**TREN KOTA DAN KABUPATEN DI INDONESIA**

**Disusun Oleh Kelompok 19:**

Muhammad Ikhwan Perwira (09011282025077)

Dan anggota lainnya

**Fakultas:**

Ilmu Komputer

**Jurusan:**

Sistem Komputer (3A)

**Mata Kuliah:**

Pemrosesan Paralel

**Dosen Pengampu:**

AHMAD HERYANTO, S. KOM, M.T

**Universitas Sriwijaya**

**Tahun Ajaran 2021/2022**

**ABSTRAK**

Laporan ini dilatarbelakangi untuk memenuhi tugas proyek Pemrosessan Parallel. Pada contoh kasus Pemrosessan Parallel kami adalah menggambil nilai trending 5 tahun terakhi dari suatu daftar kata (wordlist). Wordlist (Input) tersebut berisikan nama kota dan kabupaten di Indonesia. Output yang diharapkan adalah file yang berformat CSV dengan isi kolamnya adalah kolom “kota” dan kolom “trend” dimana interval nilai trend adalah 0 sampai 100

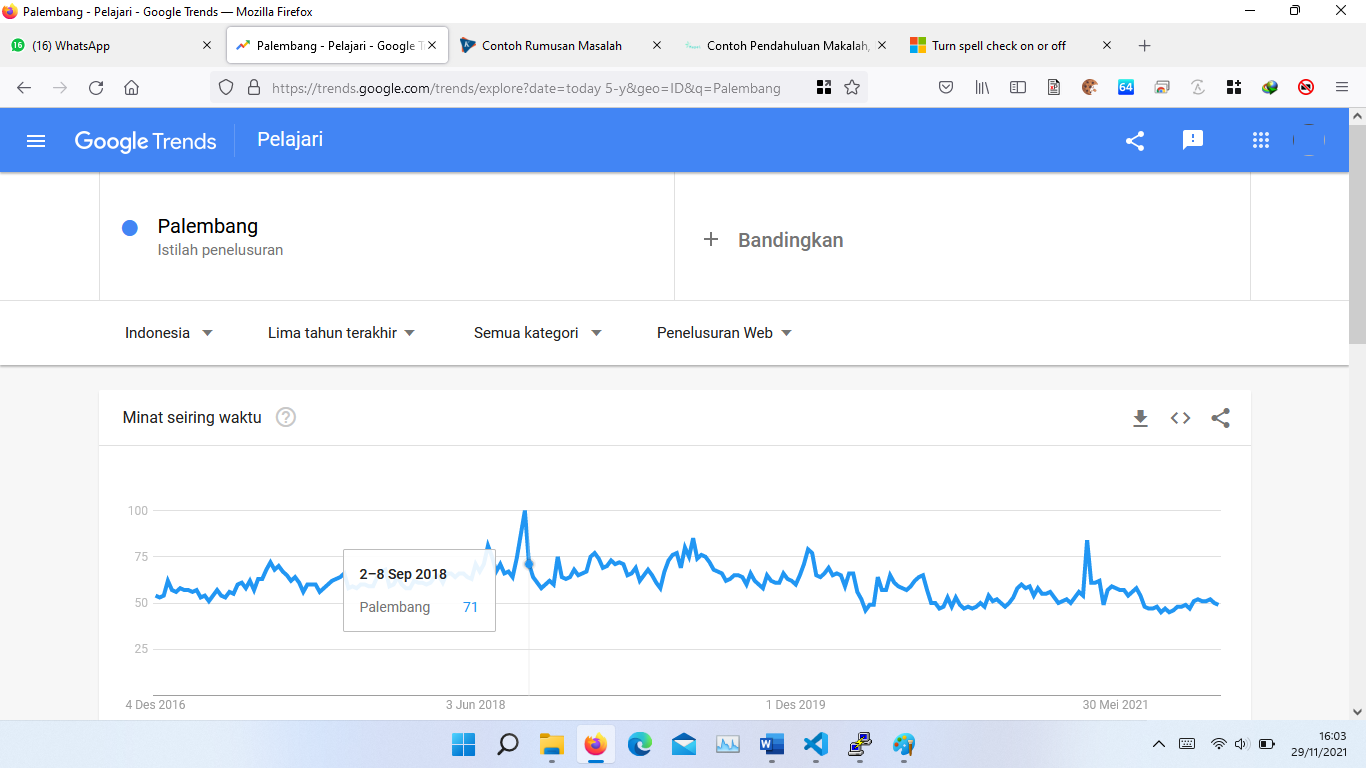
Jika nilai trend semakin mendekati 100 naka semakin popular, dan sebaliknya jika nilai trend semakin mendekati 0 maka semakin tidak popular.

