Mercan's projects

İts Mercan. İn this book, i will share my python projects how i wrote them , what was the top problems while i create this projects.

- the ngrok tunnel chatting websocket app (one society)
- THE EMAİL WATCHER
- THE IP CALCULATER
- Network Monitor with sniff&scapy

the ngrok tunnel chatting websocket app (one society)

The ngrok tunnel , WS-chatting terminal app

--TECHS: *websocket *ngrok tunnel *fastapi *Tuı

- The server : After create the ngrok tunnel ı set up as a server to raspberry pi3. And actually pi was not has a fan.

Than i wrote a python script that watch the cpu degree and if its over heat the wall warning coming up the ssh devices.

pi-thermal-guardian.py

```python
import time
from datetime import datetime
import subprocess

top_heat = 50.0
interval = 5
cooldown = 60

TEMP_PATH = "/sys/class/thermal/thermal_zone0/temp"

def read_temp() -> float:
    with open(TEMP_PATH, "r") as f:
        milli = int(f.read().strip())
    return milli / 1000.0

def wall(msg: str) -> None:
    subprocess.run(["wall", msg], check=False)

def main():
    last_alert = 0.0

    while True:
        try:
            temp = read_temp()
            now = time.time()

            if temp >= top_heat and (now - last_alert) >= cooldown:
                ts = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
                message = (
                    f"[ONE SOCIETY] PI TEMP HIGH: {temp:.1f}°C "
                    f"(has to be: {top_heat:.1f}°C) @ {ts}"
                )
                wall(message)
                print("\a", end="", flush=True)
                last_alert = now

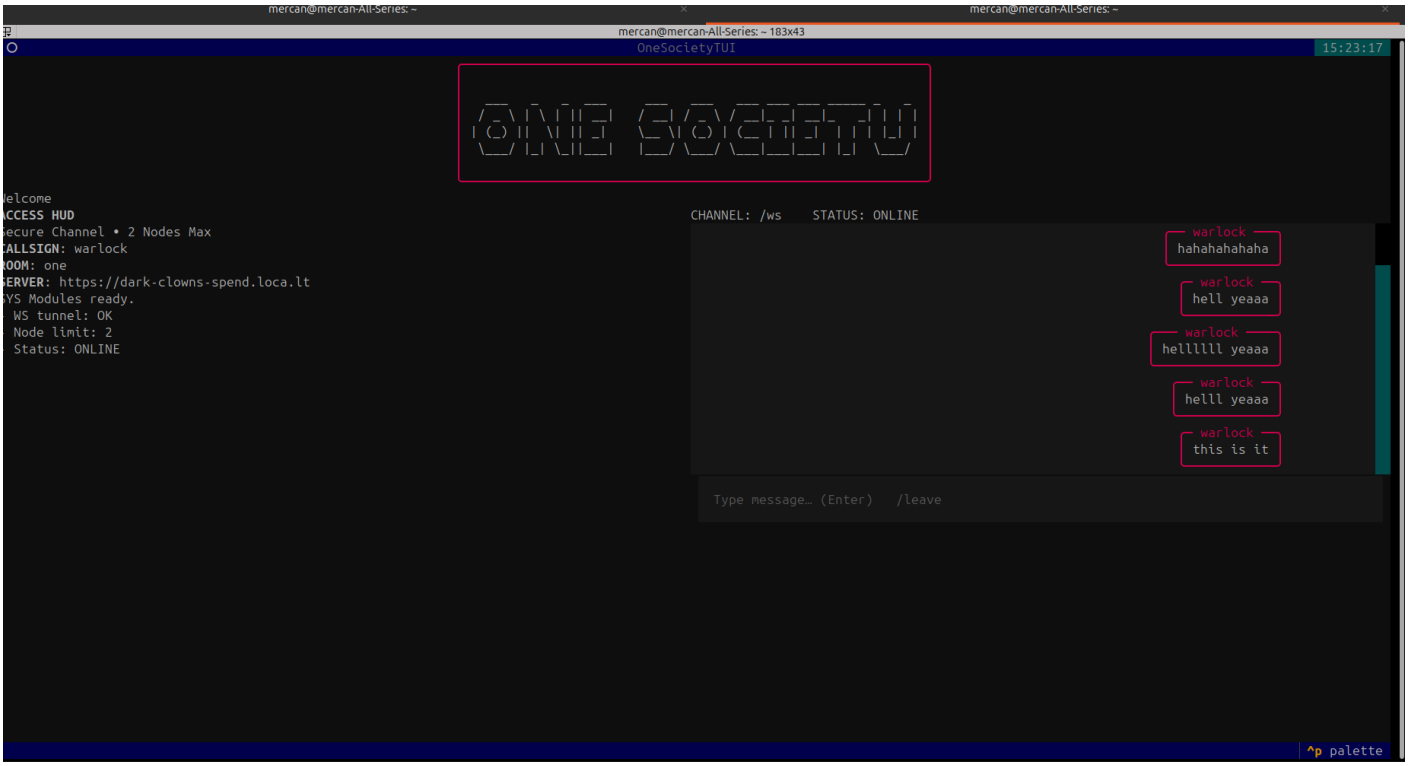
            time.sleep(interval)

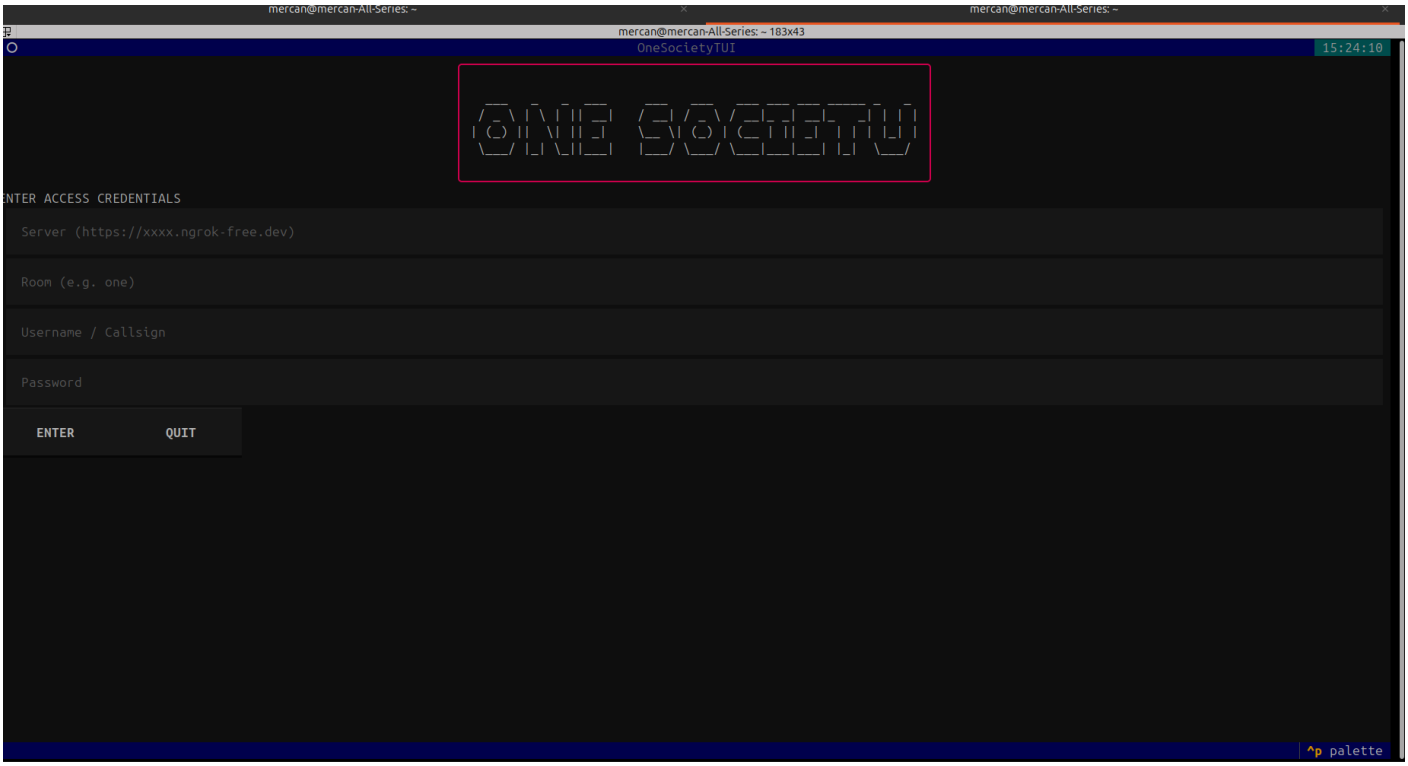
        except Exception:
            time.sleep(interval)

if __name__ == "__main__":
    main()
```

