

## Neural networks:

Test:	Data:	Outcome:
Neural network can successfully learn some features (functional)	On feeding the networks our dataset we produced images which showed a stone corner being extracted, then placed as part of an archway.	Passed
Neural network can learn to differentiate distinct classes (functional)	On feeding the network our dataset,, we could generate consistently distinct images for inputs conditioned on different classes.	Passed
Neural network can produce a human-recognisable image in response to a simple one class distribution. (functional)	During the initial testing of the neural networks,, we generated appropriate images (from a database of images of fists) using the networks and a pre-built training loop.	Passed
Neural network can produce a human-recognisable image in response to a simple multi-class distribution (functional)	During the initial testing of the neural networks we generated appropriate images of all three classes in the rock-paper-scissors dataset. However, in the final epoch, some were mis-labeled by the discriminator.	Passed
Neural network can generate an image in less than 60 seconds (acceptance)	On an 8-core computer with a clock speed of 5.6Ghz and 8Gb of RAM, a trained version of the networks was able to produce an image in 42.8 seconds.	Passed
Neural network can be trained in less than 2 days (acceptance)	On an 8-core computer with a clock speed of 5.6Ghz and 8Gb of RAM, the networks could be trained (using a pre-written training loop) for 200 epochs in under 2 days.	Passed
Neural network can be altered to accept images of size 256*256 and still function (acceptance)	On a 12-core computer with a clock speed of 4.8Ghz and 16Gb of RAM, the networks were able to be trained for 10 epochs with an image size of 256*256	Passed
Neural network can be trained for 1 epoch without crashing an average computer (acceptance)	This was possible on a Microsoft Surface pro 2.	Passed
Loss values for the discriminator and generator converge when trained. (functional)	During multiple training instances for the model on our dataset, the discriminator's accuracy function varied around 0.5, and the generator's accuracy trended up to 0.75.	Passed

## Dataset:

Test:	Data:	Outcome:
All images are used legally and without civil liability. (acceptance)	The dataset is constructed from images generated by the mid-journey AI, which gives the image generator full licenses.	Passed
Each category combination has at least 100 images in it. (functional)	All 8 categories have at least 100 images.	Passed

A random sample of AI-generated images (of size 100) contains only images humans can correctly categorise. (functional)	I manually collected 100 random images and asked a friend to attempt to categorise them, who was able to do so.	Passed
Dataset can be transmitted in less than 10 minutes over university wi-fi. (acceptance)	The entire dataset (when compressed) can be moved over university wi-fi in 5 minutes 24 seconds.	Passed
A random sample of 100 images from the dataset are all the same size. (functional)	All images from the dataset are the same size, as they were generated at 128*128 by the midjourney AI.	Passed

## Optimisation tool:

Test:	Data:	Outcome:
Optimisation tools can save an example neural network in a partially trained state. (functional)	During all training loops run for the optimiser, the model's weights were saved at every epoch.	Passed
The optimisation tool can run simultaneously on two physically distinct computers. (functional)	At many times during optimisation, three trials were running simultaneously.	Passed
Optimisation tool can optimise the learning rate for an example neural network in less than 1 week. (acceptance)	Optimisation of our neural networks' learning rates, enough to move from static to as accurate as they could be (accounting for mode collapse) took 6 days.	Passed
The optimisation tool can use feedback from previous tests to automatically generate new test values for hyperparameters. (functional)	Our optimiser requires a human to select whether the output from a test is better or worse than previously to calculate new potential values.	Failed

## Website:

Test:		Outcome:
The website can be accessed via Google Chrome, Microsoft edge, and Mozilla firefox. (acceptance)	The website can only be accessed via Microsoft Edge.	Failed
Websites can load in less than 20 seconds on wi-fi throttled down to 1Mbps, 10 for 5Mbps, and 1 for 10Mbps. (acceptance)	The website is a thin client, and can easily load within these time frames; testing of this was done using throttling tools.	Passed
All pages of the website can be accessed via the taskbar in all other pages of the website. (functional)	All pages contain a taskbar, which can be used to navigate to all other pages. (this has been exhaustively tested)	Passed

Website prompt field can accept textual input. (functional)	The field will accept input of alphabetic, numeric, special, and arcane Unicode characters.	Passed
Website API calls can be received by dummy script. (functional)	Website can be used to perform a Google image search via a dummy API call.	Passed
Websites can receive (and then display in browser) images from a dummy script. (functional)	Websites can return images resulting from a google image search, via a dummy API call.	Passed
Creating a username and password, then entering usernames and passwords other than those created, prevents one from entering the website, whereas entering the correct one allows one entrance. (functional)	This was extensively tested with a uni email address as username, and testing as password. Similar password and usernames were used, as well as random strings, and an sql injection.	Passed

