

# PHOTOREALISTIC IMAGE SYNTHESIS FROM TEXT

## DESCRIPTION USING MACHINE LEARNING

Project batch members:

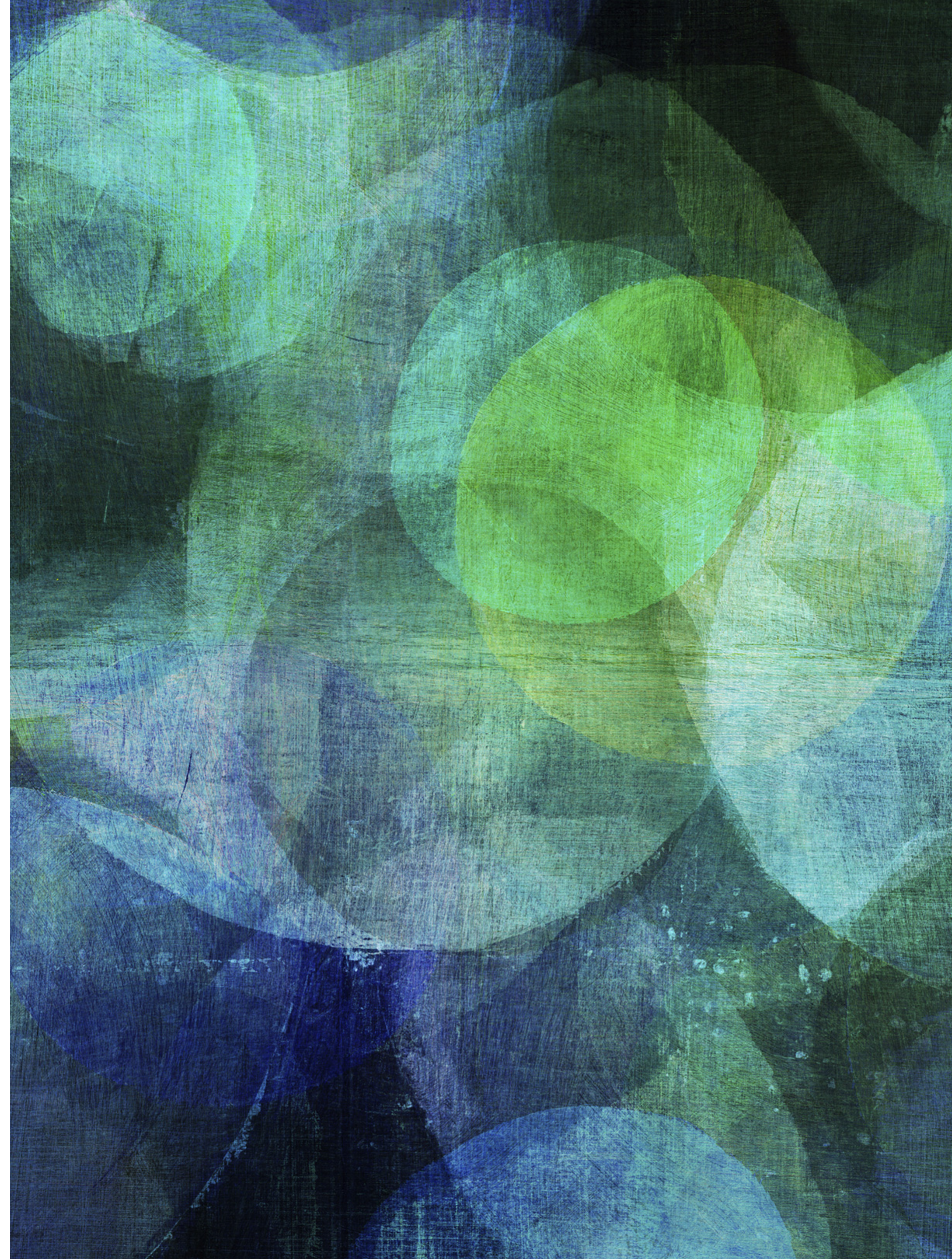
N. M. Hariharan (15DC05)

