**REPORT ON INTERNSHIP WORKS**

|  |  |
| --- | --- |
| **Name** | **D K KARMUKILAN** |
| **Roll.no** | **727621BAD016** |
| **College** | **Dr. MCET POLLACHI.** |
| **Date** | **19.06.2023 – 01.07.2023** |
| **Company Name** | **Deep2Neuron Tech Academy** |
| **Department** | **Artificial Intelligence and Data Science** |

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Day** | **Page no.** |
| **1** | **Day-1** | **2-3** |
| **2** | **Day-2** | **4-7** |
| **3** | **Day-3** | **8-11** |
| **4** | **Day-4** | **12-14** |
| **5** | **Day-5** | **15-17** |
| **6** | **Day-6** | **18-21** |
| **7** | **Day-7** | **22-27** |
| **8** | **Day -8** | **28-31** |
|  |  |  |
|  |  |  |
|  |  |  |

**Report for 19.06.2023**

**Introduction:**

Day-1 Internship majorly includes selection of Projects, Running the code on online resources. There are many challenges therewith will be discussed in the upcoming points.

**Activities assigned:**

* Need to choose the project and its title.
* Want to find algorithms related to CNN.
* Want to get code from online sources.
* Need to run the code in Colab and get the accuracy.
* Want to create a GitHub and make a new repository to post daily reports .

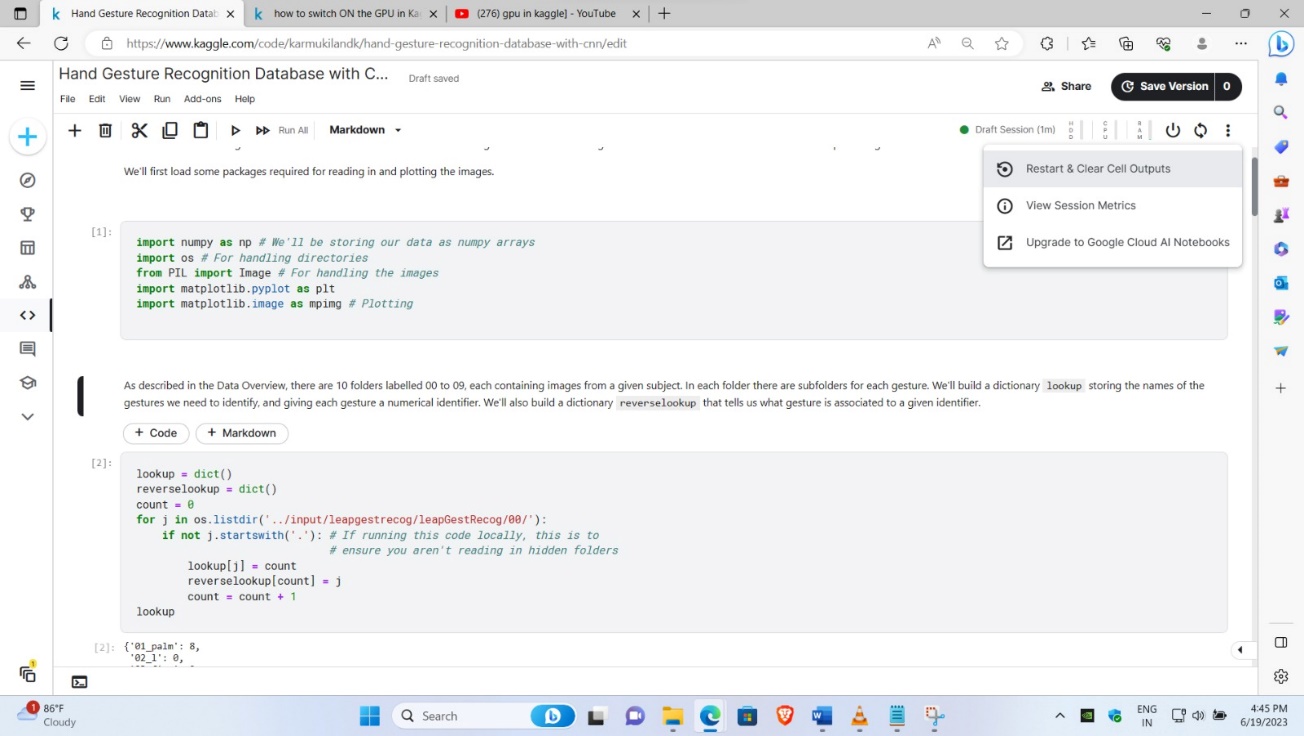
**Activities Completed:**

* Project: Hand Data recognition (Image Dataset).
* Taken CNN algorithm and run successfully for 10 epochs.
* Due to time constrains epochs has been changed from 10 to 3 in Kaggle.
* Link for Dataset: <https://www.kaggle.com/datasets/gtiupm/leapgestrecog?datasetId=39466&searchQuery=cnn>
* Code ran in kaggle had run successfully.

**Challenges & Doubts:**

* The primary task is to train the model so it did not run on Kaggle but there is no option which is visible for enabling GPU.
* **Solution:** [**https://medium.com/featurepreneur/how-to-use-kaggle-gpu-74b1e184242c**](https://medium.com/featurepreneur/how-to-use-kaggle-gpu-74b1e184242c)this link has enable GPU methods but to verify with phone numbers were not mentioned so it has been identified from various online sources and enabled.
* Running code in Colab has disadvantage of connecting the Kaggle or downloading dataset uploading and getting data processed which is simplified by Kaggle.

**Screenshots for Day – 1**



**Conclusion:**Many challenges has been cleared with lot of open online resources available and cleared doubts that were most probably common in all cases.

**Report for 20.06.2023**

**Introduction:**

Day-2 Internship majorly includes selection of Particular Project which includes data collection, Collecting Data from various sources, Preprocessing and starting with algorithms. Challenges are quite good in these Day 2.

**Activities assigned:**

* Need to choose the project related to Image classification.
* Want to get data collected from various online sources.
* Need to make the folder structure with Test which consists of 50 images in each folder, Validation of 30 Images in each folder and Train Images which is remaining.
* Total number of Images includes more that 300 images.
* Needed to upload in Drive and mount it in colab with GPU backend.