**PENDAHULUAN**

Tren menurut KBBI adalah gaya mutakhir, Nilai Tren adalah ukuran kualitatif dari suatu gaya mutakhir. Tren dipengaruhi oleh popularitas yang dibuat oleh masyarakat. Jika Nilai Tren semakin tinggi maka objek tersebut semakin baru, sedangkan semakin rendah nilai Tren maka objek tersebut bisa disebut Kuno. Dalam kasus ini kami mencari semua kemungkinan nilai trend dari nama kota dan kabupaten di Indonesia dengan **Google Trends**

**Google Trends** adalah layanan google yang memungkinkan user mendapatkan nilai trending dari suatu *keyword* secara gratis. Sehingga kami dapat memanfaatkan layanan tersebut untuk mencari nilai trending dari semua nama kota dan kabupaten di Indonesia 5 tahun terakhir. Disini kami menggunakan 100% bahasa pemrograman **python.** Arsitektur Jaringan yang kami gunakan adalah master-slave. Perangkat yang kami gunakan terdapat 3 slave dan 1 master (cloud) serta server **Google Trends**  itu sendiri (trends.google.com)

Dengan adanya program yang kami buat diharapkan pembaca dapat mengetahui nilai-nilai trend dari tiap kota tanpa perlu mencari satu persatu dari web browser.



**RUMUSAN MASALAH**

1. Apa saja perangkat slave dan master yang digunakan dalam kasus ini?
2. Apa arsitektur jaringan yang digunakan?
3. Bagaimana slave berkomunikasi dengan master dan sebaliknya?
4. Bagaimana cara mendapatkan nilai trend dari suatu keyword?
5. Bagaimana intisari pemrosessan parallel dari contoh kasus ini?

