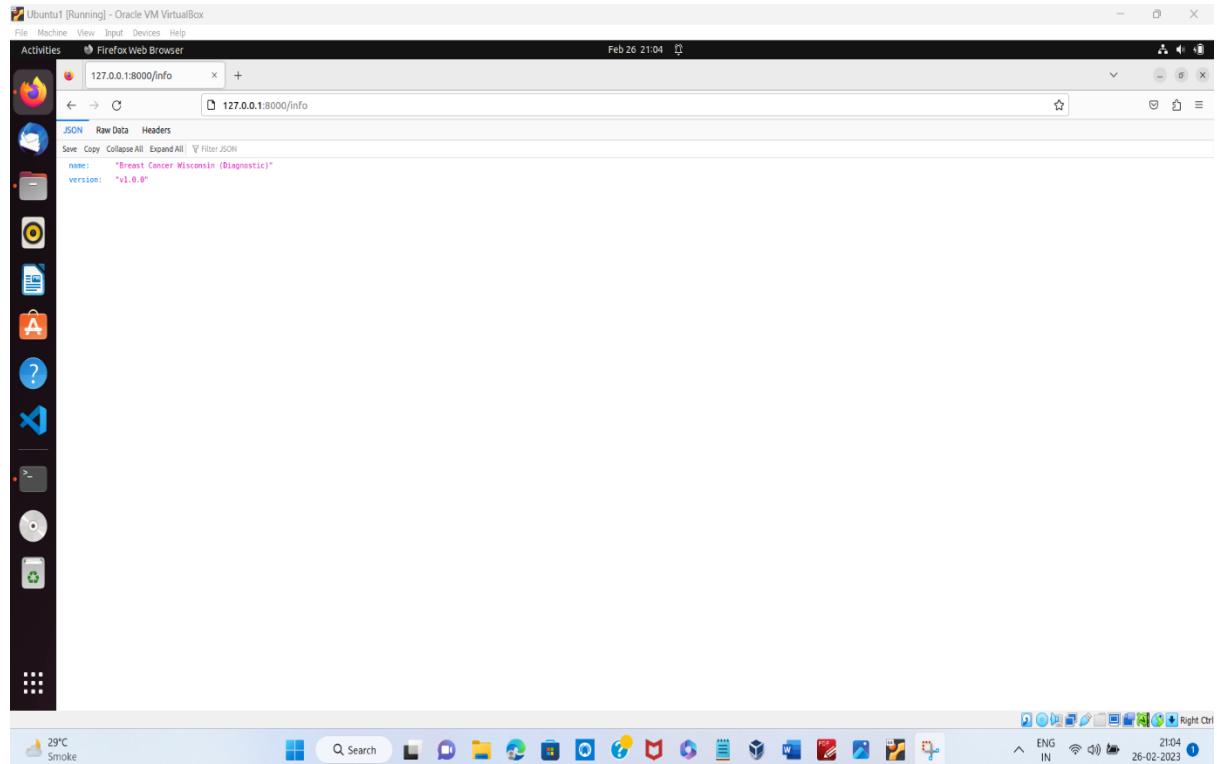
```
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$ docker build -t microservices:latest .