-Then i just set up the project files to server.İt was okey but we was still has some problems that we dont need.

but ı will talk abaut that on problems part..





--PYTHON BACKEND: I use fast api and websocked with backend.The real time room system, To enter room you has to enter password username and the public url that ngrok tunnel give to you. When user exit he erase to list and sending a message another user that (user leave).The all rooms have maximum 2 user rules..

https://github.com/mercan798/Ngrok-Tunnel-WS/tree/main

--THE UI: I use a Tuı library to make it and i thing it was nice, not that complicated. The best part of uı, i try to

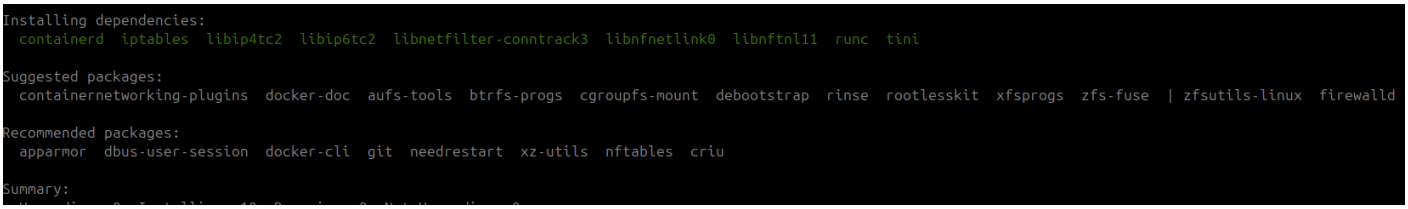
keep it simple and ı thing its looking good at all. And the css ı will not making special space for css ı hate it its just a bunch of colors, i think i have to beat this habbit like i said the megagigawhat page but this is me and my book so i cant keep secrets.. I make this project for just me and my uncle.