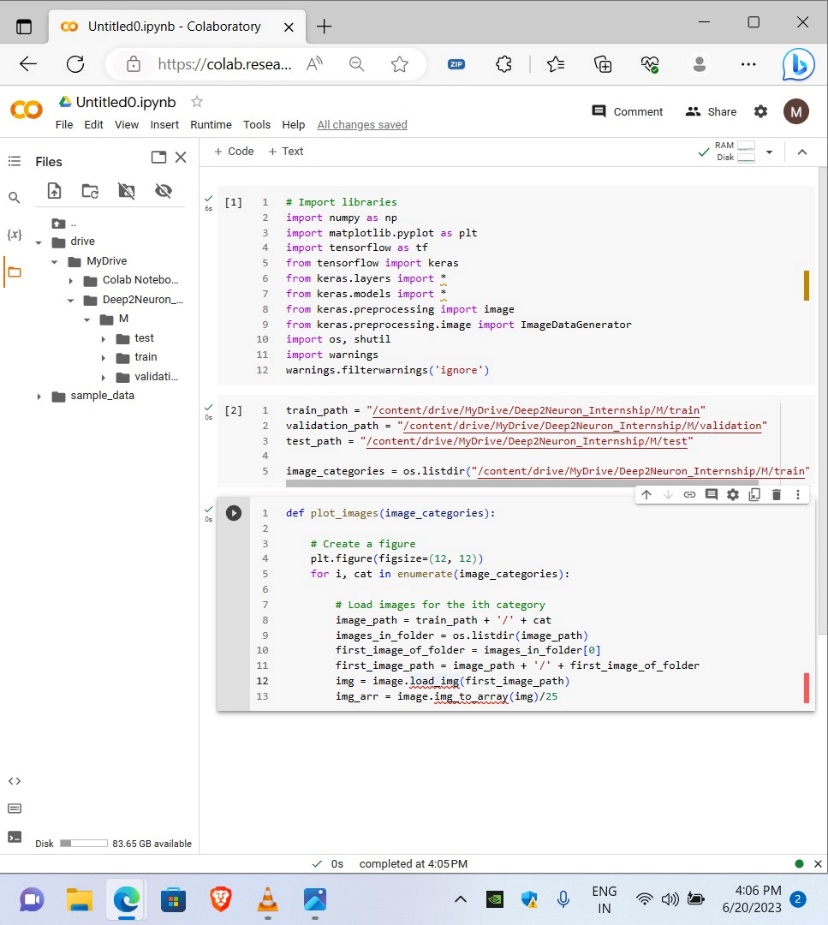
**Activities Completed:**

* Project: Image classification of Happy or Sad with CNN.(Multiclass)
* Code has been given during the meeting and changes has been made to our code accordingly.
* Code ran in Colab has successfully run with 20 epochs and plotted graph for the training loss, validation loss, Training accuracy and validation accuracy.
* Image has been tested for its prediction.
* Learned about Binary and Multiclass and what are changes to be made if it is binary or multiclass has been known from the meeting.

**Challenges & Doubts:**

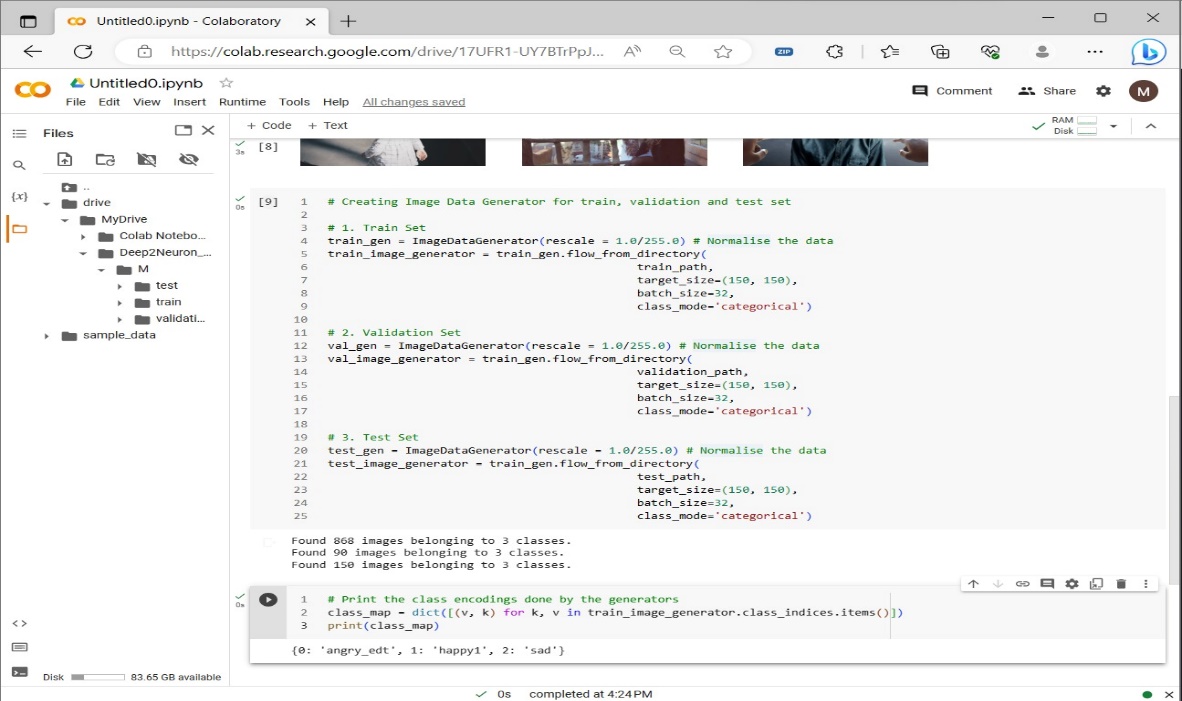
* The error started with the loading of images where keras.preprocessing.image has been depreceated so there by searching online I found and made the solution work in our code which is running import keras.utils as image worked well.
* After finding graph had difficulty in understanding why these were plot and did not get any clarity about the code but soon after started to learn and know from various sources.

