***\*\*brief writeup of what you were able to get work and what could be done if you had more time or resources\*\****

**DeepDesign LoGAN**

LoGan was designed to bring the power of GANs to the public. Our technology allowed users to generate images on the fly with only a small amount of time spent waiting. The pipeline consists of several steps to bring the generated image to the user.

The pipeline starts with the heart of our technology, our model. Using the StyleGan2-ada pytorch implementation found [here](https://github.com/NVlabs/stylegan2-ada-pytorch) over the course of a few weeks we were able to train our model to produce as high quality images as possible. With the use of an RTX 3080 and around four days of training time we were able to produce favorable results. After training the model a user was able to generate an image via a seed number. This number could have ranged up to a value of (2x10^9 -1) allowing for a broad range of images to be generated.

Due to resource constraints the images generated were 256x256. The next part of the pipeline takes the image generated by our model and 4x the size of the image using AI supersampling. The model ESDR was used and implemented via a module in OpenCV, a popular computer vision library in python.

With the image generated and upscaled the image is then displayed to the user through our front end UI which was created using Flask a useful library in python that is used as a web microframework. After the image was generated for the user it was saved in our database for later use.

With no real experiences developing user interfaces within our group besides the CTO’s coding bootcamp experience. We were building a bridge while also learning how to do it at the same time. Given more time the user interface could have been cleaned up to provide a more user friendly interface. This would have included giving users the ability to input parameters to affect their generated images as well as the option to register their image and remove the generated images seed from production. This would allow the replication of that image to be impossible within that given model.

Given unlimited computer resources, training images natively at 1024x1024 or larger would have been an ideal solution for the team and eliminate the need for AI upscaling. With eight top of the line enterprise gpus we would have been able to train a much larger model in just a few days to a week. With unlimited resources as stated above we would have been able to provide a higher quality image in a fraction of the time.