

# Gems Sorting Machine

Based on Image Processing and powered by  
Artificial Intelligence

Made By:

Utkarsh Gupta - R177219194

Vanshaj Goel - R177219196

Tanya Malhotra - R177219191

Submitted to:

Dr. Kiran Kumar Ravulakollu

GitHub Repository:

[github.com/AvGeekGupta/Gems-Seperator.git](https://github.com/AvGeekGupta/Gems-Seperator.git)

Website:

[avgeekgupta.github.io/gems-seperator/](https://avgeekgupta.github.io/gems-seperator/)

# Introduction

In today's busy world, when we have an opportunity of reducing the work on man's shoulder, then why not implement it! Sorting an item based on its colour or attributes is a time-consuming task. That is why colour sorters are highly usable in big industries for separating items according to their colour and in filtering the things which do not fall within the acceptable criteria or which are desired by the user.

Similarly, gems sorter machine is used to arrange multi coloured gems individually which is powered by Artificial Intelligence which is rapidly booming nowadays.

## Features

- With a really immense user interface, this machine is extremely easy to use as well as control.
- The neural network is well trained with real time practical data.
- The software (the main Python application) is efficient as it clicks a picture and crops it to the specific size of a gem as per its needs.
- Due to a properly constructed and coherent neural network, this machine is having an accuracy of approximately 83.3%.
- New colours can be added to the system without any hassle.
- The good design of the system promotes easy addition and removal of objects.

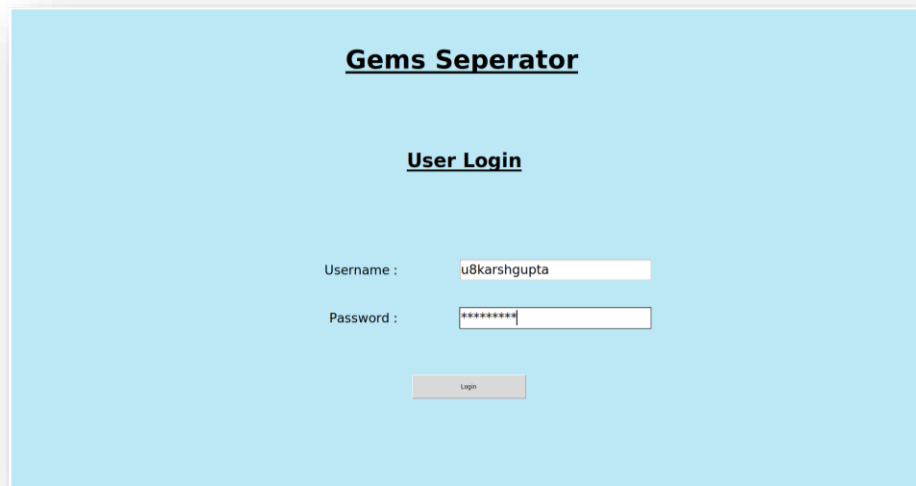
## User Interface

### Made with Tkinter on PAGE

PAGE is a drag-and-drop GUI generator for Python and Tkinter which generates Python modules which display a relatively simple GUI constructed from Tk and ttk widget sets using the Place Geometry Manager. PAGE is a cross platform tool running on any OS which has Tcl/Tk installed. PAGE output requires only Python and Tkinter and runs on Linux, Unix, Windows, OS X, and even Raspbian.

## Login Page

This page has a simple login option for the user to login to the main application.




The screenshot shows a light blue login page titled "Gems Separator" and "User Login". It contains two input fields: "Username :" with the value "u8karshgupta" and "Password :" with masked characters "\*\*\*\*\*". A "Login" button is positioned below the password field.

---

## Main Dashboard

This is the main operating application page. On the right half of the Dashboard we can find a frame where we can see the pictures of the gems that are clicked by the application. Below that is a table that keeps count of the no of gems of each colour. The left Partition is the control that has buttons to log-out, train the model, start the machine, stop the machine and also to write data to database.