R. Jeevananthan (15DC06)

S. Praneet (15DC17)

P. Yathinder (15DC27)

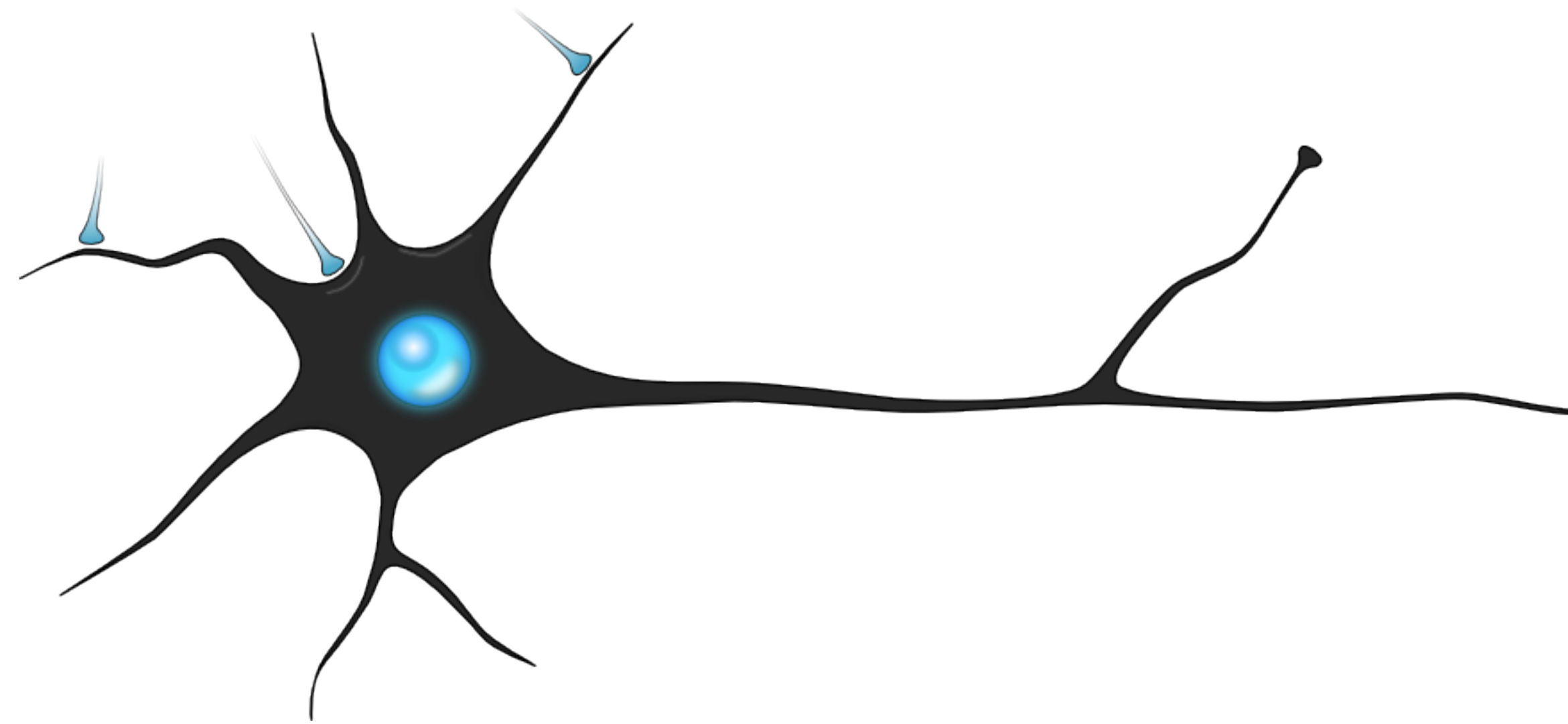
Project Guide:  
Mrs. T. P. Kamatchi





# PROBLEM DEFINITION

Photorealistic image synthesis has been difficult till date with the existing technologies however Artificial Intelligence makes this possible, we can benefit a lot from the wide application of this emerging technology. It can be employed to replace human labors in completing many tedious tasks. The objective of this project is to generate photorealistic images from a text description given by a user of a specific thing, object or being using machine learning. This project can be used by designers, engineers to generate designs based on concepts that they've thought.