**ISI**

**Perkenalan Perangkat Yang Digunakan:**

**Master** yang kami gunakan adalah server Muhammad Ikhwan Perwira dengan spesifikasi server:

**MASTER**  
Public IP: 27.112.79.120 (Static)  
Domain: ikhwanperwira.ml  
OS: Ubuntu 20.04 LTS  
RAM: 1 GB  
CPU: 1

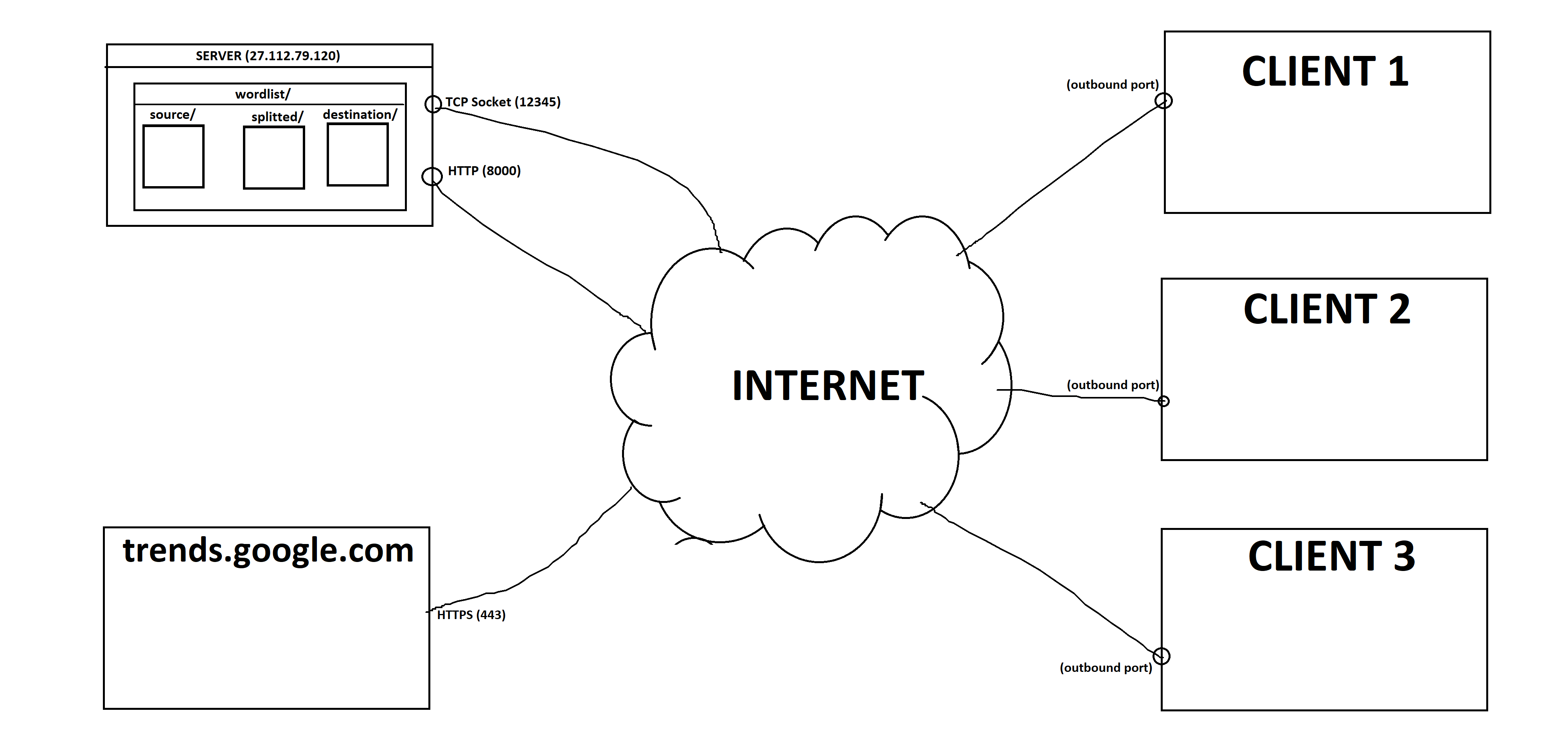
**Slave** yang kami gunakan adalah Laptop Muhammad Ikhwan Perwira, Server hasil *hack****,*** danGoogle Colab Runtime.

