

CovidHub official documentary

This is a repository for mini-projects about COVID-19: EpidemLab, Corbot and Cormap on website, which

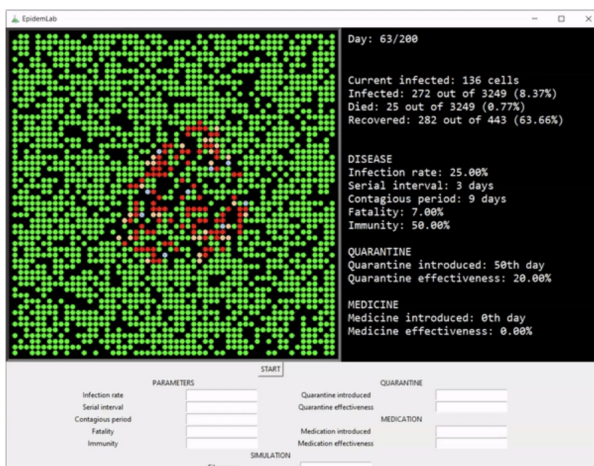
- simulate the spread of the disease with your parameters
- keep you updated with the daily coronavirus situation in a country you choose, build comparison plots
- display the current coronavirus situation around the globe

respectively. It was created as a semester work from the programming basics course of the two students of Ukrainian Catholic University.

EpidemLab

This application simulates a disease spread with you parameters. It is simple yet versatile and effectively shows and visualizes the dependency of the spread of a disease on its parameters and mitigation measures. The idea was developed thanks to the article from <https://towardsdatascience.com>.

With quarantine



No quarantine



Usage

It is recommended to launch the simulation with default parameters on the first time, without entering anything. In summary, there are 13 parameters, 10 of which you can enter by yourself.

DISEASE:

- `Infection rate`: float [0, 1] - chance of infected cell infecting its neighbour
- `Serial interval`: int [0, inf) - how fast the disease spreads
- `Contagious period`: int [0, inf) - how long the cell is contagious
- `Fatality`: float [0, 1] - the chance of infected cell dying when the contagious period ends
- `Immunity`: float [0, 1] - the chance of healthy cell not being reinfected

QUARANTINE:

- `Quarantine introduced`: int [0, inf) - the day on which the quarantine is introduced
- `Quarantine effectiveness`: float [0, 1] - the chance of infected cell being found and put on quarantine forever

MEDICATION:

- `Medication invented`: int [0, inf) - the day on which the medicine is invented
- `Medication effectiveness`: float [0, 1] - the chance of infected cell recovering after taking the medicine

SIMULATION:

- `Filename`: str - the file's name in which the results of the simulation will be saved
- `Number of days`: int 200 - how long the simulation lasts (can't be entered)
- `Width`: int 57 - the number of cells on each side of the grid (can't be entered)
- `Population density`: int 0.7 - the percentage of board covered with cells (can't be entered)

The last three parameters cannot be entered by the user, because they proved themselves to be optimal and there's little sense in changing them. But if you want, you can do it by editing the code.