# OBJECTIVES OF THIS PROJECT

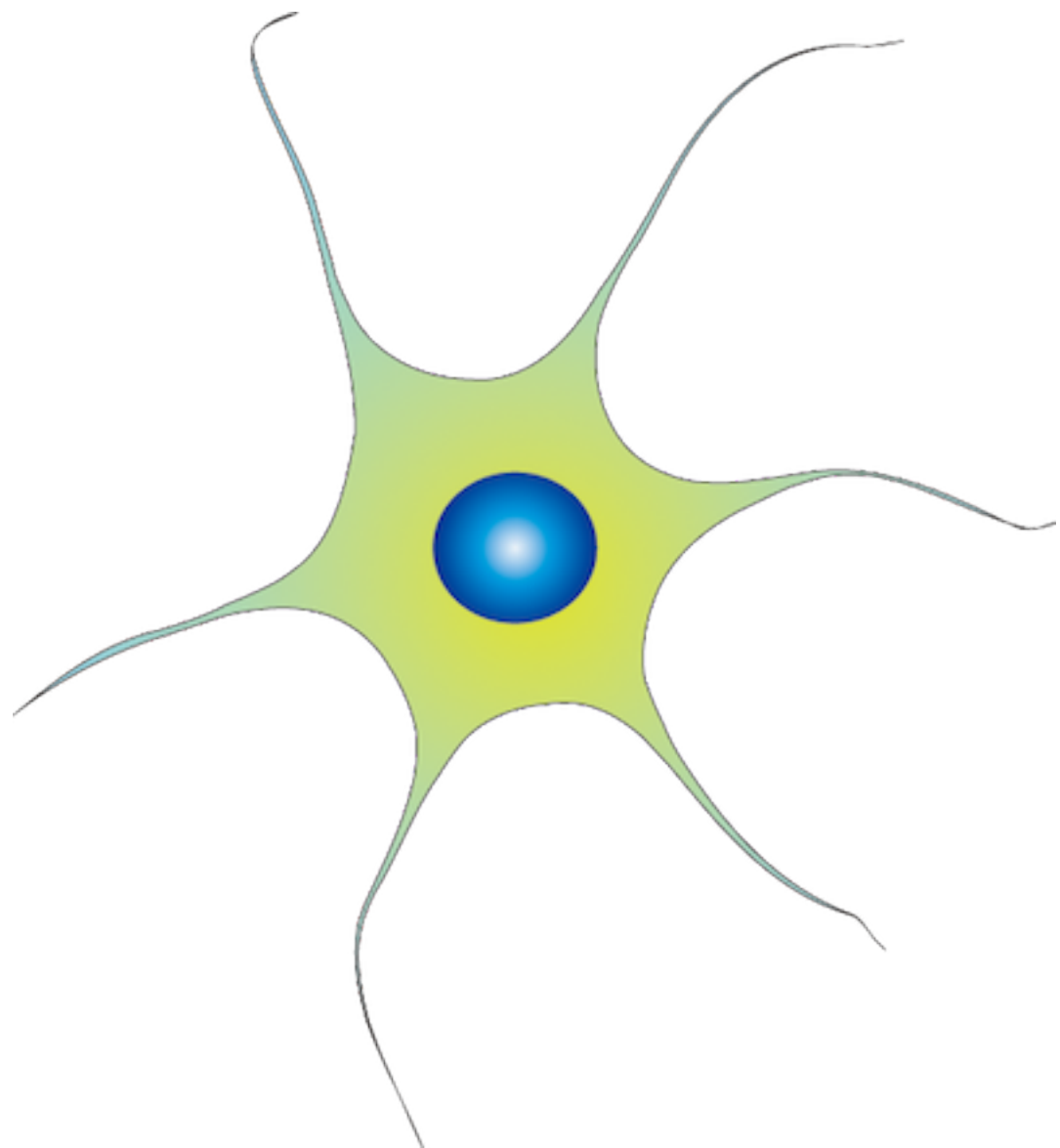
---

- To generate realistic images based of concepts and ideas based of the images used to train the model
- To make the machine learning model generic, adaptable to be trained on different datasets.



