



Examen Parcial - CC312

Sergio Sebastian Pezo Jimenez - 20224087G

El notebook y código de esta respuesta se encuentra en el comprimible .zip presentado.

Parte 1

1.1 ¿Cómo añadiría con el comando ip, el IP 192.168.1.50/20 a la interface eth0?

Ejecutaría el siguiente comando para configurar la IP 192.168.1.50/20 en la interface eth0:

```
ip addr add 192.168.1.50/20 dev eth0
```

Donde `add` es el responsable.

1.2 ¿Cómo deshabilitaría con el comando ip, la interface de red enp3s0?

Con el comando `down` podemos deshabilitar la interface de red enp3s0:

```
ip link set enp3s0 down
```

1.3 El comando neighbour se utiliza para ver la dirección MAC de los dispositivos conectados a su sistema. ¿Cómo añadiría con el comando ip, la entrada ARP con el IP 192.168.1.50/24 a la interface enp3s0?

De manera similar en la 1, para la adición de una entrada ARP para el IP 192.168.1.50/24 en la interfaz enp3s0 se realizaría con `add` y la dirección MAC correspondiente:

```
ip neigh add 192.168.1.50 lladdr <dirección_MAC> dev enp3s0
```

1.4 Si tenemos la ruta 192.168.4.0/20 que apunta a la interfaz enp3s0 ¿cómo la eliminamos? 1

Con el siguiente comando `del` :

```
ip route del 192.168.4.0/20 dev enp3s0
```

1.5 Explique qué resultado obtendrá con: `$ sudo nmcli connection add type ethernet ifname enp0s5`

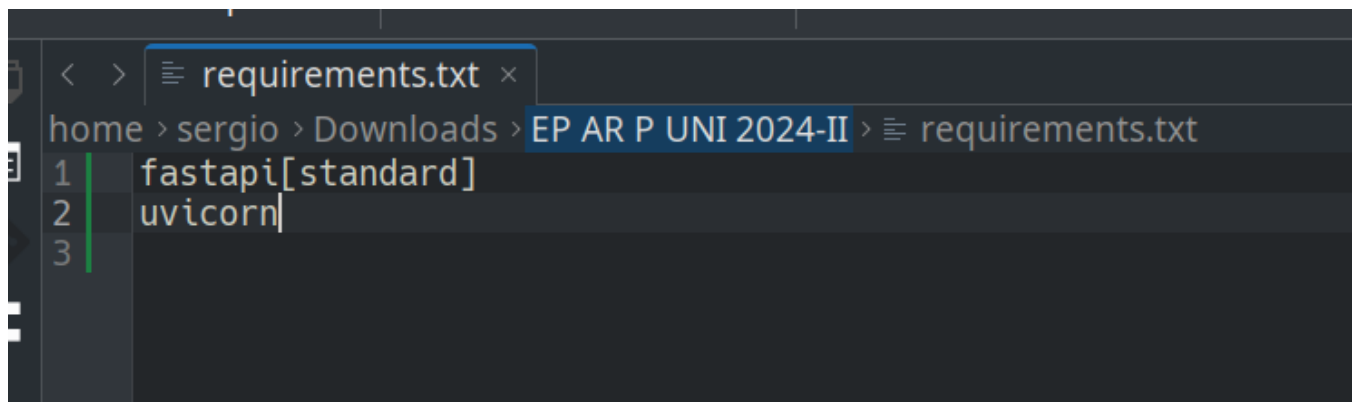
Este comando añade una nueva configuración de conexión al NetworkManager, lo que permitirá gestionar esta conexión Ethernet específica.

Parte 2

Respuesta adjuntada en el notebook `ejercicio-1_examen-parcial_2024-2.ipynb`

Parte 3:

Para construir nuestra API con fast api, primero creamos nuestro `requirements.txt` .

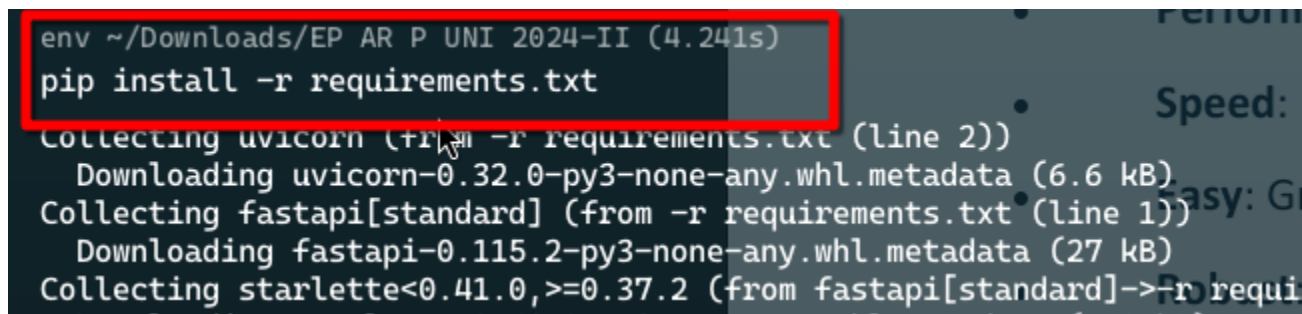
A screenshot of a code editor with a dark theme. The top bar shows the file name 'requirements.txt' with a close button. The breadcrumb navigation shows the path 'home > sergio > Downloads > EP AR P UNI 2024-II > requirements.txt'. The code content consists of two lines: 'fastapi[standard]' on line 1 and 'uvicorn' on line 2. Line 3 is empty. A green vertical cursor is positioned at the end of line 2.

```
< > requirements.txt ×  
home > sergio > Downloads > EP AR P UNI 2024-II > requirements.txt  
1 fastapi[standard]  
2 uvicorn  
3
```