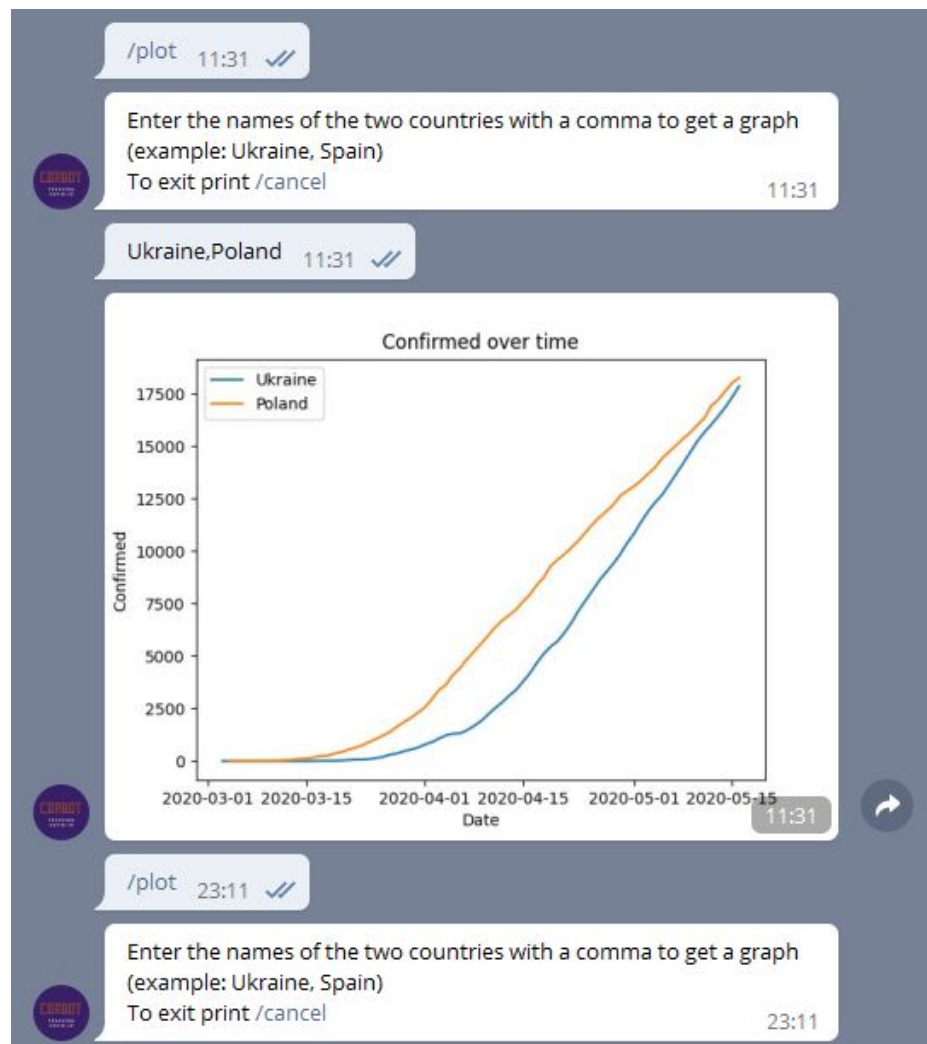
Corbot

Corbot is a bot based in the telegram application. It will help you learn the latest information about the coronavirus of selected countries. You will be able to build a graph comparing the two countries in terms of the number of patients. And, most importantly, subscribe to the daily newsletter, where you can add and remove from the list, countries on which you follow. Every day you will receive a message with general statistics on the number of deaths, illnesses and people cured.

The program has several commands. We'll tell you about each of them now.

First command:

`/plot` - comparative statistics of the two countries on the number of new patients. To better understand this command, review this image:



Now we'll talk about the subscription function and related commands. To use it, enter the `/reference` command

`/reference` - receiving a daily subscription to information about coronavirus.

After entering the command, you choose which country you want to subscribe to. From now on, every day you will receive information about the number of sick, dead and recovered.

To better customize the subscription feature, a series of commands have been added to control receipt. That is, add another country to the subscription list, delete one of the existing countries, unsubscribe. In the future, it is planned to add new features, such as clock selection.

To better understand subscription, We summarize the list of commands:

`/del_reference` - to stop using subscription.

`/add_country` - adds country to your subscription.

`/del_country` - deletes country from your subscription.

Also, a few more commands have been added. For example, to obtain general information about the country (sick, dead, recovered), to receive separate information, only about one thing (sick, dead or recovered). Now here is a list of these commands:

`/deaths` - the number of deaths from the coronavirus in the selected country at the moment.

`/confirmed` - the number of confirmed patients from the coronavirus in the selected country at the moment.

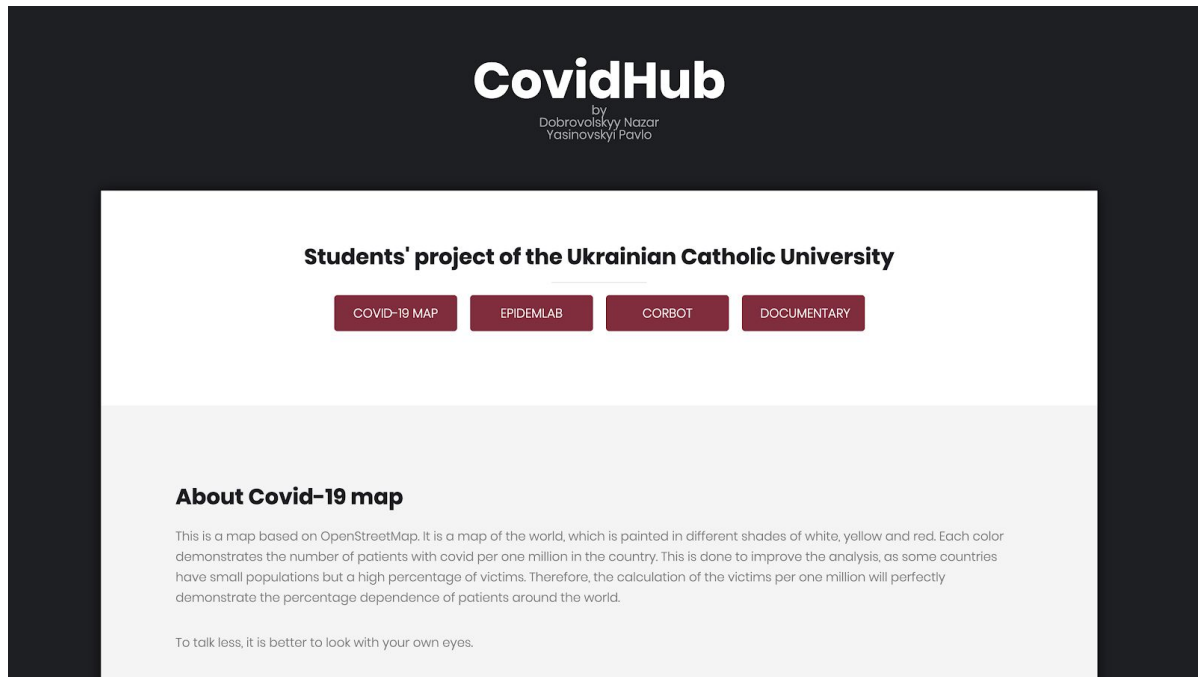
`/recovered` - the number of recovered people from the coronavirus in the selected country at the moment.