[+] Building 4.4s (12/12) FINISHED
--> [internal] load build definition from Dockerfile
--> transferring dockerfile: 35B
--> [internal] load .dockerignore
--> transferring context: 2B
--> resolving image config for docker.io/docker/dockerfile:1
--> [internal] load metadata for docker.io/library/ubuntu:22.04
--> [1/S] FROM docker.io/library/ubuntu:22.04@sha256:9a0bdde4188b89ea372804be2384015e90e3f84900b750c1a53539b505fbbe7f
--> [internal] load build context
--> >> transferring context: 28B
--> [2/S] RUN apt-get update && apt-get install -y python3 python3-pip
--> [3/S] RUN pip install flask==2.1.*
--> [4/S] RUN pip install pandas==1.3.5
--> [5/S] COPY app.py /
--> exporting to image
--> >> writing image sha256:5d20177df058fc622f14cbc8ce9410867ad387e02293f4e18de9c95c46babaf
--> >> naming to docker.io/microservice/microservices:latest
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$ docker images
REPOSITORY          TAG      IMAGE ID   CREATED             SIZE
m_serv1              latest   5d20177df058  4 hours ago    624MB
microservices        latest   5d20177df058  4 hours ago    624MB
<none>              <none>   1c8dd2b92956  4 hours ago    624MB
<none>              <none>   a8b9b2295039  4 hours ago    624MB
<none>              <none>   1ef8f26103cb  4 hours ago    624MB
n_s                 latest   f6c2fa8ade037  2 days ago     472MB
n_services           latest   f6c2fa8ade037  2 days ago     472MB
micro_service        latest   f6c2fa8ade037  2 days ago     472MB
micro_services       latest   80abfe613b50  2 days ago     472MB
hello-world          latest   feb5d9f9ead5  17 months ago   13.3kB
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$
```

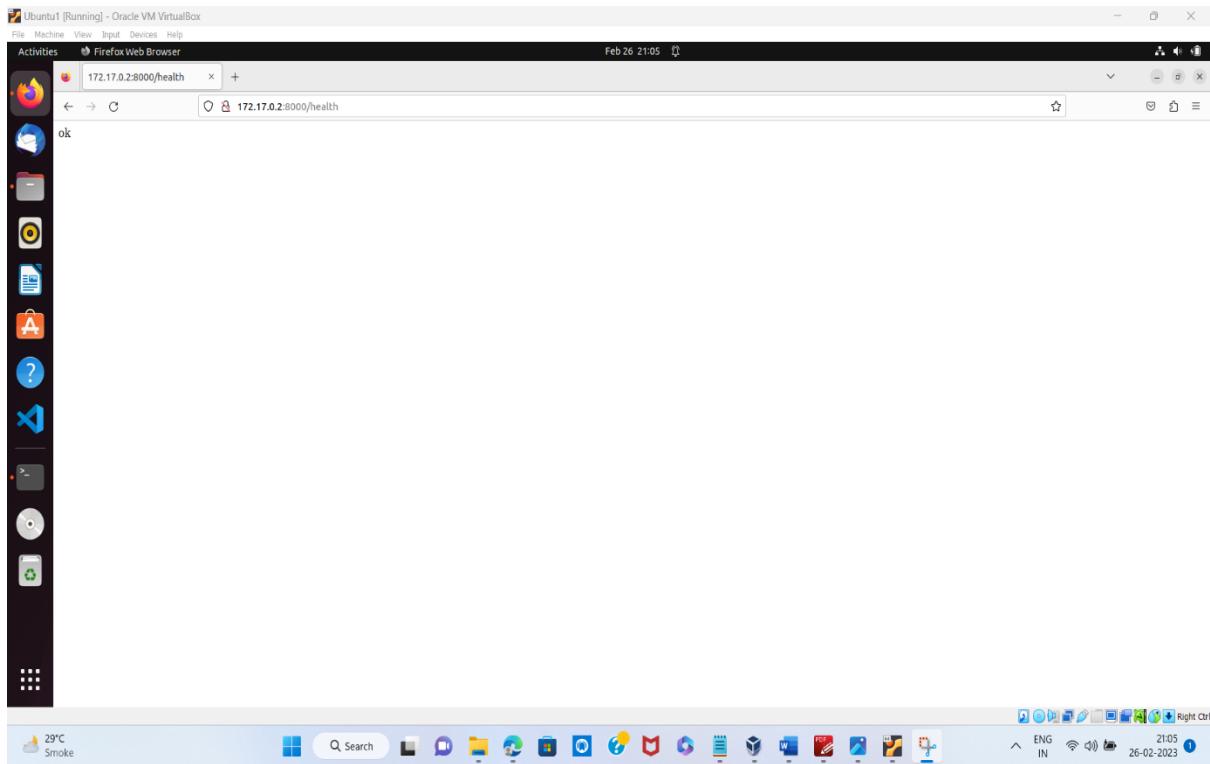
11.Run the containerized application as a prediction service and test it locally by passing some example calls and get the prediction.

Command : docker run -p 8000:8000 microservices:latest



```
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$ docker build -t microservices:latest .