Como usaremos python, primero creamos un entorno virtual llamado `env` pues estoy en Arch Linux.

```
python3 -m venv env
source env/bin/activate
# En caso esté en windows usar
.env\Scripts\activate
```

Una vez activado, ejecutamos `pip install -r requirements.txt`.



```
env ~/Downloads/EP AR P UNI 2024-II (4.241s)
pip install -r requirements.txt
Collecting uvicorn (from -r requirements.txt (line 2))
  Downloading uvicorn-0.32.0-py3-none-any.whl.metadata (6.6 kB)
Collecting fastapi[standard] (from -r requirements.txt (line 1))
  Downloading fastapi-0.115.2-py3-none-any.whl.metadata (27 kB)
Collecting starlette<0.41.0,>=0.37.2 (from fastapi[standard]->-r requi
```

Ahora si en `main.py`, procedemos a escribir nuestro código en nuestro editor favorito, usaré VSCode y completamos el código.

Completamos el código de la siguiente manera:

```
from typing import Union
from fastapi import FastAPI

# Inicializamos la aplicación FastAPI
app = FastAPI()

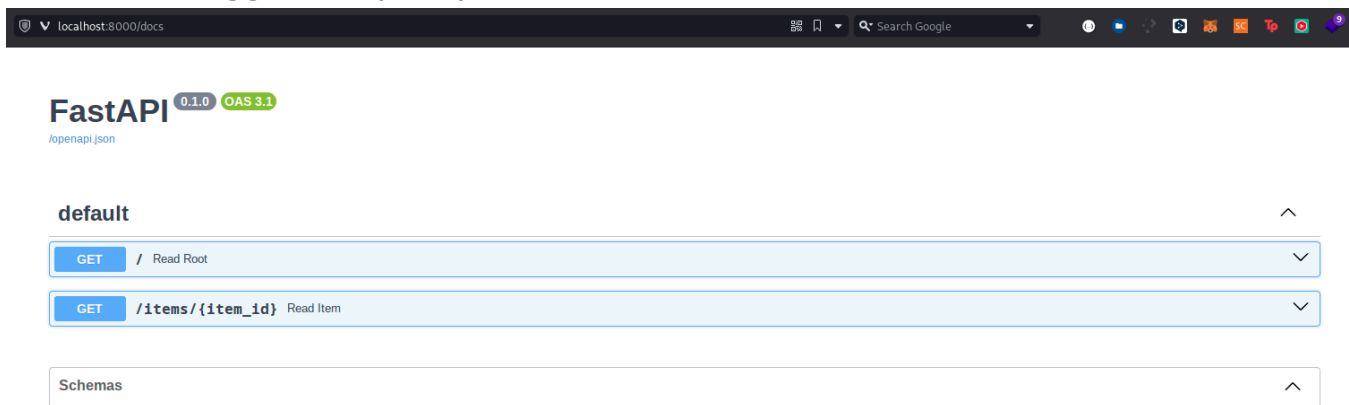
# Definimos la ruta raíz
@app.get("/")
def read_root():
    return {"Hola": "CC 312"}

# Definimos una ruta para manejar items con un ID
@app.get("/items/{item_id}")
def read_item(item_id: int, q: Union[str, None] = None):
    return {"item_id": item_id, "q": q}
```

Ahora ejecutemos `uvicorn main:app --host 0.0.0.0 --port 8000` para iniciar nuestra API en el puerto 800 usando `main.py`.

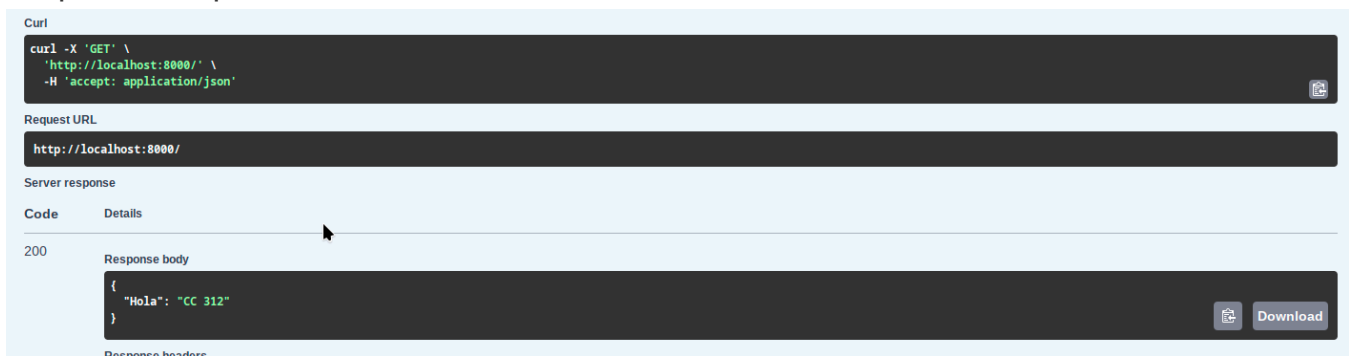
```
env ~/Downloads/EP AR P UNI 2024-II
uvicorn main:app --host 0.0.0.0 --port 8000
INFO: Started server process [98267]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
```

Vamos al swagger de openapi:



Todo ok, veamos que obtenemos al hacer un par de peticiones.

Empecemos por la raíz.



Todo OK. De igual manera para el endpoint `items/{item_id}`, con el cual podríamos usar un query, como:

```
http://localhost:8000/items/1?q=Hola
```

```
~/Downloads/EP AR P UNI 2024-II (0.025s)
curl http://localhost:8000/items/1?q=Hola
{"item_id":1,"q":"Hola"}
```

Bueno, hemos finalizado con la inicialización de nuestra API en python con fastapi.