`/country_total` - displays general statistics of the country at the moment.

But to understand all these actions, commands, We recommend trying this bot on your own, just follow the link https://t.me/corbot_bot

Website with map

Link: <http://covidhub.pythonanywhere.com/>

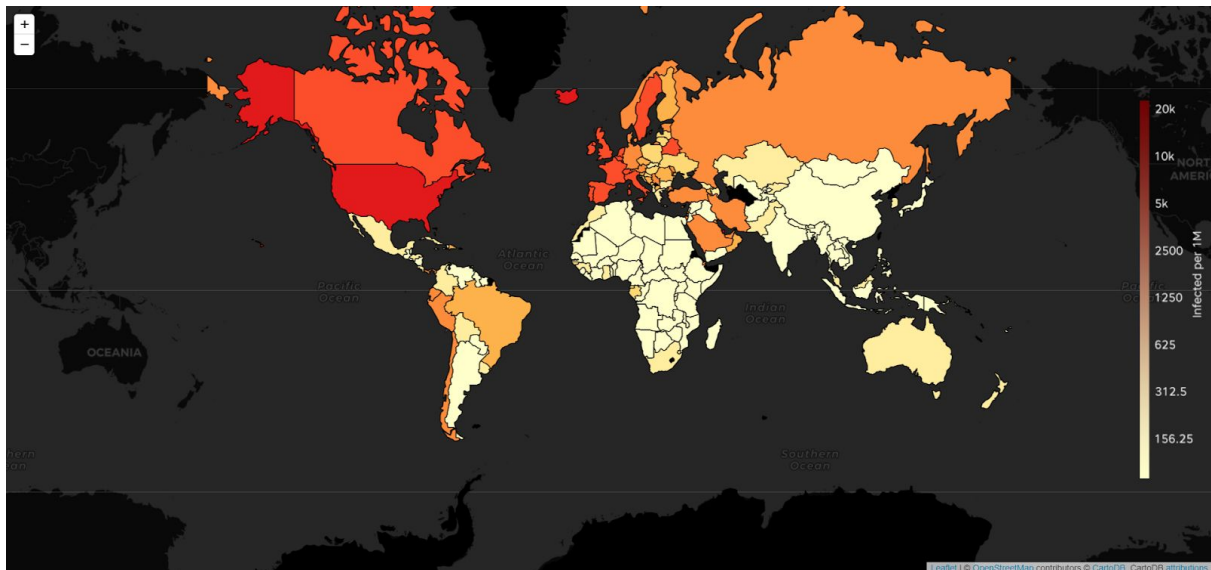


This part is a website that connects all parts of our project, ie telegram bot, simulation application, world map and documentation.

There you will get access to four implementations. They are highlighted with red buttons.

The first button - COVID-MAP - takes you to a site with a map of the world, which shows each country painted in shades of white, yellow and red. Each color indicates the number of coronavirus patients per million population. This is a simple choropleth map that displays the number of infected per one million. It is created via folium and uses the Country ADT which stores all the needed information about the country it represents.

Example of this map:



The second button - EpidemLab - download with the same name application, which simulates coronavirus infection, which we have already described above.

The third button - CorBot - is a link to the implemented telegram bot, which we also have already described above.

And the fourth button - Documentary - a link to the documentation, which contains general information about the project, its methods and brief instructions for use.