Steps

[1] Fibonacci Server registers its hostname with Authoritative Server

[2] Authoritative Server creates a DNS record for the Fibonacci server.

[3] Authoritative Server a response back indicating the success or failure.

[4]User visit:

http://localhost:8080/fibonacci?hostname=fibonacci.com&fs\_port=9090&number=10&as\_ip=172.22.0.2&as\_port=53533

[5] User server parse the hostname from the query and query the DNS Authoritative server via DNS query

[6] Authoritative Server returns back the lP address of the Fibonacci HTTP Server

[7] User server request http://FIBONACCI SERVER IP/fibonacci?number=8

[8] Fibonacci Server returns the answer with 200 codes.

[9] User server returns the result to the user

Test:

cd C:\Users\wotno\PycharmProjects\pythonProject\dns\_app

docker build -t bulutmf/fs:latest .

docker build -t bulutmf/as:latest .

docker build -t bulutmf/us:latest .

docker network create nt

docker run --network nt --name as -p 53533:53533/udp -d bulutmf/as:latest

docker run --network nt --name us -p 8080:8080 -d bulutmf/us:latest

docker run --network nt --name fs -p 9090:9090 -d bulutmf/fs:latest

docker ps -a

docker network inspect nt

**test FS** - http://localhost:9090/fibonacci?number=10

docker inspect --format='{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' as

**test US -**

http://localhost:8080/fibonacci?hostname=fibonacci.com&fs\_port=9090&number=10&as\_ip=172.22.0.2&as\_port=53533

Get-NetTCPConnection | Where-Object { $\_.LocalPort -eq 53533 }

FS\_ip = 127.22.0.4

FS port 9090

US\_ip = 127.22.0.3

US port 8080

AS\_ip = 127.22.0.2

AS port 53533