**SLAVE 1 (Laptop Muhammad Ikhwan Perwira)**  
Public IP: (Dynamic)  
OS: Windows 11  
RAM: 8 GB  
CPU: 6

**SLAVE 2 (Server hasil hack)**  
Public IP: (Dynamic)  
OS: Ubuntu 16.04 LTS  
RAM: 2 GB  
CPU: 2

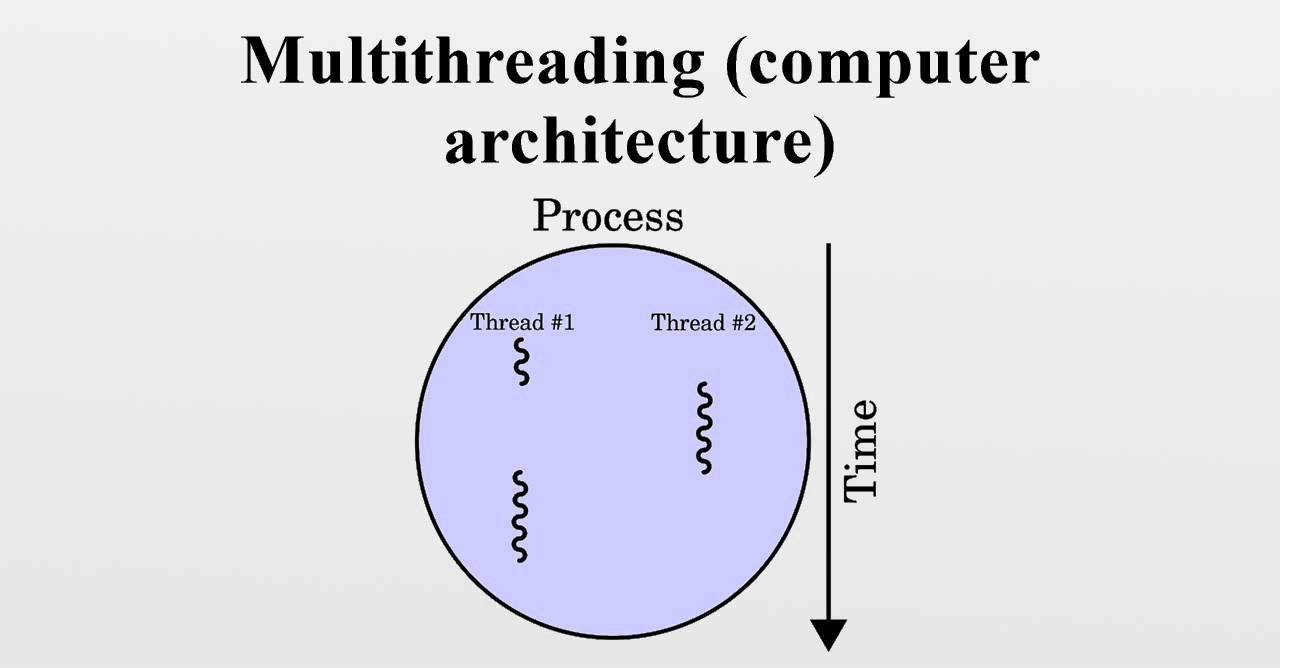
**SLAVE 3 (Google Colab Runtime)**  
Public IP: (Dynamic)  
OS: Ubuntu 20.04 LTS  
RAM: 12 GB  
CPU: 2

**Arsitektur Jaringan**

Arsitektur Jaringan yang digunakan ada pada gambar di atas, dapat dilihat Area Network yang digunakan adalah Internet bukan Local Area Network. Seperti yang kita ketahui topology Internet adalah Mesh Partially.