**Screenshots for Day – 2**

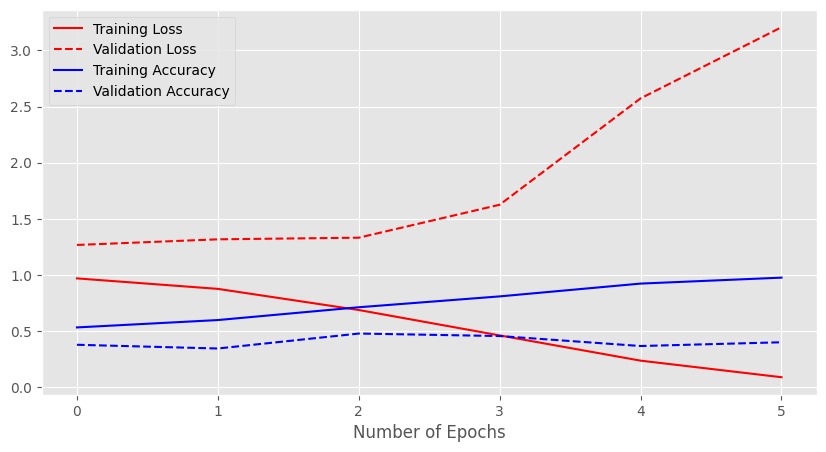


\

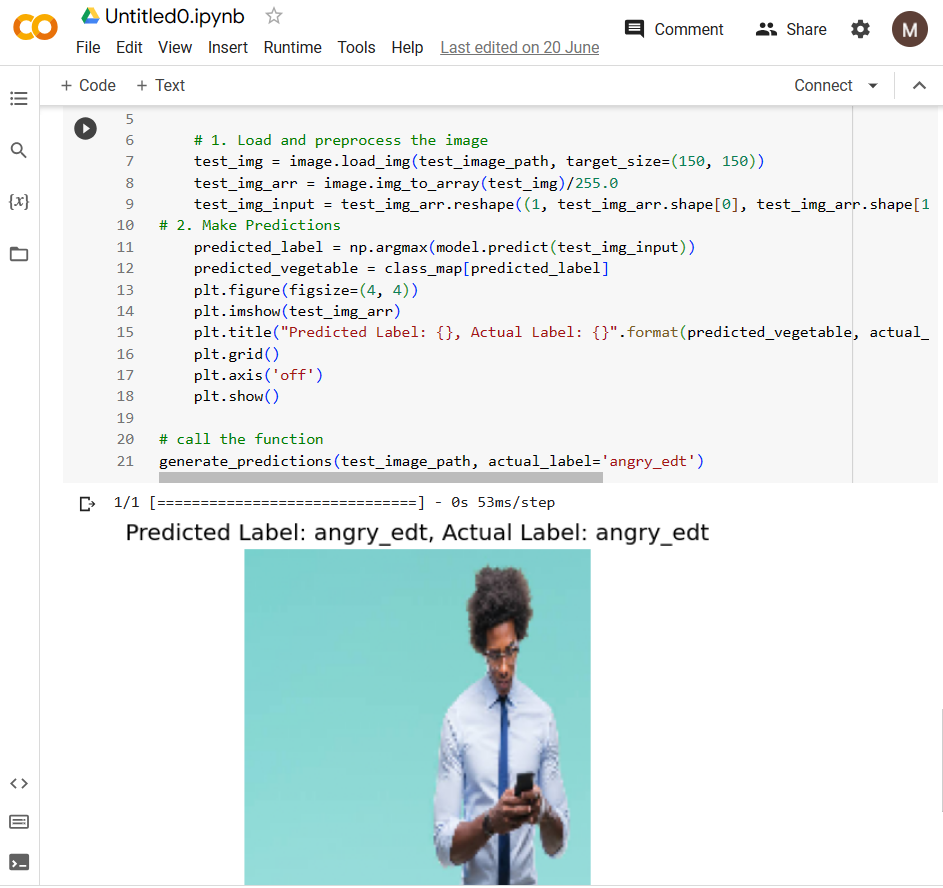
**Code with error**



**Code after implementing solution.**



**Graph plotting accuracy and loss**

****

**Model predicting the image.**

**Conclusion:**Day ended with lot of confusions and take aways from the work done. Also learned a lot of new things on the day which includes the Data Preprocessing, Training model, class modes like categorical or binary with various live codes.

**Report for 21.06.2023**

**Introduction:**

Day-3 Internship includes the process of downloading data given by the instructor and need to upload and train the model using CNN as it is a Image classification problem.

**Activities assigned:**

* Need to download and upload given Exam Data
* Want to apply CNN code based on past learnings and identify class accordingly and make changes according to the classes.
* Wanted to run code for 20 epochs at least.

**Activities Completed:**

* Uploaded and respective preprocessing techniques were applied based upon class as it was 5 class classification problem where it consists of folder Train Validation and Validation 2.
* I have used Validation 2 as my test data and trained the model and plotted the confusion matrix, respectively.
* That confusion matrix infers that it predicted the model with class 1 accurately as compared to others and other classes has decent percent of validation which concludes that it has found better.

**Challenges & Doubts:**

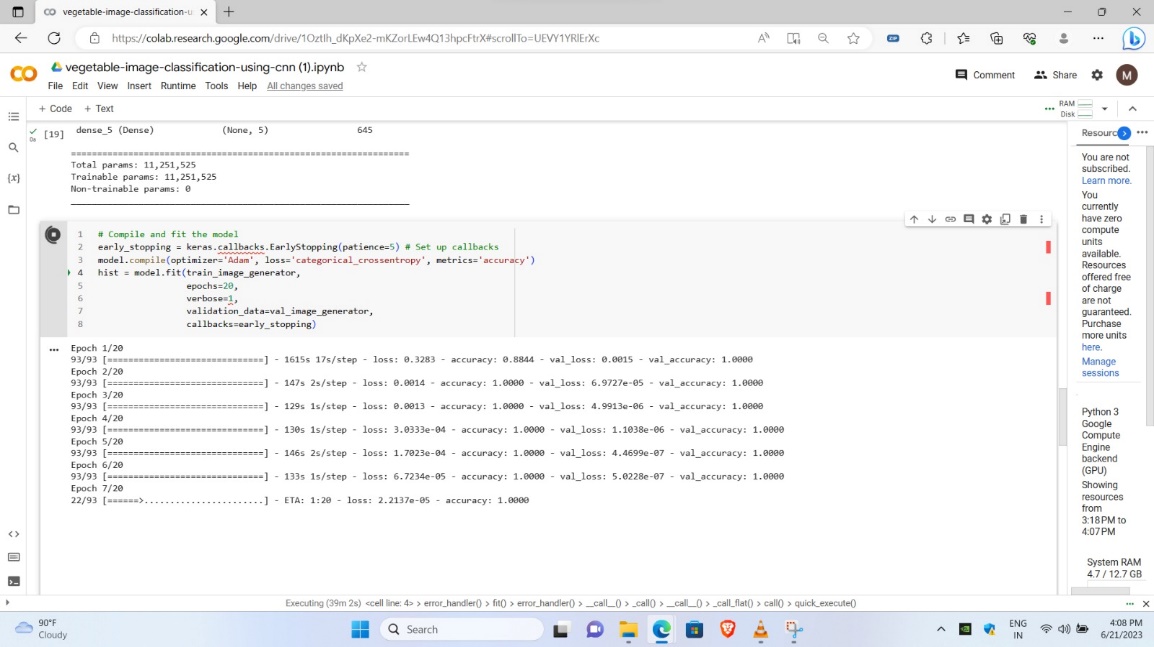
* Got an accuracy of 1 and had doubt that it is overfitting since it is epoch 2.