# OVERVIEW OF THE MODEL

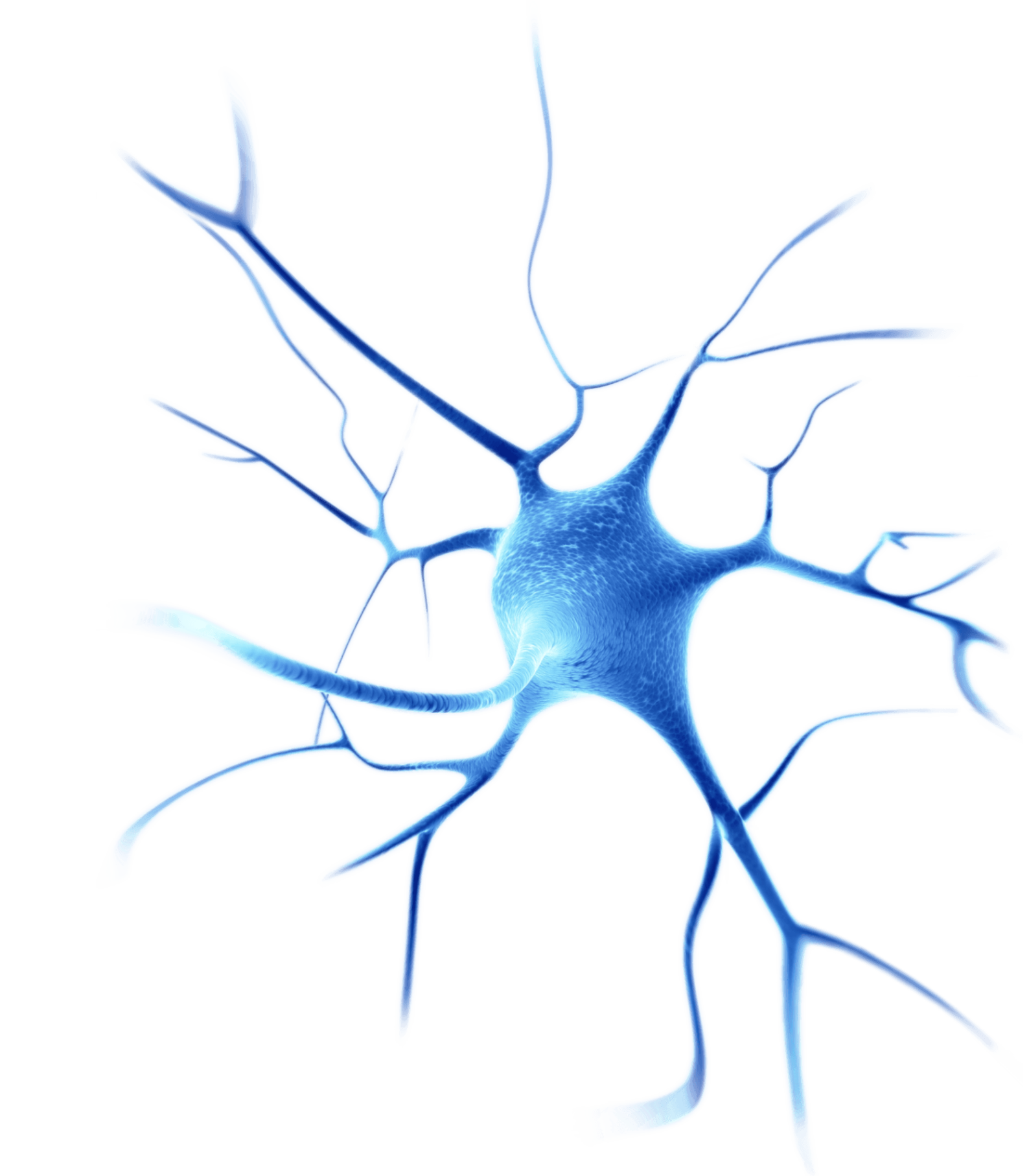
**We propose a system using a Generative Adversarial Network (GAN) which is a machine learning model which consists of a Generator and a Discriminator both of which are trained with the same dataset, the Generator is used to generate fake images based of the real images from the dataset that's used to train it. The Discriminator classifies the generated image as fake or real, when the Generator generates an image so compelling and realistic the Discriminator classifies it as real.**