**MULTITREADING**

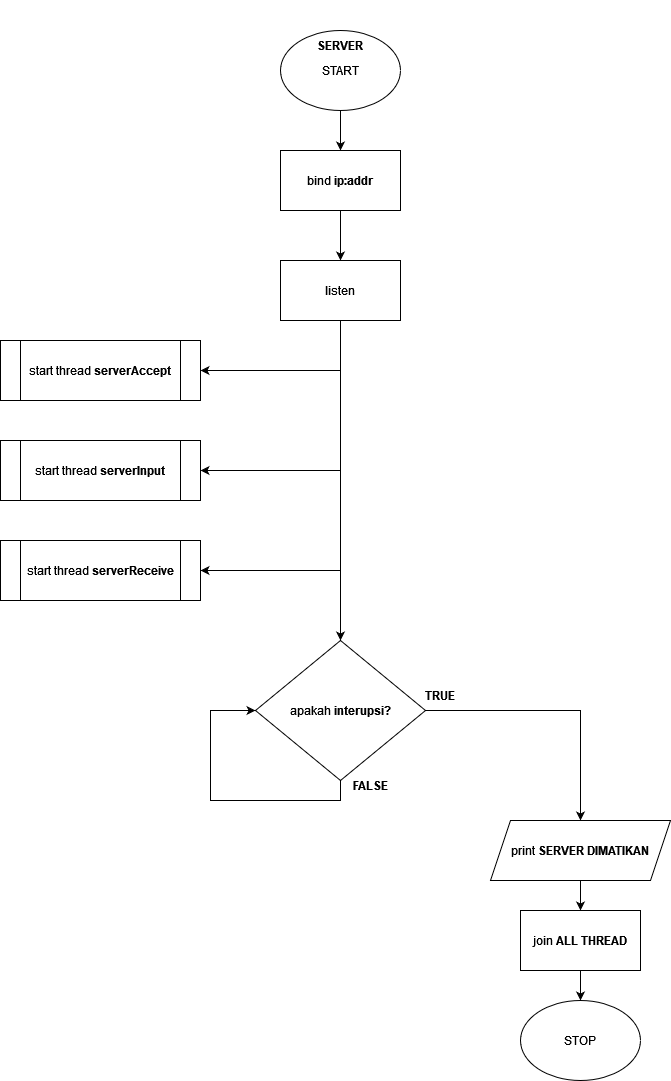
Dalam sebuah proses bisa terdiri dari beberapa thread. Artinya beberapa tugas di suatu proses dapat dijalankan. Konsep paralelisme pada multithreading adalah prosesor melakukan sebagian tugas (thread) atau utas kemudian berganti ke sebagian utas lainnya dan kembali lagi ke utas sebelumnya sehingga kedua tugas terlihat dikerjakan secara bersamaan namun sebenarnya hanya dilakukan secara serialisme dengan waktu ganti yang cepat.



Contoh penerapan multithreading pada program ini dengan python stdlib threading

Ada di file client.py dan server.py

Mehami kedua source code itu memang membutuhkan waktu, maka dari itu alur kerja program diatas dapat dipersingkat dengan flowchart dan graph.

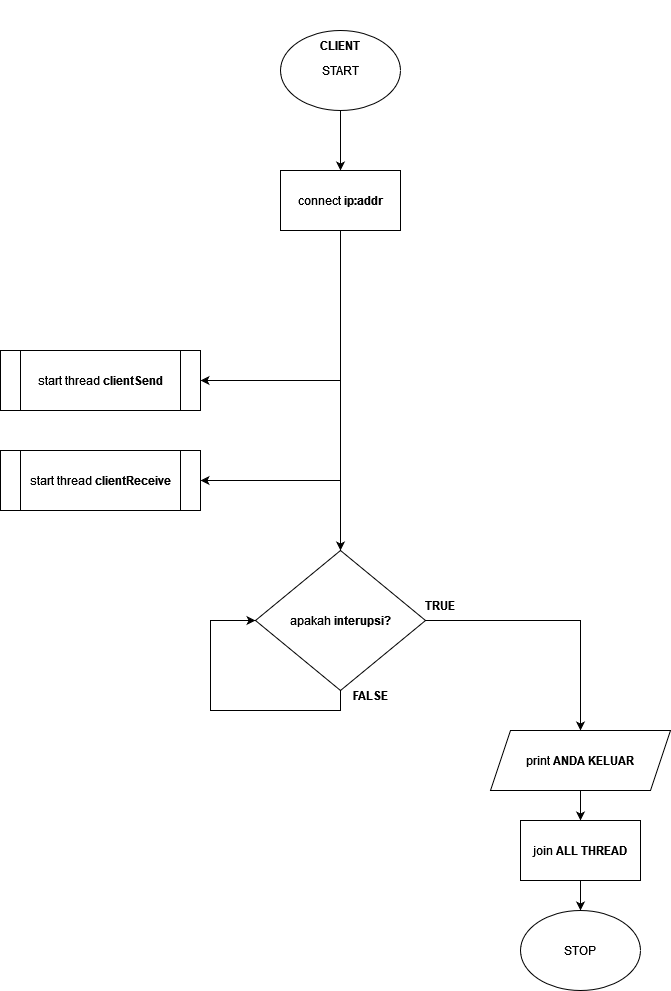


Ada 3 thread pada server.py yaitu

* serverAccept yang bertugas untuk menerima koneksi dari siapapun dalam jaringan.
* serverInput yang bertugas untuk menginput pesan oleh server administrator atau broadcast ke seluruh client.
* serverReceive yang bertugas menerima pesan dari semua client.