## Integration testing:

Test:	Data:	Outcome:
Neural Network can be trained via Optimization tool in less than 2 days. (acceptance)	Optimisation tools training loop is functional and can train the networks for 200 epochs in 2 days.	Passed
Neural Network can be saved in a partially trained state via the optimisation tool. (functional)	The same tests that passed this in unit testing pass this.	Passed
Neural Network can be trained simultaneously on multiple computers, to optimize. (acceptance)	This is functionally true, however resources were limited and we were only able to run 3 trials at once.	Passed
Learning rate of the Neural Network can be optimized by neural network in less than 4 days. (acceptance)	The optimisation of learning rate seems to take fairly consistent time regardless of data complexity, and thus optimisation took only 4 days.	Passed
Neural Network can be trained on our dataset. (functional)	The neural networks correctly extracted and combined simple features from the dataset.	Passed
Neural network can produce human-recognisable images in response to any one class in the dataset. (functional)	When trained on a single class from the dataset, recognisable images were produced by the model.	Passed
Neural Network can produce human-recognisable images in response to all combination categories in the dataset. (functional)	When made to learn the distribution of all 9 classes, the model suffered slightly from mode collapse after approximately 150 epochs, however recognisable images were still generated.	Passed
Website can call the Neural Network without triggering the training apparatus. (functional)	Website was called thirty times with different valid prompts, and	Passed

	appropriate images were passed back each time.	
Neural Network can receive prompts from Website. (functional)	Neural network was seen though the console to begin generation upon receiving external input from the website.	Passed
Website can receive (and then display) images generated by Neural Network. (functional)	Website was called thirty times with different valid prompts, and appropriate images were passed back each time.	Passed
Neural networks can produce human recognisable images in response to a stripped version of the true dataset containing only brick houses. (functional)	When the complexity of our dataset was reduced, and we increased the number of layers in the network, we were able to achieve this without encountering mode collapse.	Passed

## Proof of met requirements:

The original system objectives were as follows:

Aim 1:	Underlying GAN can generate random, human-recognisable images without prompts.	Objective 1: Images thus generated are correctly classified by pre-existing classifiers as buildings 70% of the time. Objective 2: The model can produce images in less than 15 seconds.	(primary aim)
Aim 2:	GAN is successfully updated into a CGAN and can generate recognisable images of specific building types in response to single-word prompts.	Objective 1: images thus generated are correctly classified by pre-existing classifiers as their building type 70% of the time. Objective 2: The model can produce images in less than 20 seconds.	(primary aim)
Aim 3:	CGAN can be called through a web-based API.	Objective 1: The website can be accessed via Microsoft Edge and Google Chrome. Objective 2: a remotely stored version of the model can be tasked through the website.	(secondary aim)
Aim 4:	CGAN has optimised hyperparameters and dataset.	Objective 1: images thus generated are correctly classified by pre-existing classifiers as their building type 90% of the time. Objective 2: The model is re-trained in less than 8 minutes.	(secondary aim)
Aim 5:	CGAN gives four options in response to each prompt.	Objective 1: all four images are correctly classified. Objective 2: human users report that images are diverse.	(secondary aim)
Aim 6:	CGAN can respond to multi-word prompts	Objective 1: images may be generated in response to 3-word prompts. Objective 2: images thus generated are correctly labeled by existing classifiers 70% of the time.	(tertiary aim)
Aim 7:	database of login information.	Objective 1: the user may create accounts through the website. Objective 2: users may log back into their accounts using stored login credentials.	(tertiary aim)

Over the course of the project, we also added the aim of producing a proprietary dataset.

## Proof of met requirements:

We have largely completed the first aim, as the model is capable of learning a data distribution and producing output images based on it. However, the complexity of the data is such that those images sometimes lack accuracy.

We have also completed the second aim, as the models are capable of being conditioned on labels, and will produce consistent distinct outputs for different prompts, however these images once again may not always be accurate.

We have completed the third aim, as the website we have hosted on the university servers is capable of calling the model.

We have completed the fourth aim through the use of the optimisation case tool, and this can be seen through the difference in quality between images produced by a naively trained version of the model and an optimized one:

The fifth aim turned out to be trivial, since it only involves putting a for loop in the code to generate four images every time the generation subroutine is called, however we have completed it.

The sixth aim is functionality completed, even though it was actually the dataset that had to be updated in order to make this possible, and technically the aim should read “model can respond to multi-label prompts”

The seventh aim is completed, as can be seen when using our website, it is necessary to make a username and password, and then enter a correct username and password before utilizing the site.

Our new aim of creating a proprietary dataset is also completed, as we have 5,000 images of our 8 classes stored in our GitHub repository.

(screenshot of subdirectory and a subset of image files in the subdirectory)