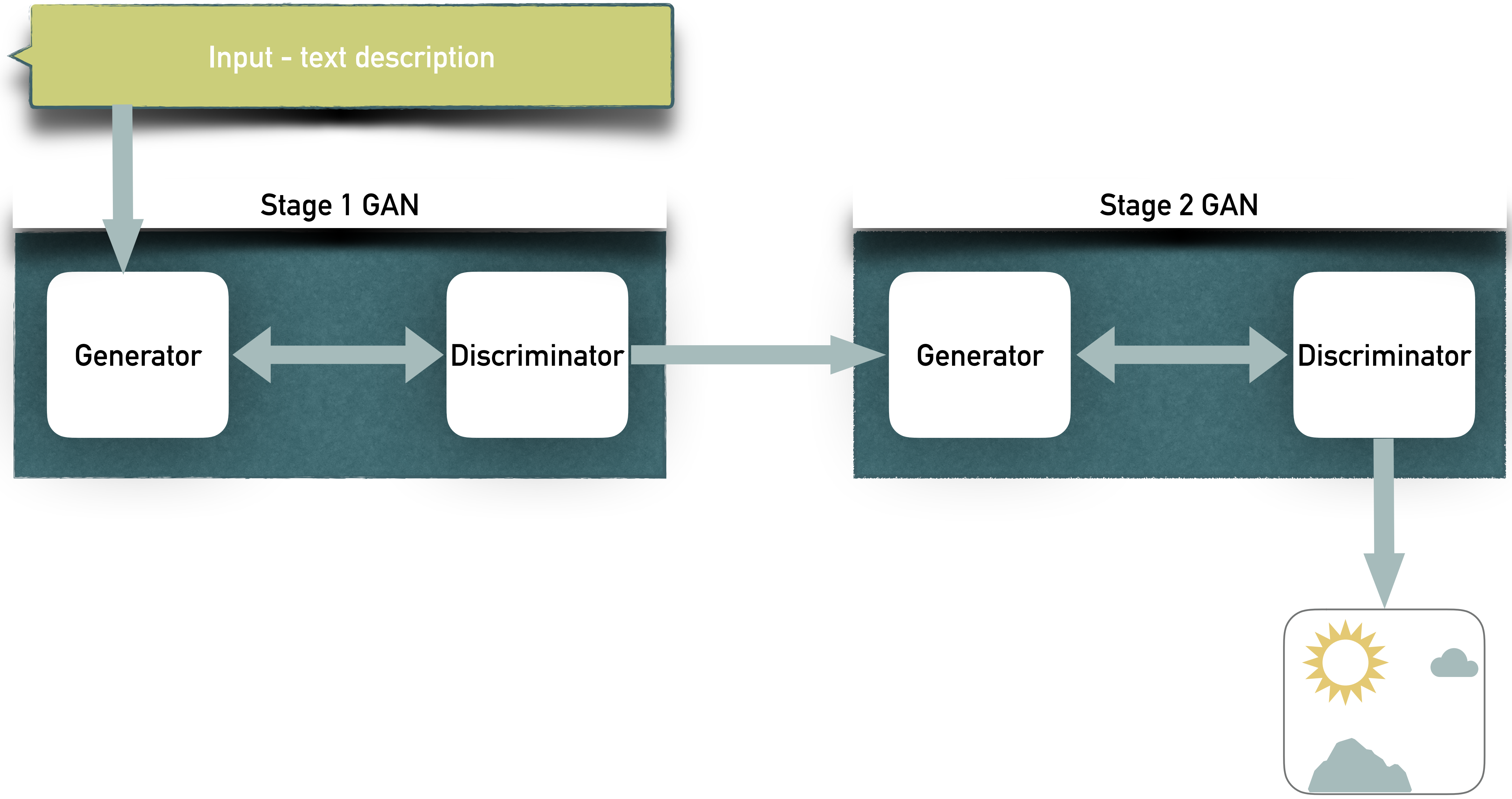
# OVERVIEW

**This project is built using a stack GAN which consists of two GANs, the first GAN is called a stage 1 GAN which takes the sentences which are represented as word vectors and generates an image with primitive shapes and basic colors, it is a low resolution image, the second GAN is a stage 2 GAN which takes the image generated by the stage 1 GAN and the original text description as the input and generates a much higher resolution version of the image by completing the details.**





# Stack Gan Workflow diagram





# DEPENDENCIES FOR THE PROJECT

---

## Software dependencies

- Python programming language
- Tensorflow - Open source machine learning library