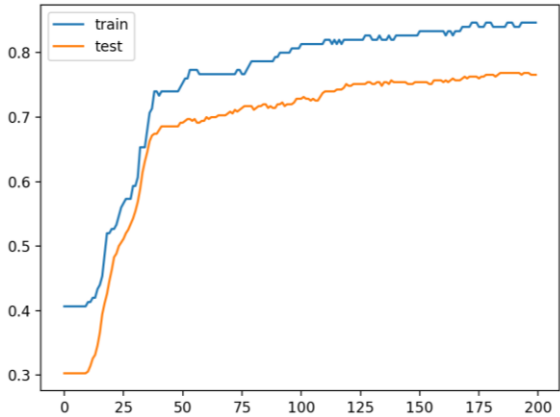
The screenshot shows the main dashboard of the "Gems Separator" application. It features a light blue header with "Welcome Utkarsh Gupta" and the title "Gems Separator". The dashboard is divided into three main sections:

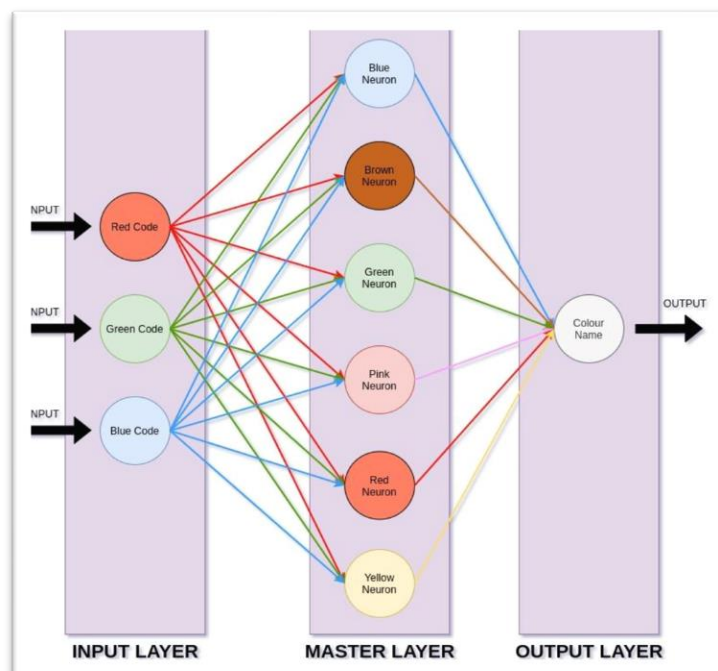
- General Options:** Contains buttons for "Log Out" and "Train System".
- Engine Controls:** Contains buttons for "Upload Data", "Start Engine", and "Stop Engine".
- Write Data on file:** Contains buttons for "Training Graph" and "Run Results".

In the center, there is a large image frame displaying a purple gem. Below the image frame, a table titled "Current Gems data" shows the count of gems by color:

| Color         | Count |
|---------------|-------|
| Blue Gems :   | 0     |
| Green Gems :  | 0     |
| Red Gems :    | 0     |
| Brown Gems :  | 0     |
| Pink Gems :   | 0     |
| Yellow Gems : | 0     |
| Total Gems :  | 0     |