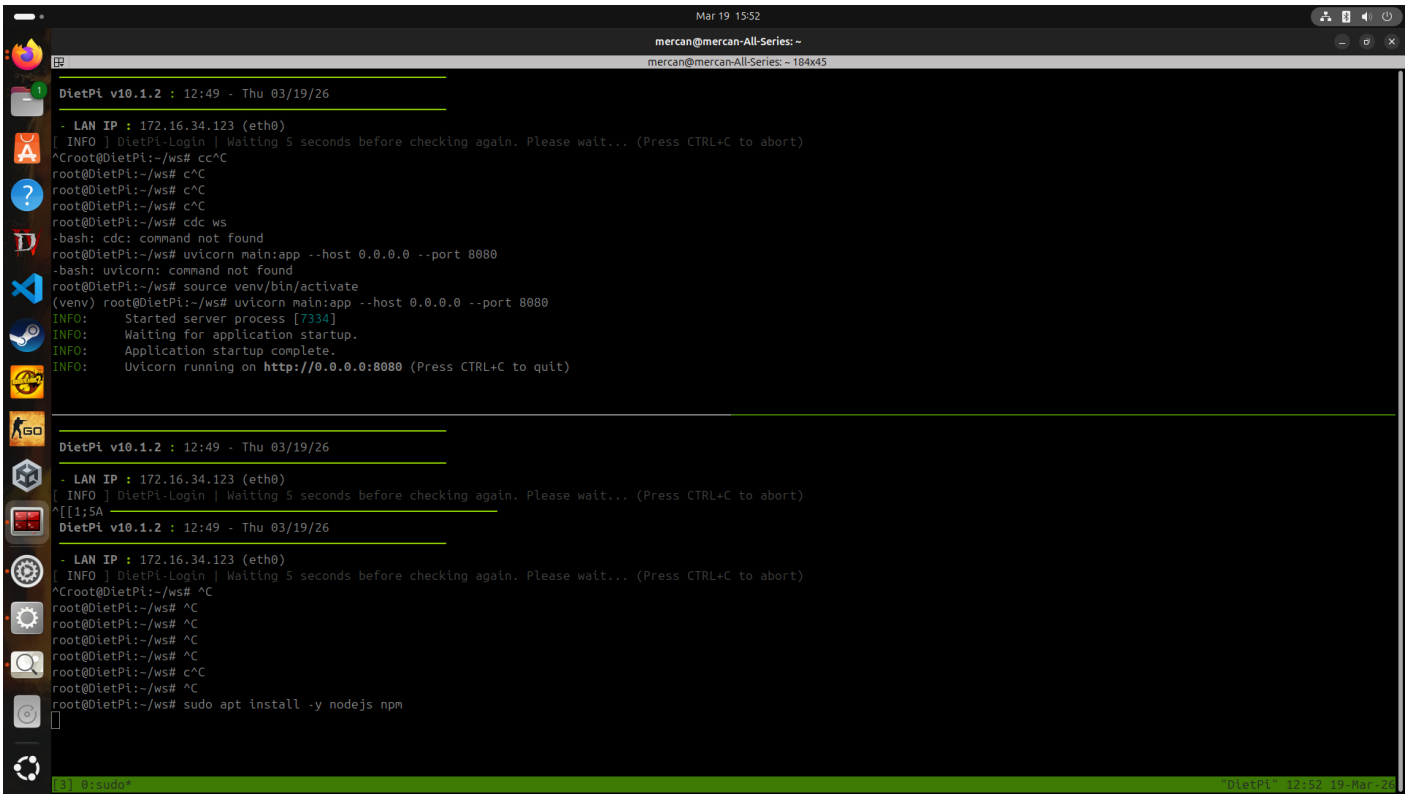
--HOW WE ARE USE İT??: So to use it we have to set up all files to each computer and run the term.py , then a security wall will waiting for us and we give all infos. Like i said we have our ofices the server that keep ngrok tunnel and the project files, and the degree watcher.