[+] Building 4.4s (12/12) FINISHED
   => [internal] load build definition from Dockerfile
   => [internal] load .dockerignore
   => [internal] transfer context: 2B
   => resolve image config for docker.io/docker/dockerfile:1
   => CACHED docker-timage://docker.io/docker/dockerfile:1@sha256:39b85bfa7536ad5feceb7372a8817649ecb2724562a38360f4d6a7702a409b1a
   => [1/1] FROM docker.io/library/ubuntu:22.04@sha256:9ab0bde4180b896a372864be238401e90e7f84906b750c1a53539b585fbbe7f
      => [internal] load build context
      => [internal] transfer context: 2B
      => CACHED [2/2] RUN apt-get update && apt-get install -y python3 python3-pip
      => CACHED [3/2] RUN pip install flask==2.1.*
      => CACHED [4/2] RUN pip install pandas==1.3.5
      => CACHED [5/2] COPY app.py /
      => exporting to image
      => => writing image sha256:920b177d7058fc022f14cc0ce94100867ad387e02293f4e18de9c95c408abaf
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$ docker images
REPOSITORY          TAG        IMAGE ID       CREATED             SIZE
m_serv1           latest    5020177df058  4 hours ago     624MB
mMicroservices    latest    5020177df058  4 hours ago     624MB
<none>            <none>   1c8dd2b29596  4 hours ago     624MB
<none>            <none>   ab9b2295839  4 hours ago     624MB
<none>            <none>   1ef8f26105cb  4 hours ago     624MB
m_s               latest    f0c2faade037  2 days ago     472MB
mMicroservices    latest    f0c2faade037  2 days ago     472MB
micro_service     latest    80abf0e1b058  2 days ago     472MB
micro_services    latest    feb5d9freada5  17 months ago   13.3KB
hello-world        latest    feb5d9freada5  17 months ago   13.3KB
vboxuser@Ubuntu1:~/Desktop/Microservice/Microservices$ docker run -p 8000:8000 microservices:latest
 * Serving Flask app 'app' (lazy loading)
 * Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8000
 * Running on http://172.17.0.2:8000
Press CTRL+C to quit
```





```

Ubuntu1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Firefox Web Browser
File Machine View Input Devices Help
Activities Firefox Web Browser
172.17.0.2:8000/health 172.17.0.2:8000/health
Feb 26 21:05
ok
29°C Smoke
2105 26-02-2023 Right Ctrl
ENG IN WiFi 2105 26-02-2023

vboxuser@ubuntu:~/Desktop/Microservice/Microservices$ docker build -t microservices:latest .
[+] Building 4.4s (12/12) FINISHED
   [internal] load build definition from Dockerfile
   => transferring dockerfile: 355B
   => [internal] load .dockerignore
   => [internal] load context: 2B
   => resolve image config for docker.io/docker/dockerfile:1
   => [internal] load metadata for docker.io/library/ubuntu:22.04
   => [internal] load build context
   => [internal] transfer context: 2B8
   => [internal] RUN apt-get update && apt-get install -y python3 python3-pip
      CACHED [3/3] RUN apt-get update && apt-get install -y python3 python3-pip
      CACHED [4/4] RUN pip install flask==2.1.1
      CACHED [5/5] RUN pip install pandas==1.3.5
      CACHED [5/5] COPY app.py /
   => exporting to image
   => exporting layers
   => exporting Image image sha256:5e20177df058fc622f14bcace9410867ad87e02293f4e18de9c95c468abaf
   => naming to docker.io/library/microservices:latest
vboxuser@ubuntu:~/Desktop/Microservice/Microservices$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
m_serv1 latest 5020177df058 4 hours ago 624MB
microservices latest 5020177df058 4 hours ago 624MB
<none> <none> 1cb092295939 4 hours ago 624MB
<none> <none> d009b92295939 4 hours ago 624MB
<none> <none> 1efbf26103cb 4 hours ago 624MB
m_s latest f6c2fa0de037 2 days ago 472MB
m_services latest f6c2fa0de037 2 days ago 472MB
micro_service latest f6c2fa0de037 2 days ago 472MB
micro_services latest 80abbfe613b50 2 days ago 472MB
hello-world latest febd5d9feada5 17 months ago 13.3kB
vboxuser@ubuntu:~/Desktop/Microservice/Microservices$ docker run -p 8000:8000 microservices:latest
* Serving Flask app 'app' (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
WARNING: THIS IS A DEVELOPMENT SERVER. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8000
* Running on http://172.17.0.2:8000
Press CTRL+C to quit
*172.17.0.1 - - [26/Feb/2023 15:33:50] "GET /info HTTP/1.1" 200 -
172.17.0.1 - - [26/Feb/2023 15:34:22] "GET /health HTTP/1.1" 200 -

```