Solution got is to add **dropout** which is popular regularization technique used to prevent overfitting. <https://towardsdatascience.com/dont-overfit-how-to-prevent-overfitting-in-your-deep-learning-models-63274e552323> link from instructor.

Tried and reached 20 epochs without overfitting of model.

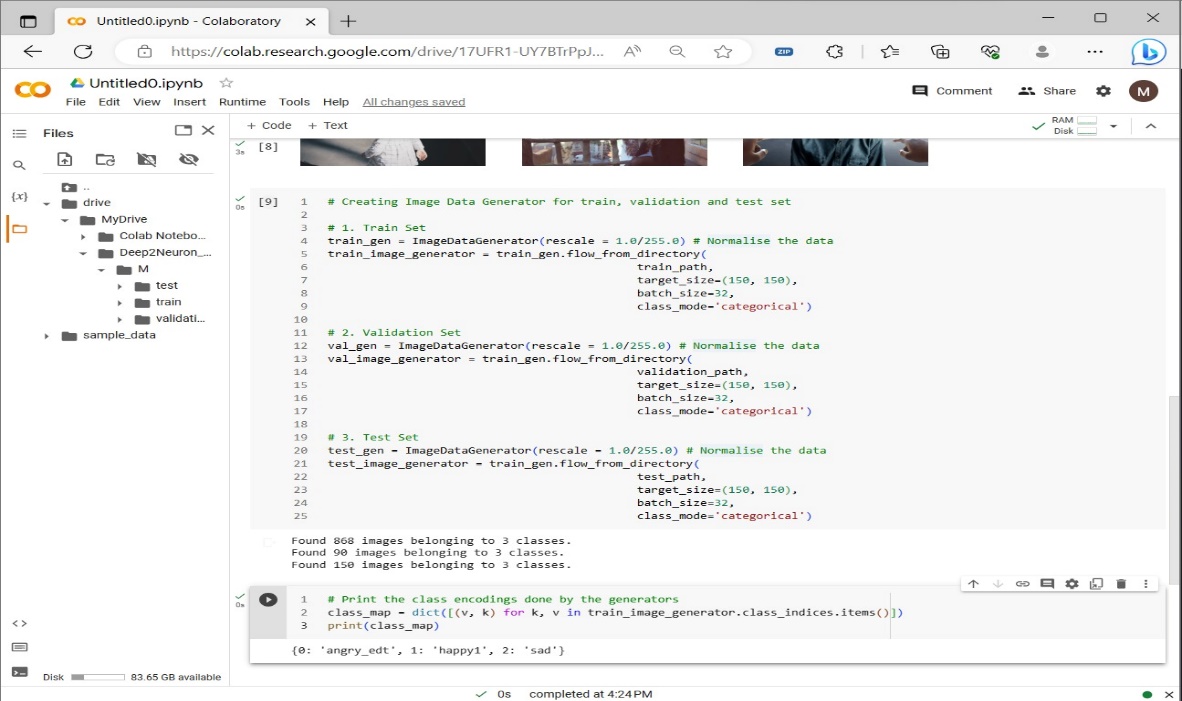
* Since getting overfitting it is hard to find that model is training appropriately so plotted the confusion matrix and graph where it gave better result and concluded that model is training properly as from graph as the epochs increases the validation accuracy increases which is a positive sign.

**Screenshots for Day – 3**

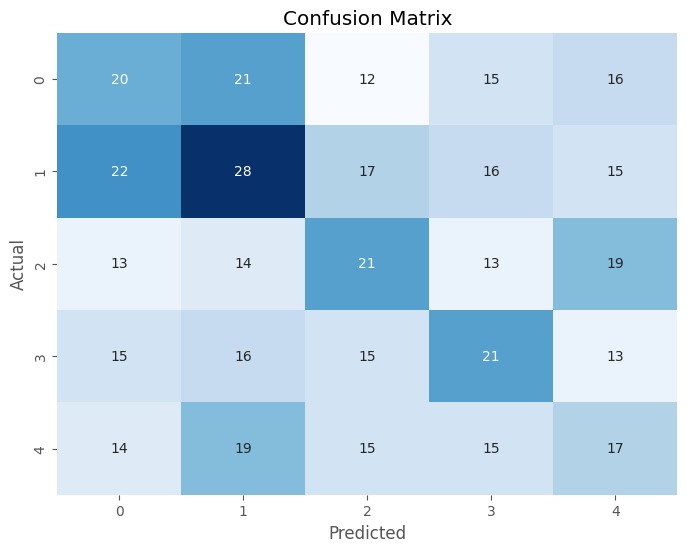


\

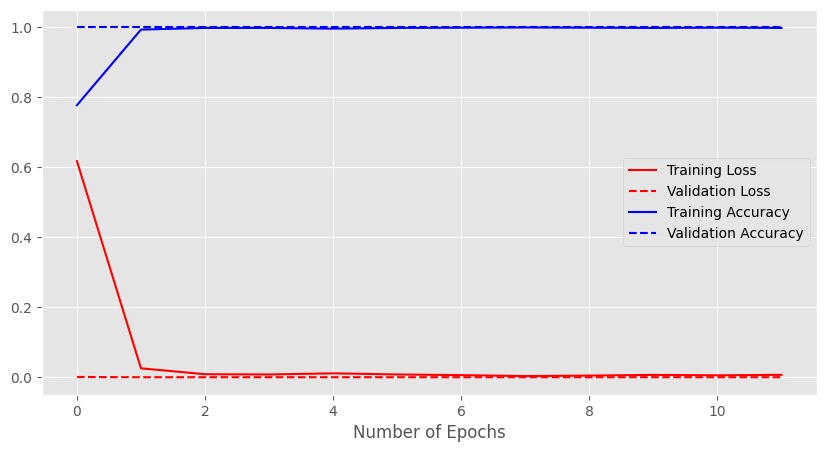
**Algorithm overfitting**



**After adding dropout**



**Confusion matrix with class**

****

**Graph plotting accuracy and loss**

**Conclusion:**Day has been concluded with lots of inputs and outputs. Also learned a lot of new things on the day which includes regularization techniques and many more new concepts.

**Report for 22.06.2023**

**Introduction:**

Day-4 Internship continues with Day 3 activities and need to fine tune model and get the output for that training model.