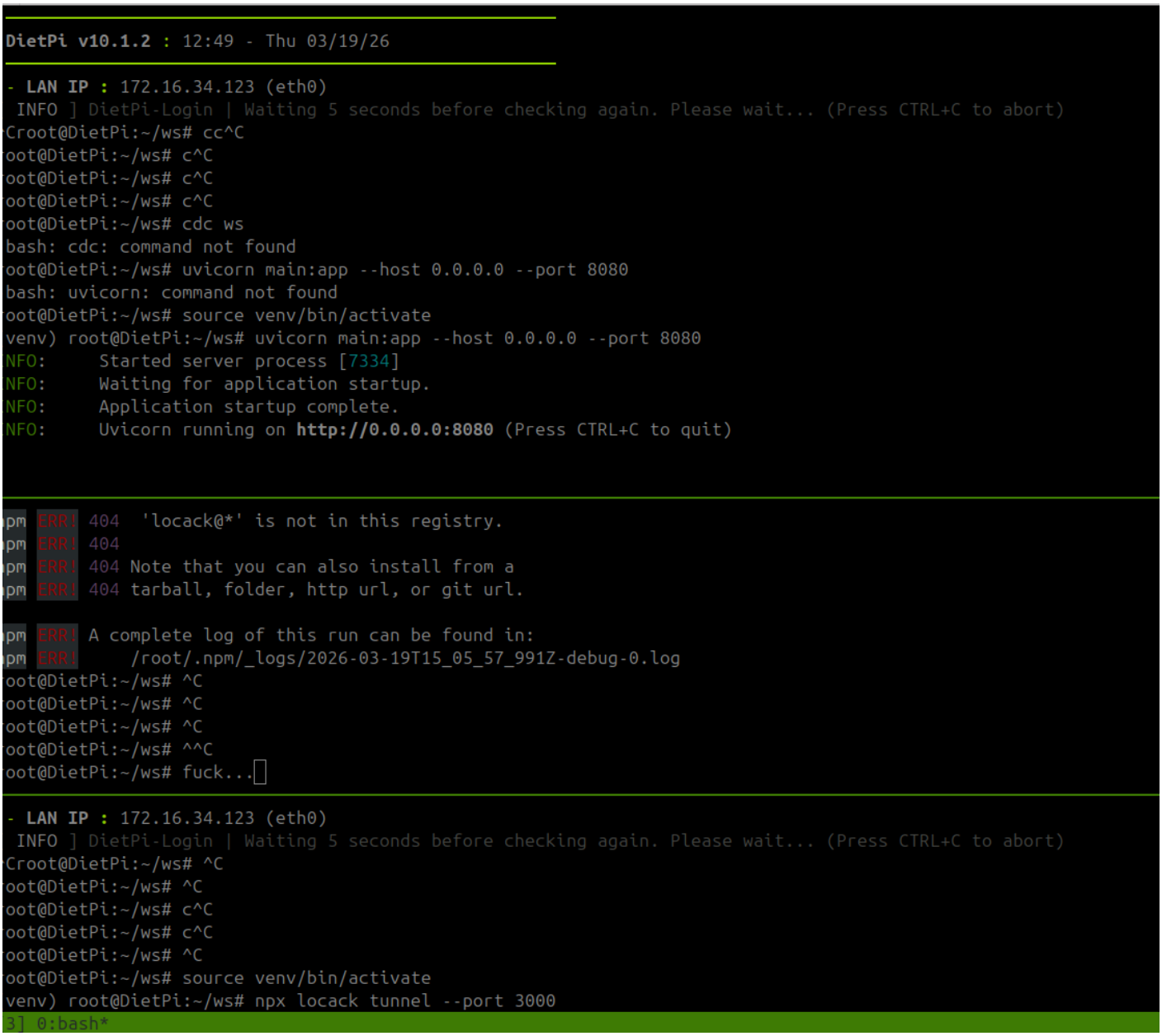
--THE PROBLEMS WHİLE İ BUİLD THİS PROJECT:

- The Type Mismatch: When ı develop this project, i set up rooms as rooms ={} right. But user data sometimes take over like a list. And the system was waiting dictionary ,but user method it was not suitable actually.
- The solution: İ gave the user the user id. İt was become the dic alredy. For every websocket object creatunique id. Kind of working. I beat this problem like this. İt shows ı need to work on python fundamentals more :))
- The 502 504 problems : The fast api was aint working for a while and never find out till i take a the problem Ngrok cant reach the backend problem. Then ı turn it on this was the soludition.

--The token proplem : i have out of token and i cacnt do billing for now and i will change tunnel as local unnel for now . But now i will seet up docker to dietpi ;




Mar 19 15:52
mercan@mercan-All-Series: ~
mercan@mercan-All-Series: ~ 184x45
DietPi v10.1.2 : 12:49 - Thu 03/19/26
- LAN IP : 172.16.34.123 (eth0)
[INFO] DietPi-Login | Waiting 5 seconds before checking again. Please wait... (Press CTRL+C to abort)
^Croot@DietPi:~/ws# cc^C
root@DietPi:~/ws# c^C
root@DietPi:~/ws# c^C
root@DietPi:~/ws# c^C
root@DietPi:~/ws# cdc ws
-bash: cdc: command not found
root@DietPi:~/ws# uvicorn main:app --host 0.0.0.0 --port 8080
-bash: uvicorn: command not found
root@DietPi:~/ws# source venv/bin/activate
(venv) root@DietPi:~/ws# uvicorn main:app --host 0.0.0.0 --port 8080
INFO: Started server process [7334]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8080 (Press CTRL+C to quit)
DietPi v10.1.2 : 12:49 - Thu 03/19/26
- LAN IP : 172.16.34.123 (eth0)
[INFO] DietPi-Login | Waiting 5 seconds before checking again. Please wait... (Press CTRL+C to abort)
^[[1;5A
DietPi v10.1.2 : 12:49 - Thu 03/19/26
- LAN IP : 172.16.34.123 (eth0)
[INFO] DietPi-Login | Waiting 5 seconds before checking again. Please wait... (Press CTRL+C to abort)
^Croot@DietPi:~/ws# ^C
root@DietPi:~/ws# ^C
root@DietPi:~/ws# ^C
root@DietPi:~/ws# ^C
root@DietPi:~/ws# c^C
root@DietPi:~/ws# ^C
root@DietPi:~/ws# sudo apt install -y nodejs npm
[3] 0:sudo*
"DietPi" 12:52 19-Mar-26