Program berjalan sampai variabel flag Bernama **interupsi** bernilai **TRUE.** Karena semua thread blocking nya bergantung pada variabel interupsi. Variabel interupsi di-trigger Ketika server administrator menekan CTRL+C **KeyboardInterrupt** atau Ketika input berisi nilai **!keluar**

Ketika interupsi bernilai true. Semua timeline pada threads menyatu Kembali ke timeline utama dikarenakan method **join**



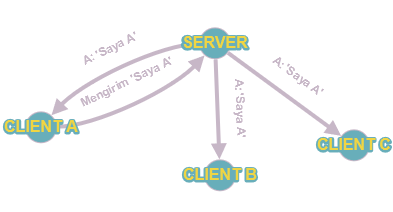
Ada 2 thread pada client.py yaitu

* clientSend yang bertugas untuk menerima input dari user dan mengirim pesan ke server
* clientReceive yang bertugas untuk menunggu data broadcast yang masuk dari server.

Program berjalan sampai variabel flag Bernama **interupsi** bernilai **TRUE.** Karena semua thread blocking nya bergantung pada variabel interupsi. Variabel interupsi di-trigger Ketika user client menekan CTRL+C **KeyboardInterrupt**  atau user input bernilai **!keluar**

Misalkan ada Server dan 3 Client sebut saja CLIENT A, CLIENT B, CLIENT C.

Ketika Client A mengirim pesan **‘Saya A’.** Maka graph ini menjelaskan apa yang terjadi