**Activities assigned:**

* Wanted to run code for 20 epochs at least and plot training and accuracy matrix.
* Need to run our code on local system and either download and make code work in local system.

**Activities Completed:**

* Downloaded and used PyCharm which is product from jetbeans and successfully run the same code in the local system.
* Run my code successfully with faster training of model but with varied output as compared to colab.

**Challenges & Doubts:**

* Got a Varied outputs while running in colab and local system as colab is based purely on network and if it is good then works faster but local system supported better than the colab and got output finally.
* Learned to install packages as it is varied from one to another as for comparison:

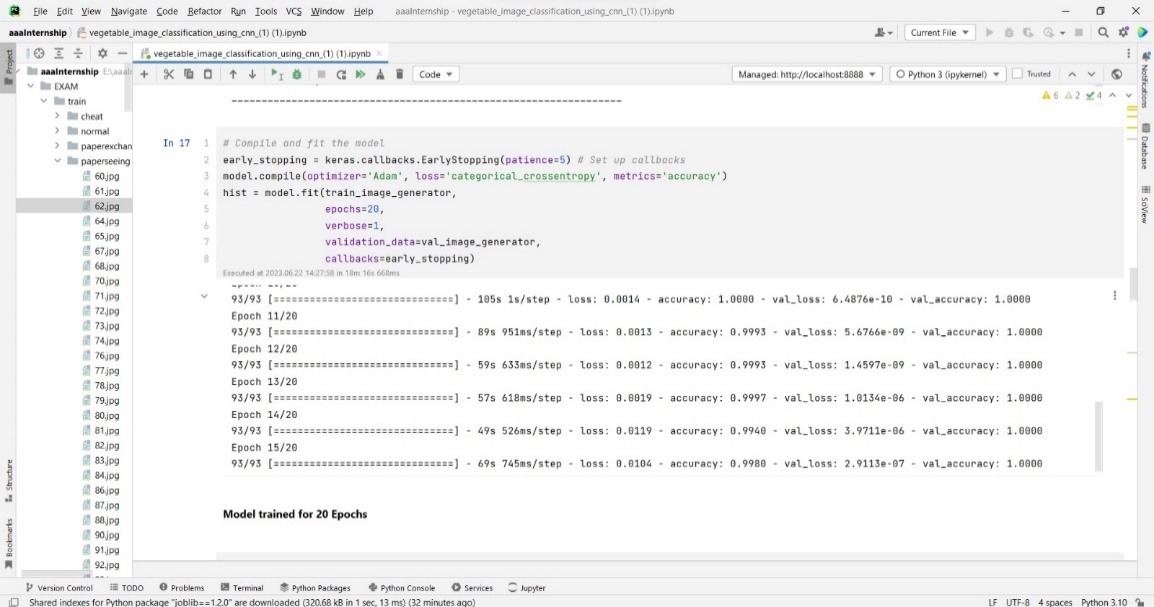
**Colab:**

pip install tensorflow

**PyCharm:**

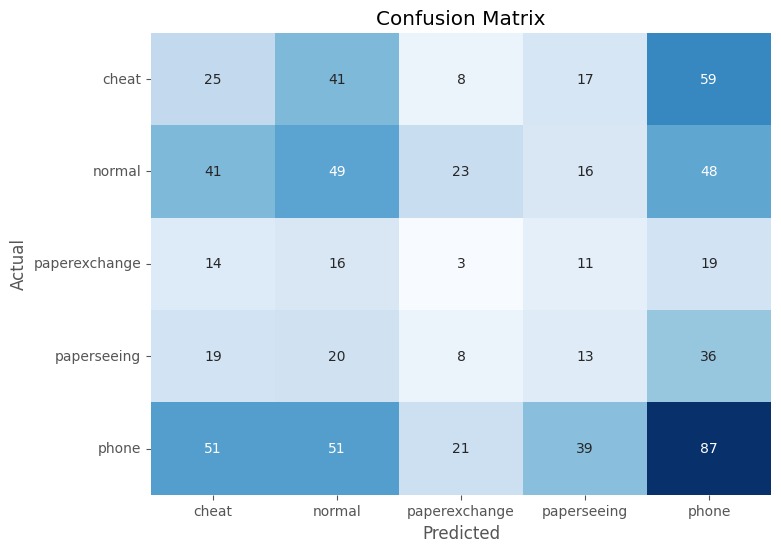
! pip install tensorflow

**Screenshots for Day – 4**

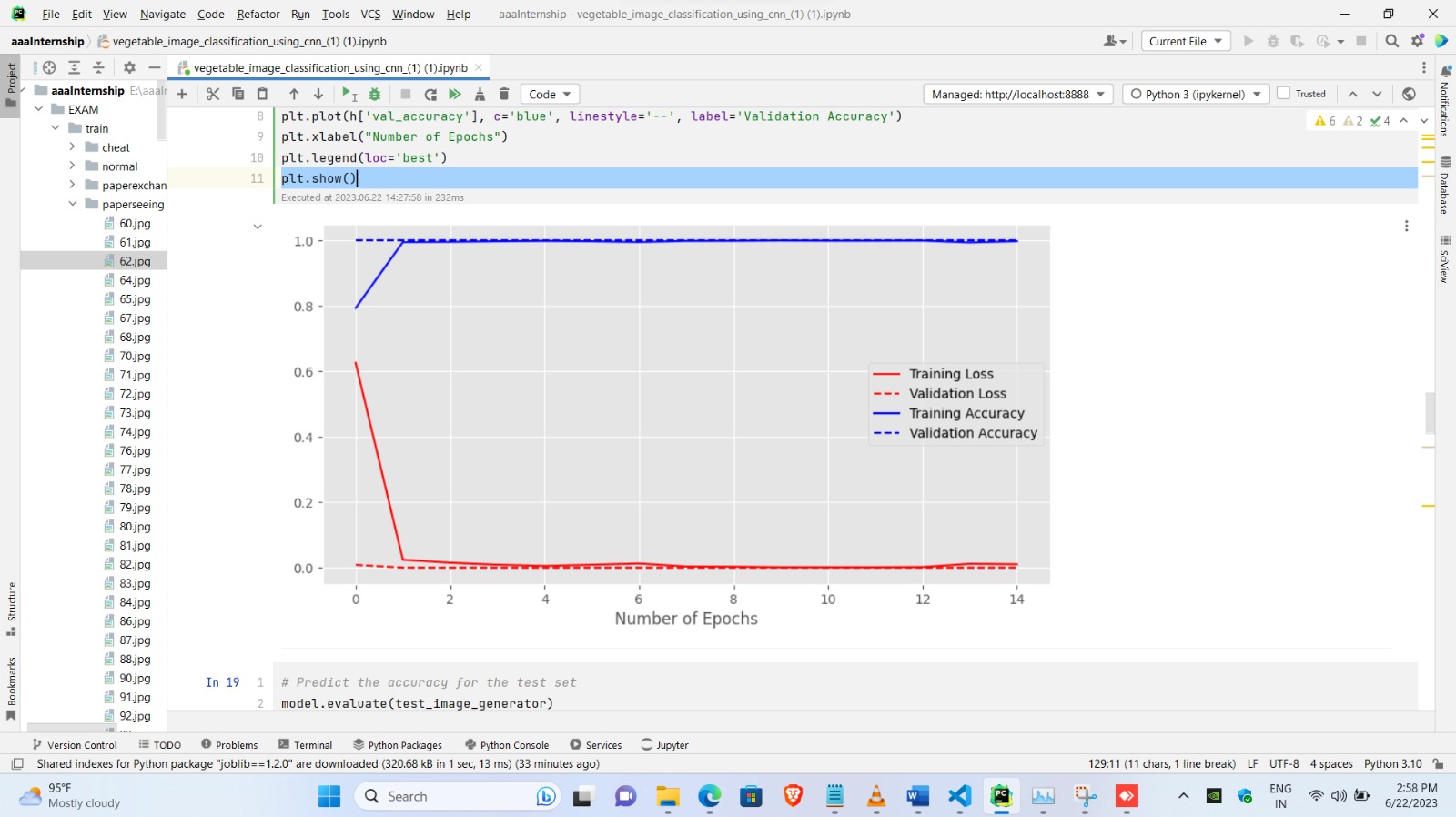


\

**Trained model model**



**Confusion matrix with class names**

****

**Graph plotting accuracy and loss**

**Conclusion:**Day has been concluded with training the model with same code in the local system and has better clarity to set paths and rectify errors in local system.