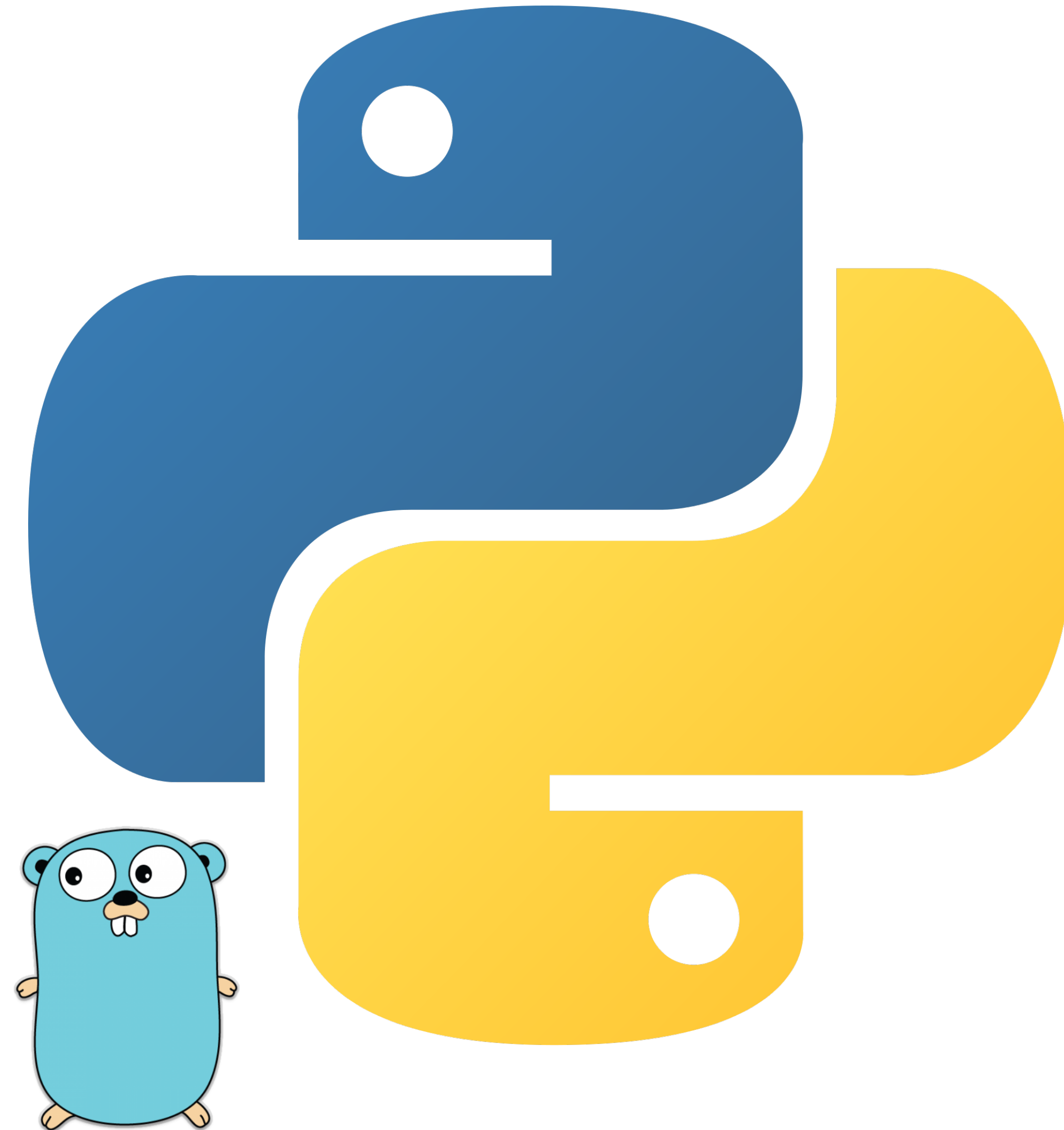
## Hardware dependencies

- A PC with a powerful 64-bit multi core processor and at least 4 gigabytes of RAM.