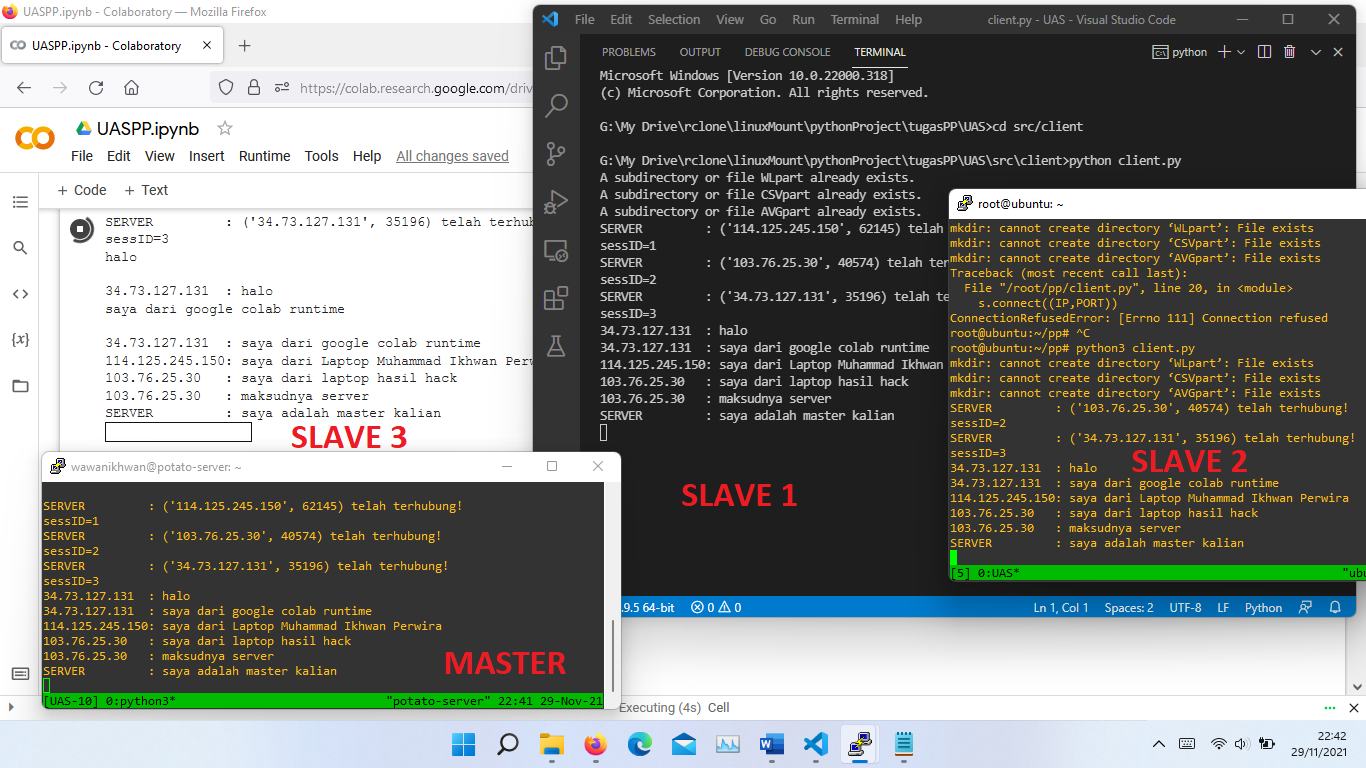


Server membroadcast pesan ‘**Saya A’** dari client A ke semua Client (termasuk client A itu sendiri).

Program ini menjadi dasar controller untuk master (server) yang mengendalikan slave (client)

Pada gambar di bawah ini dijelaskan bahwa master membuka socket dan berjalan pada port 12345 dengan protocol TCP yang berkomunikasi dengan slaves.

Gambar di bawah ini juga menjelaskan bagaimana slave dan master saling berkomunikasi.



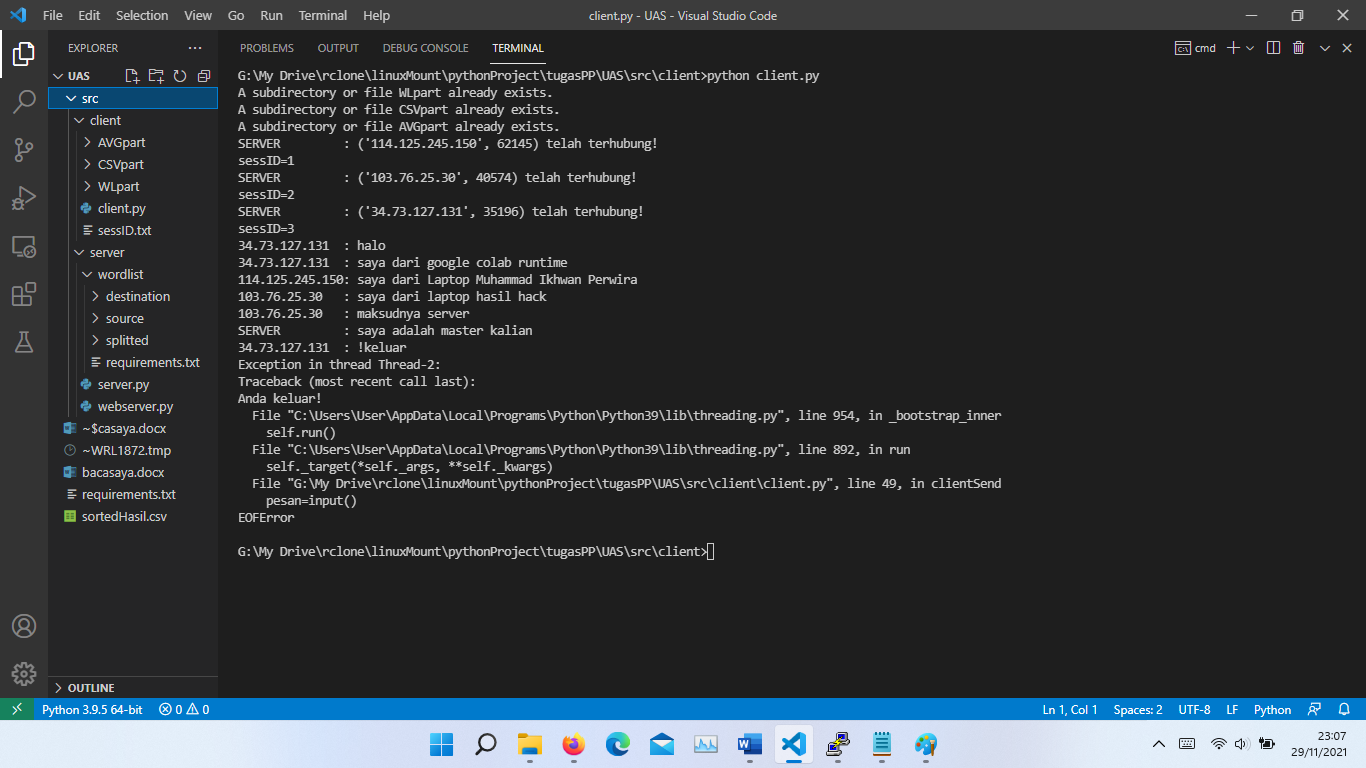
Cara mendapatkan nilai trends dengan menggunakan Google Trend secara resmi adalah satu-satunya dengan mengunjungi trends.google.com melalui web browser.

