1. Install Python. Everything below worked for Python 3.12.8 and pip 24.3.1 . Anything else: צה"ל לא אחראי
2. Create a virtual environment. In a Windows shell (“cmd”) do the following:

// Create the virtual environment in folder gg\_venv

python -m venv gg\_venv

cd gg\_venv

// Activate the virtual environment

Scripts\activate.bat

// you should now see “(gg\_venv)” at the left side of the command prompt.

// Get the application from github (I am assuming you have git installed on your machine. If not, install it first)

git clone https://github.com/mmstahl/gevergiver.git

// install the requirements:

pip install -r requirements.txt

// Run the application:

python app.py

1. The application is accessible from:

<http://127.0.0.1:5000>

and from the IP that you get from the WiFi access point, if you are on WiFi, also on port 5000. E.g.

<http://10.0.0.1:5000>

1. To make the app available for others, via the internet, you need to install it on a web server. I used AWS.

* Created an **instance** via EC2
  + Pay special attention to the inbound / outbound network rules. I just enabled all IPv4 traffic, for HTTP, HTTPS, and Custom TCP. May need also to do something specific for port 5000 on Custom TCP. I don’t remember…
  + I used the AWS Linux, and I think it also worked with an instance with Ubuntu. Since this is Python, any Linux flavor should work.
* Started the instance it and opened a console to it
* Installed Python 3.12 and pip (check Google or your favorite AI how to do this)
* Ran the commands on step 2 above

The application is reachable via the public IP of the instance, at port 5000.

NOTES:

* AWS changes the public IP each time you start the instance. Since they charge for the time the instance is running, you will probably want to stop the instance when not in use, so the IP will change
* To keep the same IP you need to use a domain you own. Can be on AWS (through their Route 53 service; this is what I used) or via some other provider
  + Need to install load balancer and a certificate. Ask Copilot of some other AI for step by step guidance
  + This way you can also make the URL an HTTPS and not HTTP, which is better (not that I am too concerned about anyone hacking the GeverGiver…)