# PYTHON

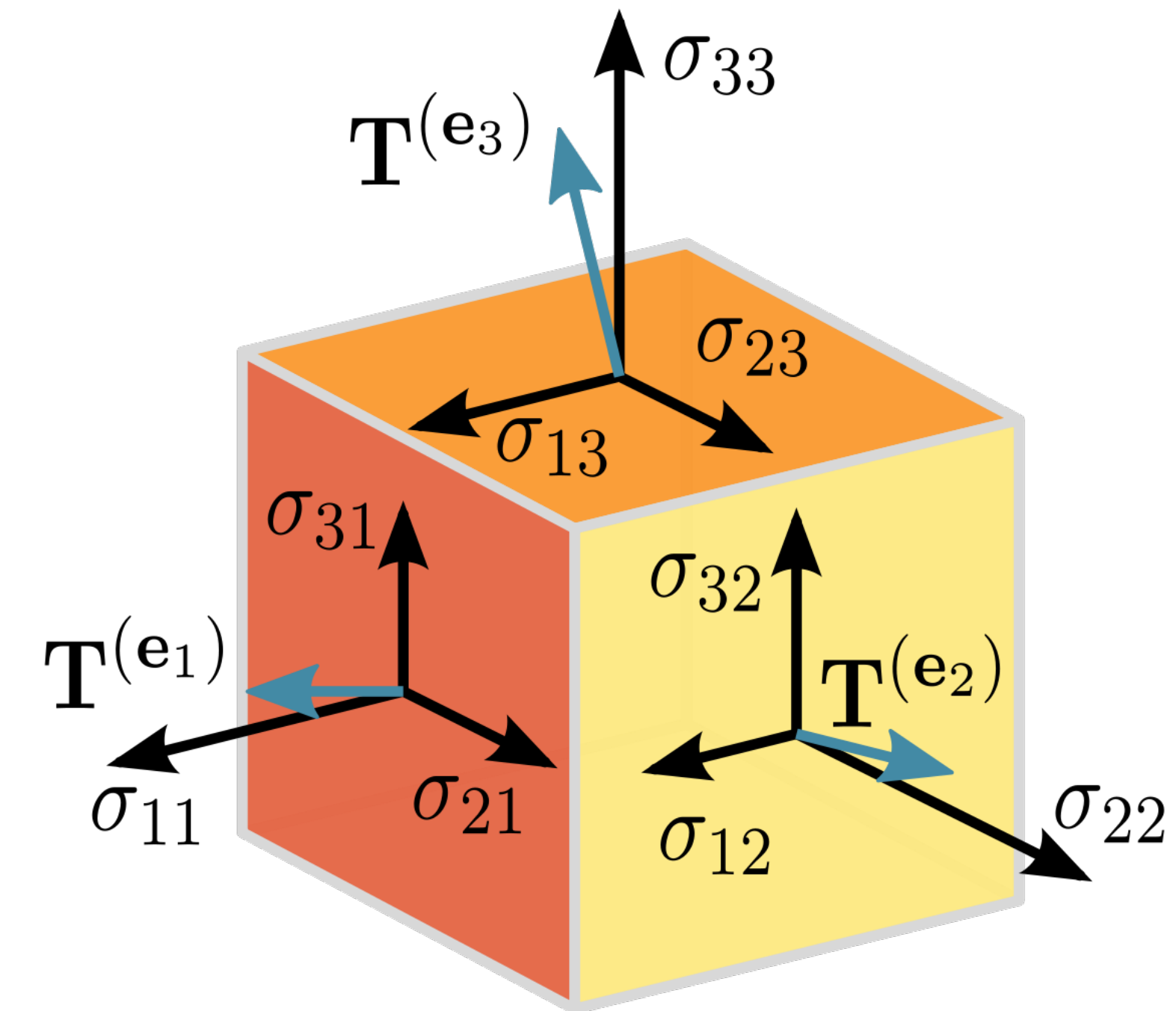
**Python though slower supports a wide range of libraries and easier to find and install dependencies for python than other programming languages capable of ML like “Go” from Google and Python is open source and has good community support making it easier for debugging**





# TENSORS

A tensor is a mathematical object analogous to but more general than a vector, represented by an array of components that are functions of the coordinates of a space.





