

## Project 2 – Ambulance service – People

### People – Ambulance service:

This communication will be a 2-way communication. The users will send request to the ambulance service. Let's say I'm user A, I have been tested recently positive to the virus in the nearest lab near my town and I feeling bad so I ask for an ambulance:

- I send a request for assistance to the ambulance service (ambulance server)
- The ambulance service server acknowledge that they have received my request, I have to wait
- The worker at the ambulance service starts to treat my request
- Once they started to treat my request, I send my users data (name, phone number, lab I have been tested in, number of user in the lab, ...)
- The ambulance service automatically checks my file (does the lab files correspond? Have the user ever been to a lab?)
- The person working at the service ends what they are doing (other phone calls), open my file and call me. She already know if I'm positive or not, for how long, if I have other problems... and the ambulance can almost start to leave to come rescue me after the worker checked that I am really in need (if it is necessary according to my profile)

### FILE

1. There is a directory for the Ambulance service server: Ambulance\_service.

In this directory, you will find the file with the record of the lab in the directory Record (the content of this file is the same as for the lab-Ambulance\_service\_communication).

You will also find there the file of the A.S. server: AS\_server.py

And finally you should see the file: Data\_Current\_User.txt . It is the file used by the ambulance service to read the personal data of the client.

2. There is a directory for the Ambulance service server: ClientX.

In this directory, you will find the file with the personal data of the user: me.txt.

```
Etienne000
Age: 24ans
Diseases: No known problem
Phone number = 01.02.03.04.05
```

You will also find the file of the client server: User\_server.py

### TEST

If you want to test my servers:

Make the AS server run ("ready to receive").

Make the client server run (user\_server.py) one or more (in client and/ client2 and/or client3).

Go in the Ambulance\_service directory. There, you should have a file looking like:

('192.168.47.1', 59864)\_Request\_state.txt

Wait 30 seconds and you will see the file: ('192.168.47.1', 59864)\_1.txt appear. The last number will increase as long as you wait(increments every 30 seconds).

Open the 'Request\_state' file and replace the 'No' by 'Yes' or anything with a 'y' in it.

Save the file and close it.

The file : ('192.168.47.1', 59864)\_1.txt should disappear.

In the terminal for the AS\_server, answer the question 'All associated files closed? :' by just typing anything.

If you used:

- Client3, you should see : 'This user hasnt been in a lab', then Data\_current\_user.txt is the same as the me.txt file. This case corresponds to a user that has never been in a lab before.
- Client2, you should see : 'This user lab file hasnt been found', then Data\_current\_user.txt is the same as the me.txt file. This case corresponds to a user that has a corrupted me file or lab file. Then we use the me file because we couldn't find the lab file.
- Client, you should see : 'Lab data found and now used' hen Data\_current\_user.txt is the same as the personX\_Y.txt file of the lab records. This case corresponds to a user that has been in a lab where we could find the associated file.