**Report for 23.06.2023**

**Introduction:**

Day-5 Internship has many new learnings and ideas which could be easy to train a model rather than creating a new architecture.

**Activities assigned:**

* Want to download new data sent which is architecture dataset.
* Wanted to run code with pretrained model and need to make better model.
* Want to refer links sent which gives ideas about which pretrained models can be used.

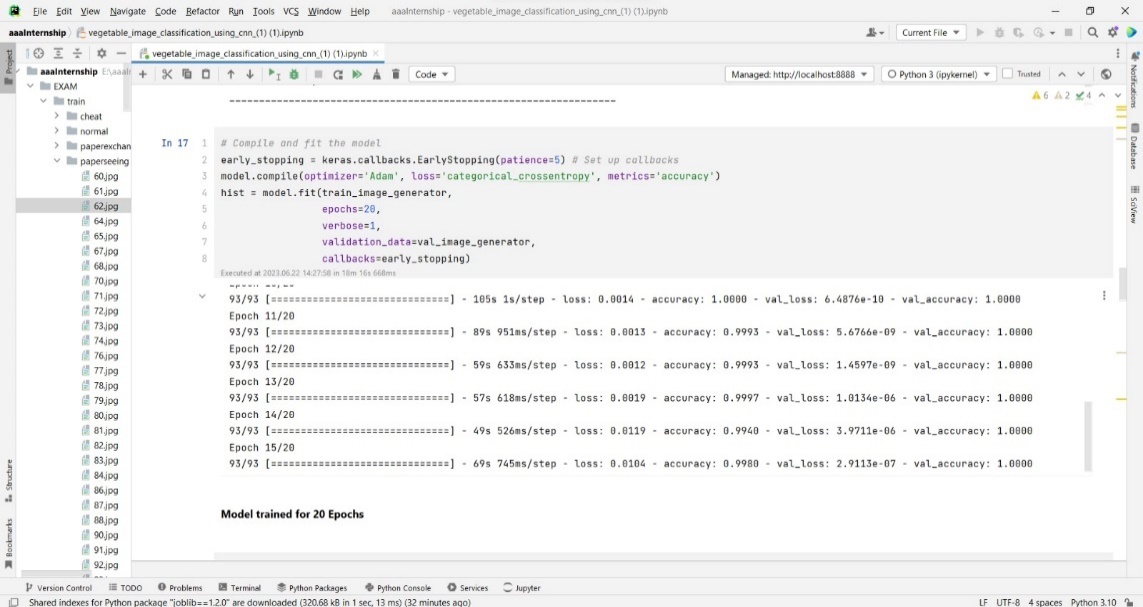
**Activities Completed:**

* Downloaded and learned about dataset.
* It was dataset with 10 classes and found it was multiclass problem.
* Changed accordingly the parameters and run code for VGG 16 pretrained model with 0.01 learning rate with 100 epochs and batch size of 64 and achieved accuracy of 92 percent approximately.

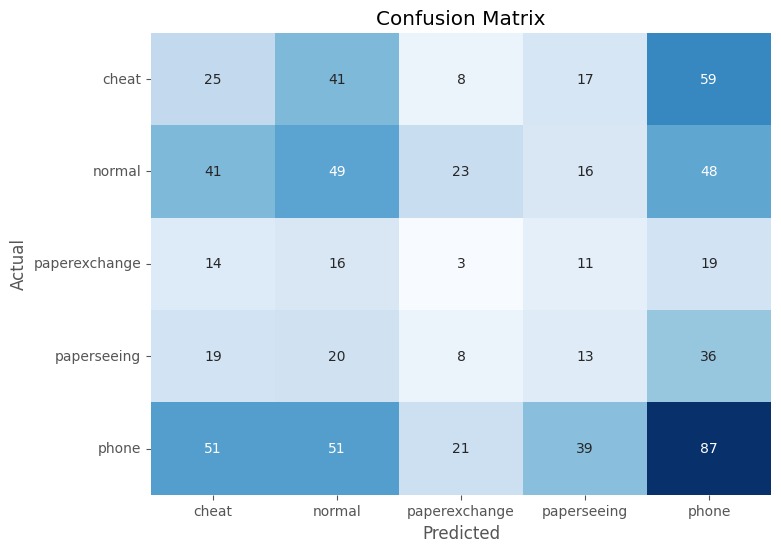
**Challenges & Doubts:**

* Code took a lot of time in training and even using GPU in colab disconnects randomly also has constraint that after using minimal amount of GPU it has limiting us to provide GPU.
* After running 100 epochs gave an output of 0.92 accuracy level.