Cara yang kami gunakan tidak seperti itu melainkan menggunakan API Unofficial dari library python yang bernama **pytrends.** Adapun library lain yang dibutuhkan ada di file yang bernama **requirements.txt**

Sebelum lebih jauh bagaimana mendapatkan nilai trend dari API library ini saya akan memperkenalkan directory dan file proyek kami.

Source codenya terletak di directory paling atas yaitu **src**, di dalam **src** ada folder **client** dan **server.** Jika perangkat saat ini adalah slave maka focus ke folder client. Dan jika perangkat saat ini adalah master maka focus ke folder server.

Pada folder client, berisikan folder yang bernama WLpart untuk mendapatkan wordlist sebagian, AVGpart yang merupakan hasil wordlist Sebagian dan CSVpart yang berisikan file-file trend CSV dari tiap-tiap keyword dalam rentang 5 waktu terakhir. Serta file sessID.txt yng berisikan nilai sessID terakhir kali dari sebuah slave. Dan yang terakhir adalah script client.py yang berisikan semua proses yang terjadi.

Pada folder server, berisikan folder wordlist dimana folder wordlist terdiri dari 3 bagian yaitu source dimana wordlist original berasal, splitted dimana memecah file wordlist menjadi beberapa bagianm dan destination yang merupakan hasil akhir wordlist. Selain folder wordlist juga ada file yang Bernama webserver.py yang berisikan script webserver sederhana yang digunakan untuk kepetingan upload dan download antar perangkat yang berjalan di port 8000. Serta ada script server.py yang berfungsi untuk menjalankan proses server.

**Media Komunikasi Antara Master Dan Slave**

Sebelumnya sudah disebutkan bahwa media komunikasi socket TCP pada port 12345 digunakan untuk monitoring slave terhadap master. Sedangkan protocol HTTP pada port 8000 atau lebih tepatnya webserver digunakan untuk media download dan upload atau transfer file antar perangkat. Meskipun ada protocol yang simple untuk transfer antar file (FTP) Namun kami memilih HTTP Webserver karena dinilai lebih simple.

Socket TCP ada di script server.py sedangkan Webserver HTTP ada di file terpisah yaitu webserver.py.

Sedangkan untuk client TCP socket ada di script masing-masing Bernama client.py dan untuk client Webserver bisa gunakan Webbrowser atau Curl atau Wget.