NOTE: The problems arent just them but, ı have try to share only higlight problems so , this is it

I wrote this code to myself and my uncle , i am glad if you check it out !! [the ngrok tunnel chatting websocket app (one society)]

THE EMA?L WATCHER

The IMAP gmail watcher.Follow your emails and when email comes play mp3 alarm.. Also you can see the your ip to right corner. And you can see your last 10 emails on your screen.Then i have just setup on raspberry pi 1 2011(b) :]] Do you now why. I will talk abaut it problems part....

--HOW CAN YOU USE İT : I wrote the auto startup script so when you enable it its automaticly run.When it open it will ask you tu ypur gmail and app password. I will talk abaut that later.

```
    set -euo pipefail

    APP_DIR="/home/satisduyuru/gmail"
    VENV_PY="$APP_DIR/venv/bin/python"
    APP_FILE="$APP_DIR/ui.py"

    if [[ ! -x "$VENV_PY" ]]; then
      echo "Python not found at $VENV_PY" >&2
      exit 1
    fi

    cd "$APP_DIR"
    exec "$VENV_PY" "$APP_FILE"
```

```
[Unit]
Description=Gmail Watcher (Textual UI)
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
User=daisy
WorkingDirectory=/home/daisy/gmail
ExecStart=/home/daisy/gmail/start_gmail_watcher.sh
Restart=on-failure
RestartSec=2
Environment=PYTHONUNBUFFERED=1

[Install]
WantedBy=multi-user.target
```

AND TO SETUP ALL CODE --- https://github.com/mercan798/IMAP-GMA-L-PROJECT/blob/main/gmail-watcher.service

--WHAT TECHS I USE WHİLE I WROTE İT? : imaplib(for gmails) ,Tuı(frontend) ,pygame(for alarm) The question on yur mind " why pygame" i now ı can use the playsound, yes i will change it , mybe i

alredy did.

--GMAİL APP PASSWORD:

If login fails with "Invalid credentials", you need a Gmail App Password:

Go to your Google Account security settings Enable 2-Step Verification if not already enabled Generate an App Password for Mail Use that 16-character password in the login screen

--DATA STORAGE:

The app stores credentials in credentials.json in the project root Last seen email UID is stored in watcher_state.json to track new emails Credentials are deleted when you log out

THE PROBLEMS WHİLE İ AM WROTE THE PROJECT:

FİRST "THE RARSPBERRY Pİ": Okey unfortunatly i was set up this project 3 times and 3 diffrent pc's. Why?

firs pi3 model b+ i set it but 3 hours later, the card holder is broken then i have to change same pi3, and then it was not opened at all ,because it wasnt take electric his CPU ,i understod that from the lights, its never turn green light even when boot cart wasnt plug. Okey then i decited to set up pi 1 that i talk abaut. And pi 1 was not coming with the soundcard , wifi ,bluethot. Then we buy for customer. Okey the setup was solved... Lets talk abaut the code problems...

SECOND THE MEMORY: What was the problem?

When the program started for the first time, last_uid was empty (None).

So this line:

```
if uid_str != state.get("last_uid"):
```

became:

5842 != None ? True

The program thought the last email was a new email.

So it triggered alarm for old mails.

Which code fixed it?