# TENSORFLOW

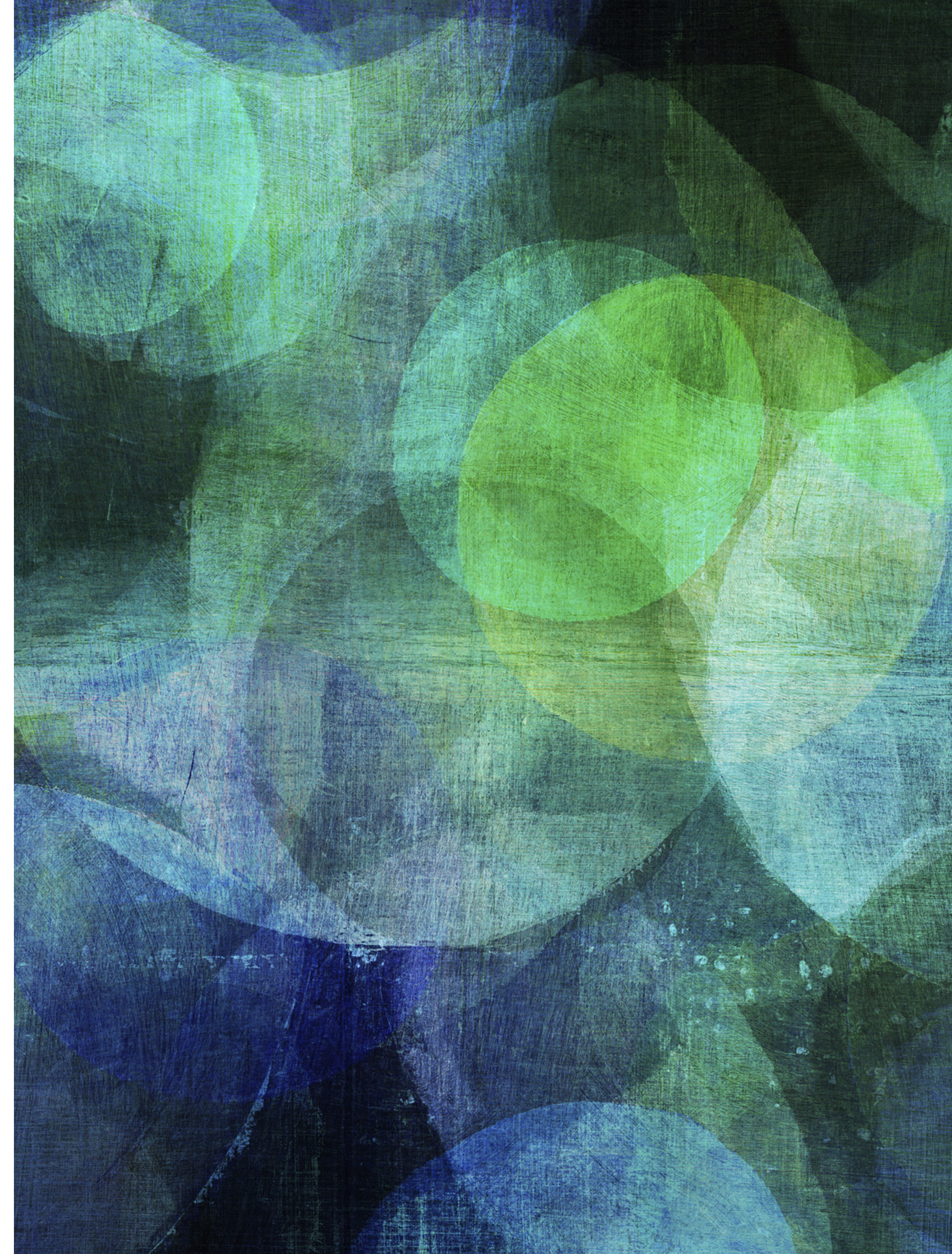
TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API. TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.





# TIMELINE

Duration	Modules
July	Learning Python, Machine Learning / Chapter 1 – Introduction
August	Learning Tensorflow / Chapter 2 – Literature survey, Chapter 3 – System study and analysis
September	Collection of the training and testing dataset and beginning of the development phase of the sGAN machine learning model / Chapter 4 – System development, Chapter 5 – System implementation
October	Developing the sGAN machine learning model / Chapter 6 – Testing and experimental result







# REFERENCES

---

- OPEN AI
- Google Deepmind
- Tensorflow
- Google Magenta



“

With artificial intelligence we are summoning the demon. In all those stories where there's the guy with the pentagram and the holy water, it's like – yeah, he's sure he can control the demon. Doesn't work out.

*-Elon Musk*