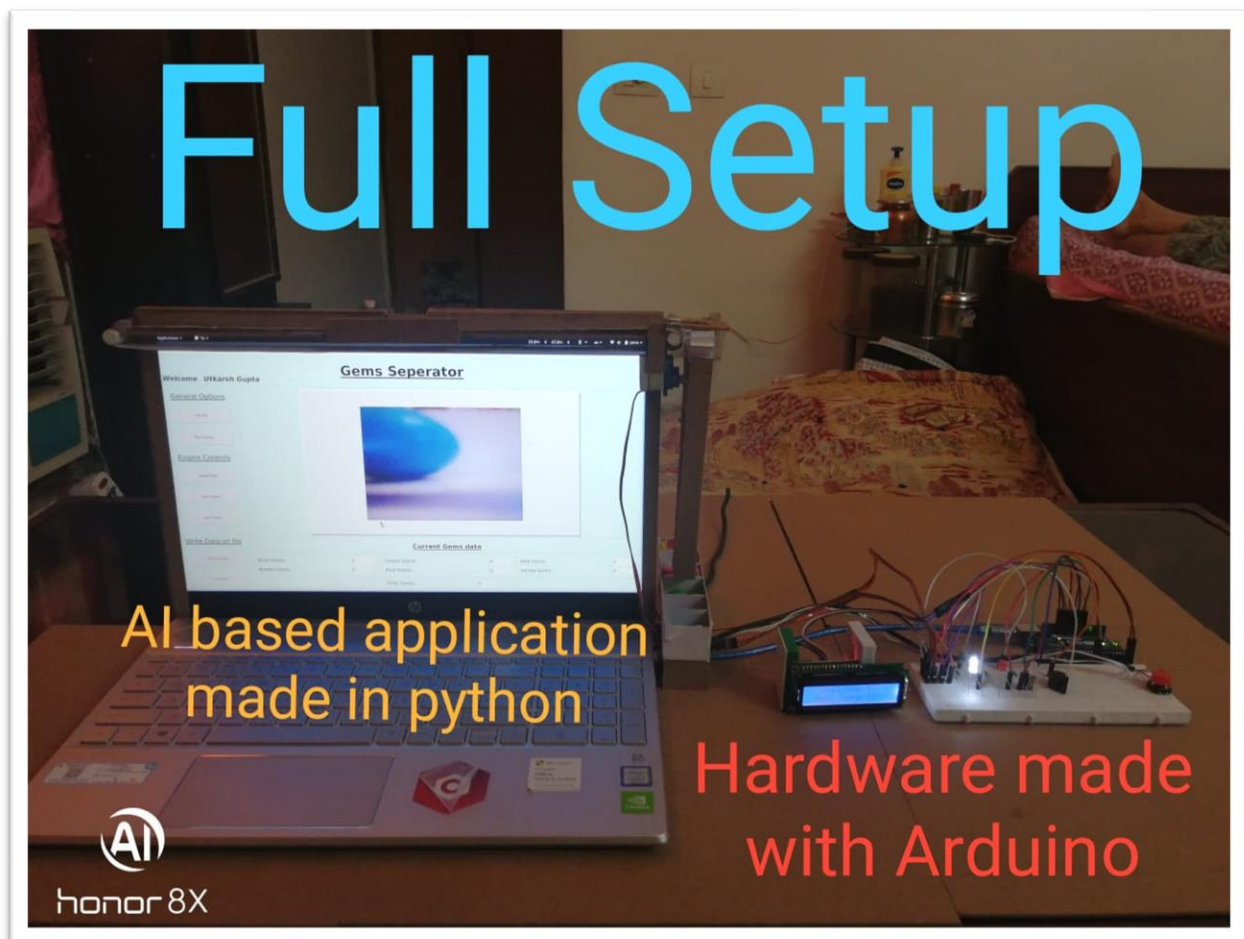
# About Neural Network

- With the help of some fundamental knowledge of a neural network, a neuron was created.
  - Sigmoid function was used as the activation function of artificial neuron.
  - Taking note of the expected value as the output, the error was calculated with the help of sigmoid function and the expected value.
- 
- Three main parameters were considered matching up to a specific colour constituent of an RGB colour model i.e., an additive colour format consisting of broad range of red, green and blue colours.
  - The Neural Network is trained for 200 epochs and achieved the efficiency of approximately 83.3%.
  - It consists of 3 layers. Input, Master and Output Layer as shown in the image below.
  - There is one neuron for each colour and returns the probability of the gem to be of a particular colour.



# Working

The main software which is a Python application clicks a picture and crops it to the specific size of a gem as per it's needs. The colour of the gem is extracted and is converted into the RGB format. After sending the information to the neural network, the exact colour is determined which is provided to the GUI. The main loop informs the Arduino through serial communication, ultimately dropping the gem in the assigned box according to the output of the software.