This code:

```
if not self.state.get("last_uid"):
    M = imap_login(self.email_user, self.email_pass)
    uid = last_uid(M)
    if uid:
        uid_str = uid.decode() if isinstance(uid, bytes) else str(uid)
        self.state["last_uid"] = uid_str
        save_state(self.state)
```

What does this code do?

When program runs first time, it takes the current latest email UID, saves it as reference, does not trigger alarm..

this is how i build this program and of course this in not the all problems just the ones to worth it to highlight.

THE IP CALCULATER

--How i wrote : So i wrote ip calculater with the ip adress library python. İt was good and chill the library make it ahhaha:) For UI i use the python ui library streamlit. İts make it easier.

--What the code does: Takes an IP address (with CIDR) from the user,Example input: 192.168.1.10/24 ,CIDR (/24) tells how big the network is ,Creates a network object using Python’s ipaddress module ,This lets the program calculate things like network address, broadcast, and hosts. And returns all info ass dictionary..

--What does it calculate : Network adress , Broadcast Adress , Subnet Mask , Total IP , Usable host , First Host , Last host... Also can handle the networks like /31 i /32 .

```python
import ipaddress

def calculate(input):
    network = ipaddress.ip_network(input, strict=False)
    networkAdress = network.network_address
    broadcast = network.broadcast_address
    subnetMask = network.netmask
    totalTips = network.num_addresses

    if totalTips > 2:
        usable_hosts = totalTips - 2
        first_host = list(network.hosts())[0]
        last_host = list(network.hosts())[-1]
    else:
        usable_hosts = totalTips
        first_host = None
        last_host = None

    return {
        "network_address": str(networkAdress),
        "broadcast_address": str(broadcast),
        "subnet_mask": str(subnetMask),
        "total_ips": totalTips,
        "usable_hosts": usable_hosts,
        "first_host": str(first_host) if first_host else "N/A",
        "last_host": str(last_host) if last_host else "N/A"
    }

if __name__ == "__main__":
    ip_input = input("gir")
    if "/" not in ip_input:
        ip_input = ip_input.strip() + "/24"
    result = calculate(ip_input)
```

-- THE PROBLEMS AND THE THİNGS HAS TO BE DEVELOPED:

-- Firt The CIDR :As you can see even user not enter the cidr, my code add it the defoult ip user enter. İ made it couse the code was cras when user not enter it but , i thing its still nor so good it can be reason to many crashes still.

The soludition : I am gonna add the input that ask user for the CIDR (/32, /24) as you can see its now only /24. What would happend if user want to enter /32. This is why i will add this development..

Network Monitor with sniff&scapy

I wrote a network monitor that can queries port protocol and only when its UDP or TCP code can show port number . Also code checks IP addresses in each packet. If the source IP is my computer, it is outgoing (OUT). If the destination IP is my computer, it is incoming (IN). If neither, it is OTHER

--THE CODE AND OUTPUT ->>

```
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
OUT | TCP | 172.16.34.112  :43012 → 139.144.71.6   :443
IN  | TCP | 139.144.71.6   :443   → 172.16.34.112  :43012
```

