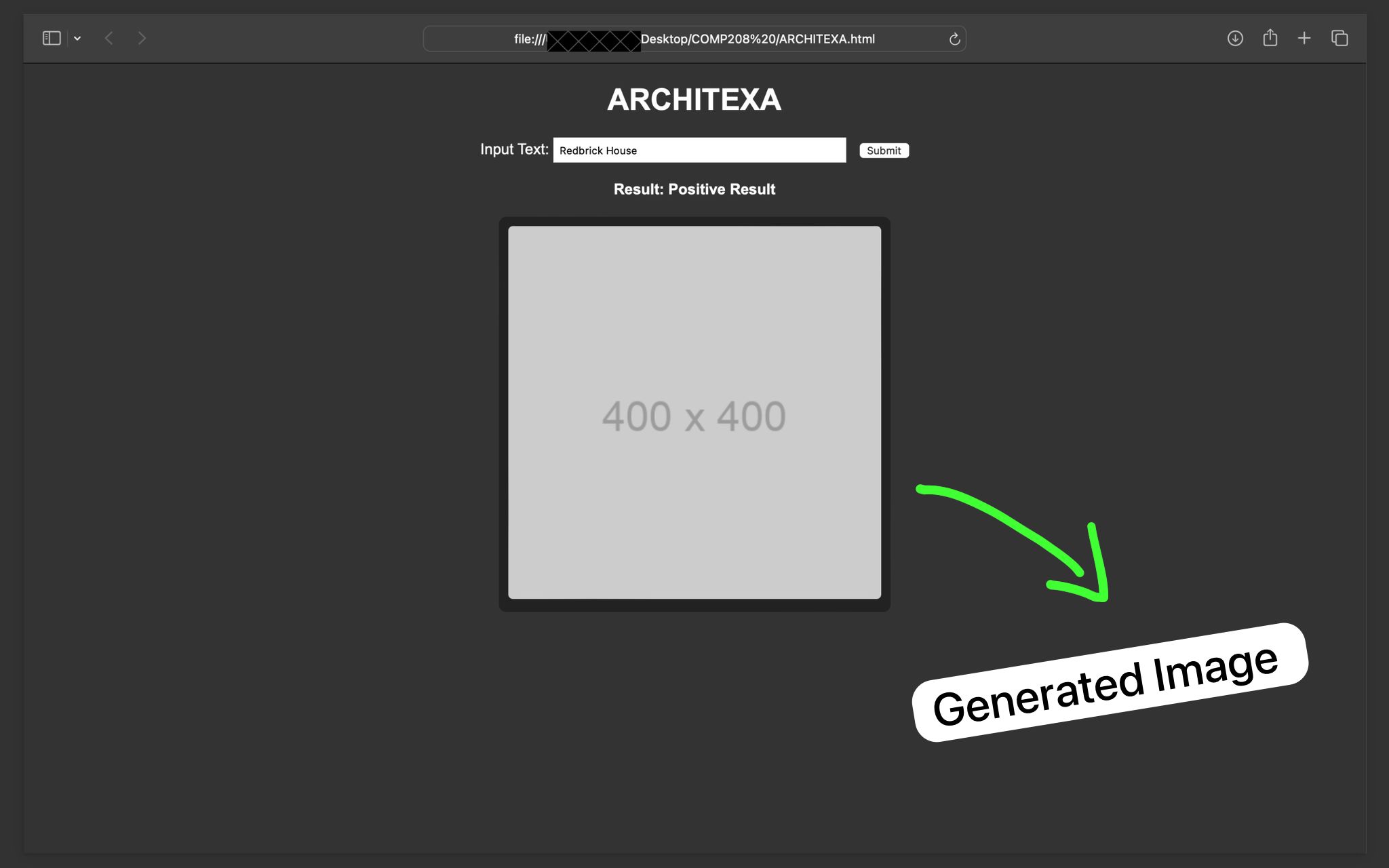
Architexa users manual:

Architexa is a CGAN neural network capable of generating images of buildings based on your prompts.

Common users:

To use Architexa navigate to this website: (URL)

Where you will see this:



(our website also has an about us section if you are interested)

From here you will need to input a short sentence broadly describing the image you want generated and then click submit.

After allowing 75 seconds for generation the UI will then deliver four generated images matching your prompt.

Case tool users:

Our neural network optimisation tool, and our dataset building tools can be found in the following repository:

[Ash237333/208-Project (github.com)](https://github.com/Ash237333/208-Project/tree/main)

Once there you will need to download either CGAN.py (from the multi-prompt-branch folder), for the optimisation tool

Or the contents of data\_preprocessing/generated\_images/code for the database building tools

In the network optimiser case you will then need to install python IDLE 3.11, and the tensorflow library for python:

[Python Release Python 3.12.2 | Python.org](https://www.python.org/downloads/release/python-3122/)

[tensorflow · PyPI](https://pypi.org/project/tensorflow/)

In the database builder case you will need:

* Google chrome.
* A discord account.
* A subscription to mid-journey.

Running the network optimiser:

For the network optimisation tool you will the want to attach your CGAN build to the tool by storing your generator and discriminator in separate files called generator and discriminator, and ensuring that their build functions are named the same.

Then feed the script some best guess starting values and wait for it to generate test suites. Then create as many separate instances of the script as your computer can run, and feed each of them one of the generated test suites, after about a day each of them will terminate with images generated from each test suite value. Select the best images out of all these images and feed that back into the original (or master) script which will then produce new test suites.

After each iteration of this, your neural networks should perform better, and eventually the hyperparameter values will converge (you will know they are completely optimized when a new test suite consists only of the same values you put into it). However you may not have time to run to this point, but wait for at least some significant convergence if you want to see significant improvement in accuracy.

Running the database builder:

After having downloaded the contents of data\_preprocessing/generated\_images/code open your computers command prompt and run the following:

export PATH=$PATH:$(go env GOPATH)/bin

Then change the directory for command prompt to the location where the .yaml files from the download are stored (this can just be /downloads)

Then run the following:

bulkai create-session

This will open a chrome tab at discord, and you will need to login to your discord account (do it via the pop up in that windows or it can cause problems)

Finally declare a command in discord then type and submit the following:

bulkai generate --config bulkai.yam

(you can stop the generation process by pressing cntrl+c)