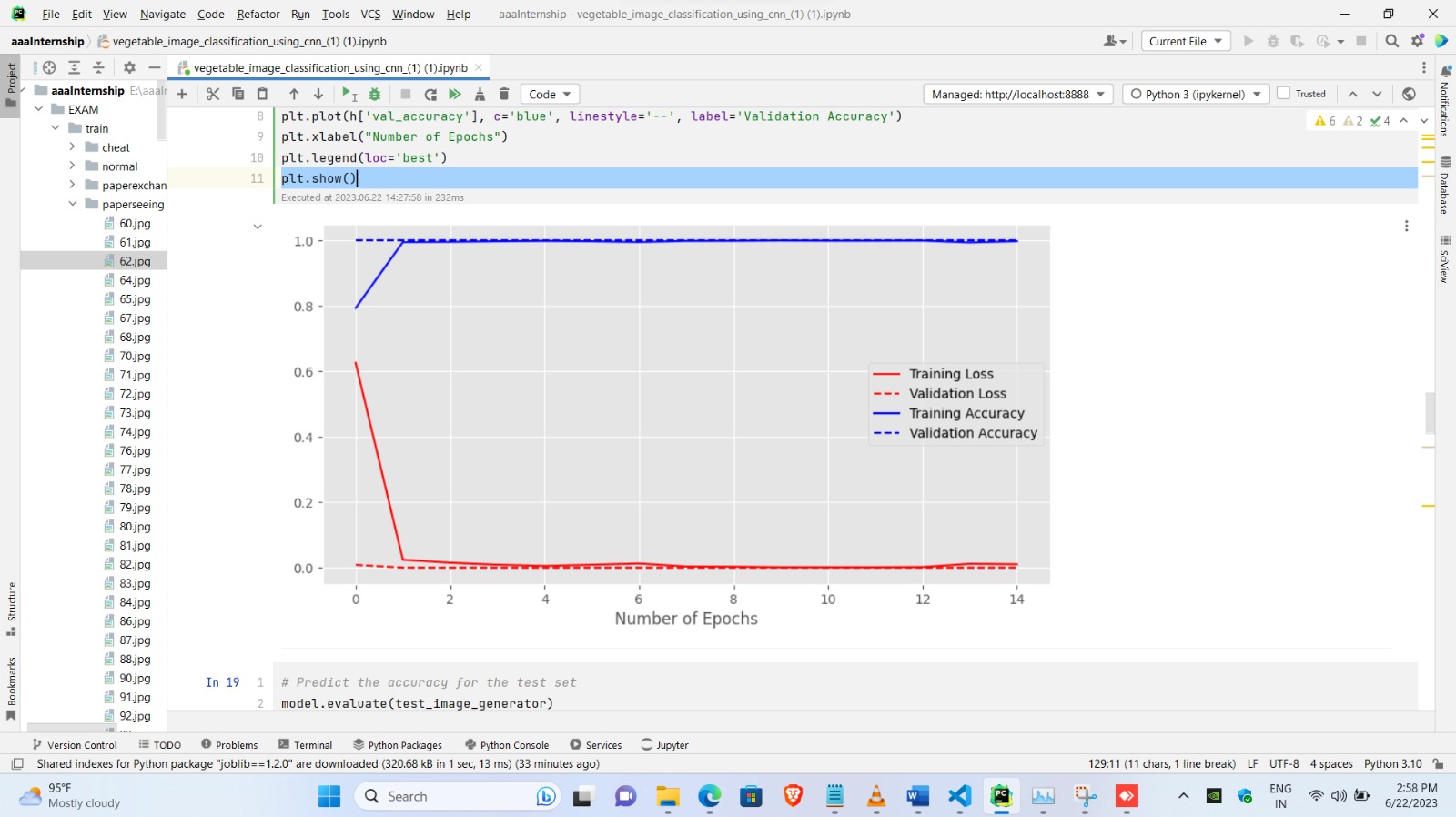
**Screenshots for Day – 5**



**Trained model model**



**Confusion matrix with class names**

****

**Graph plotting accuracy and loss**

**Conclusion:**Day has been concluded with training the model with same code in the local system and has better clarity to set paths and rectify errors in local system.

**Report for 24.06.2023**

**Introduction:**

Day-6 Internship has many new learnings and ideas which could be easy to train a model rather than creating a new architecture.

**Activities assigned:**

* Wanted to run VGG or any pretrained model code and made a detailed report for it.

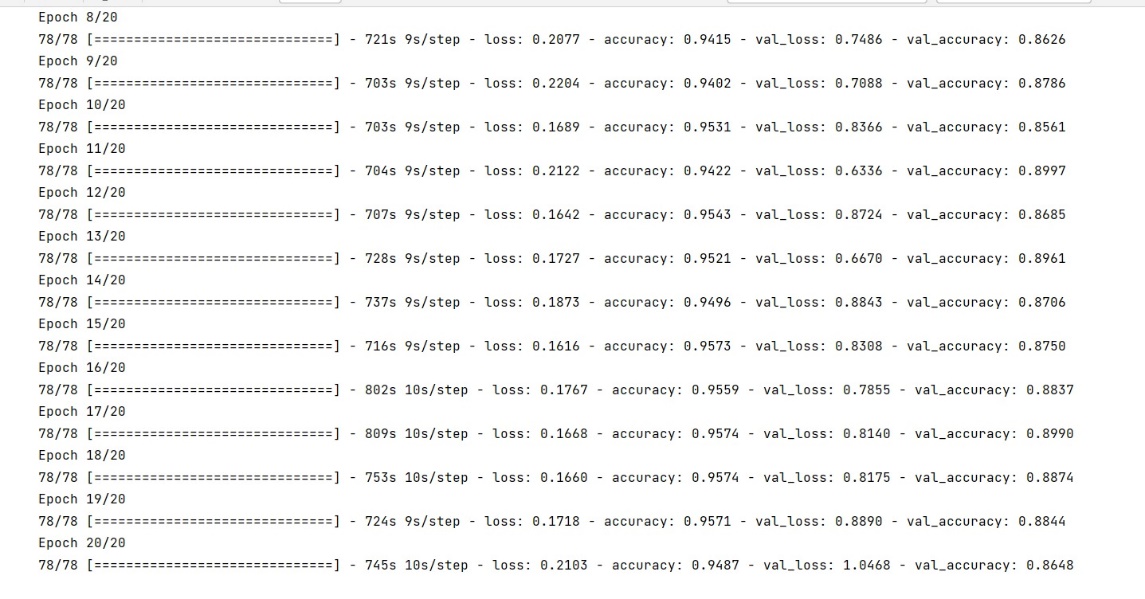
**Activities Completed:**

* Trained a VGG model with 100 epochs which took 6 hours of time which indicates that model struggled to learn the data and fit in it.
* The model has trained with the parameters of 0.01 learning rate with 128 batch size.
* It gave an accuracy of 0.92 which is quite good but it has not reflected in confusion matrix which was slight disadvantage.
* Learning rate with 0.1 for VGG16 with batch size 128 and 100 epochs gave a drastic reduce in accuracy and multiple fluctuations in training