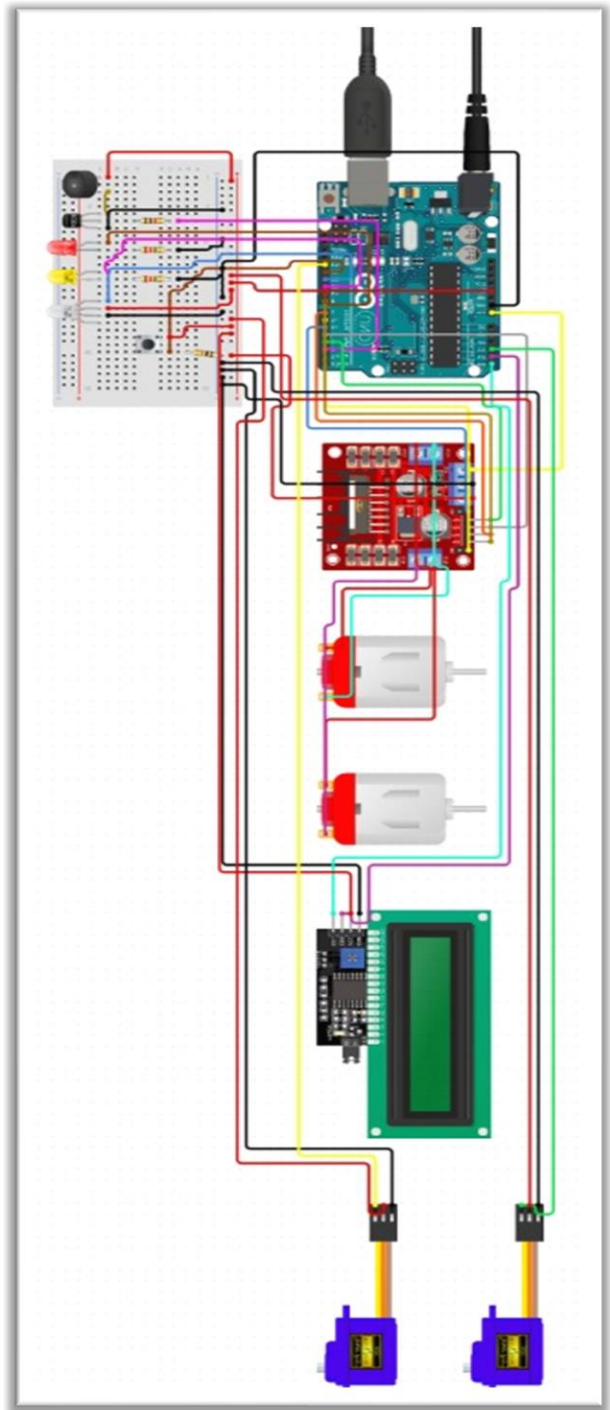
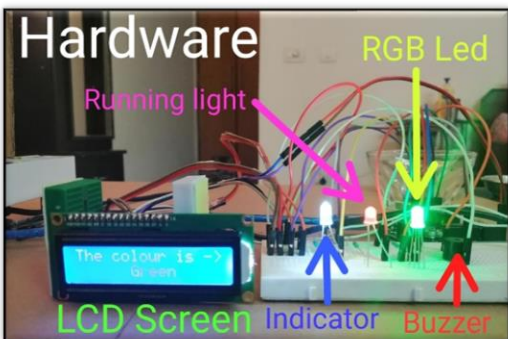


# About Hardware

The hardware is made with Arduino Uno R3 and coded in C/C++ on Arduino IDE.

The major components of the Hardware consists of L293D Motor driver to run the conveyer belt, a servo to turn the dropper towards the identified box, another servo to push next gem on conveyer belt to be identified, Indicator LED's, RGB Led to tell the colour of the gem and an LCD screen to display other essential data to the user.

It connects to the main application through serial communication.



# Conclusion

So, here we have got a pretty efficient sorting machine, which is not only very convenient to use but also has an extremely attractive design structure. The user interface is friendly and accuracy rate is unbelievably very high. The gems can now be easily differentiated and assigned into different boxes according to their colour. This machine can also be modified for some big items on a larger scale.

## Market Survey

Such sorting machines are used in the huge industries like coca cola to sort out faulty products from their high speed assembly lines.

These systems are very fast and can work at the approximate 10 to 15 bottles per second. These systems increase production speeds, reduces costs, reduces human inputs and human errors. The model needs to be trained only once and it can go for forever until the industry decides to change their product or the product design.



## TechData

- Operating System – RedHat Fedora 30 (Linux).
- Languages – Python3, C/C++, SQL.
- Database – MySQL.
- Software used – JetBrains' PyCharm, Oracle's MySQL, PAGE.
- Dependencies – OpenCV, SerialCommunication, SciPy, NumPy, Tkinter, MySQL