- As we can see ,the tool give us the going , coming ip's and query the port protocol if its TCP , UDP give us to port number ,if its other return other and nothing.

```python
#şimdi ağ kartından geçen veri paketlerini yakalıcaz belkide port number de eklerim ( eklendi hemde protokol sorgusu da yapıyor :/)

import socket
import psutil
from scapy.all import sniff

def get_my_ip():  1 usage
    for interface, address in psutil.net_if_addrs().items(): #bütün ağ kartlarını geziyorum geziyorum
        for address in address: #her kartın içindeki adreslere bakıyyuorum

            if address.family == socket.AF_INET and not address.address.startswith("127"): # localhost ve ipv4 olanları atla
                return address.address

Local_ip = get_my_ip()
print(" LOCAL IP:",Local_ip)


def listen_network(packet):  1 usage
    if packet.haslayer("IP"):
        src = packet["IP"].src
        dst = packet["IP"].dst

        if packet.haslayer("TCP"):
            sport = packet["TCP"].sport
            dport = packet["TCP"].dport
            protocol = "TCP"
        elif packet.haslayer("UDP"):
            sport = packet["UDP"].sport
            dport = packet["UDP"].dport
            protocol = "UDP"
        else:
            sport, dport = "-", "-"
            protocol = "Other"


        if src == Local_ip:
            direction = "OUT"
        elif dst == Local_ip:
            direction = "IN"
        else:
            direction = "OTHER"


        print(f"{direction:<4} | {protocol:<4} | {src:<15}:{sport:<5} → {dst:<15}:{dport:<5}")



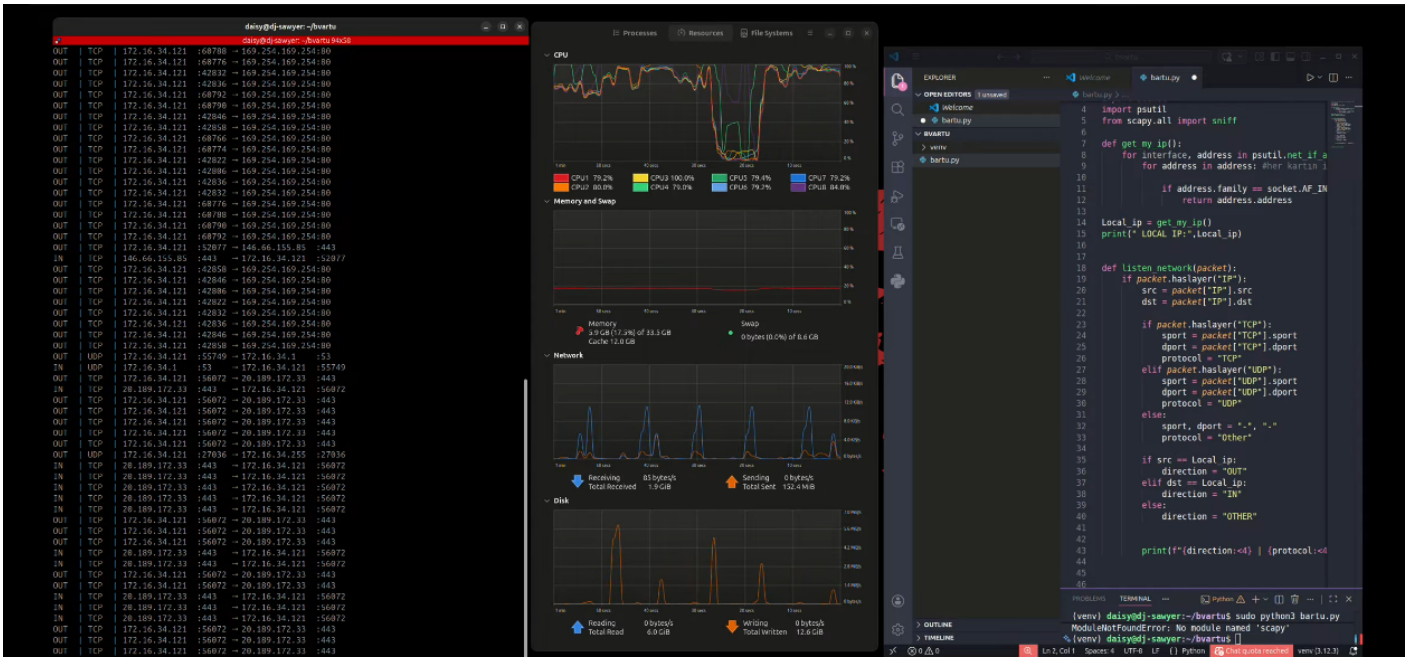
sniff(prn=listen_network)
```

- Mybe i might turn this code to website, aint wrote some filtering for now ,if its projct get bigeer, for sure i will wrote.

--CPU USSAGE TEST --> I have test it for 2 pc (i5 and i7)

? 7 WHAT ?

I now his is shouldent be that much , its becouse the monitor that connected the pc , its huge . So CPU cant reach it :/



i 5

Okey this is what has to be normally its can be reach %40 but stabilizied %20....