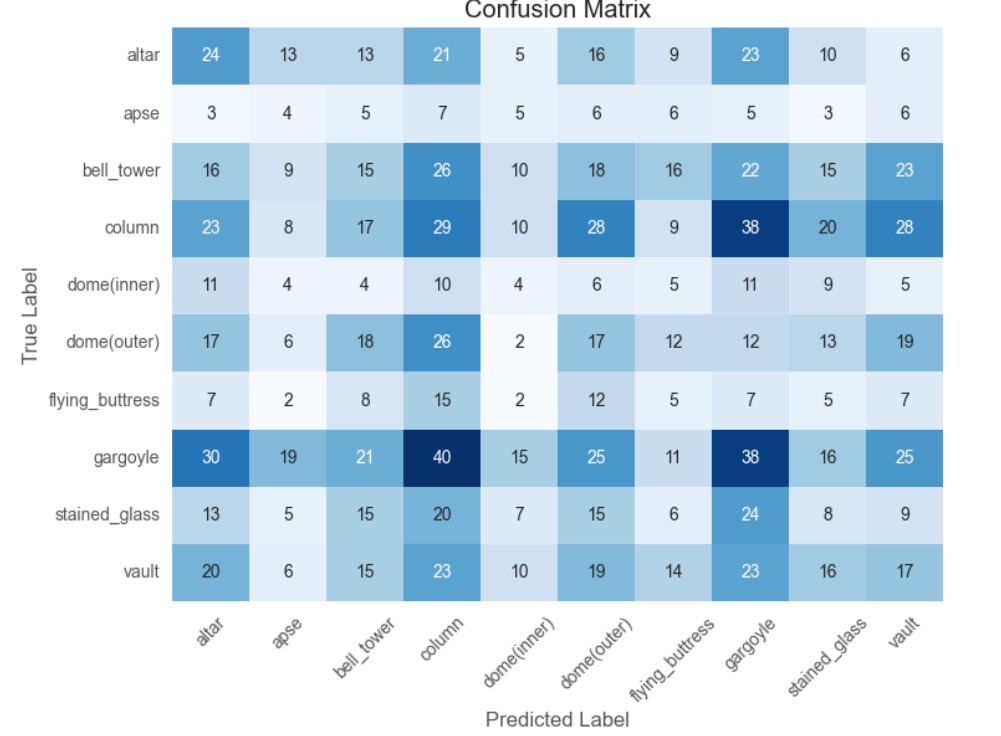
**Challenges & Doubts:**

* Code took a lot of time in training and epochs and batch sizes were too large but to train a larger dataset it took time which was too challenging.

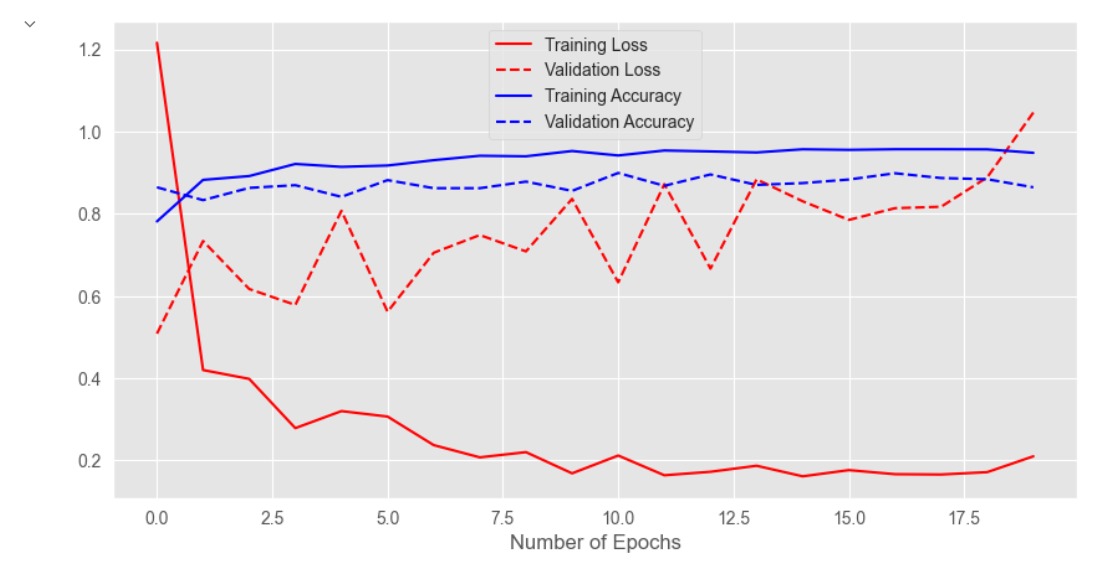
**Screenshots for Day – 6**



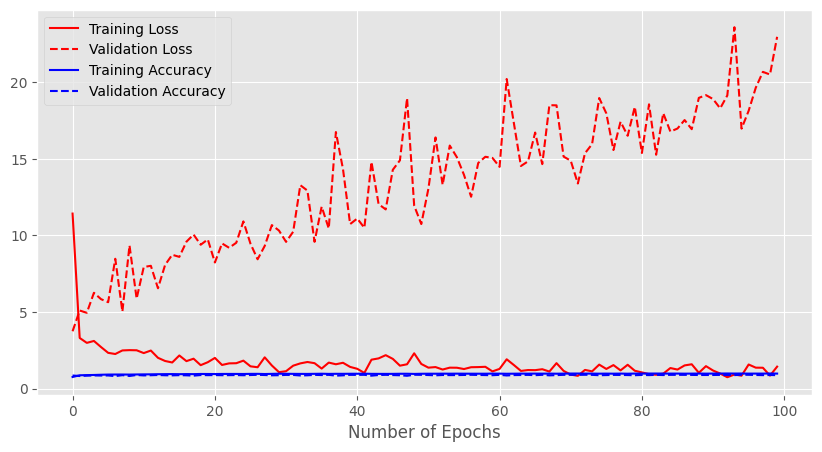
**VGG16 Trained model model (0.01 lr)**



**Confusion matrix with class names**

****

**Graph plotting accuracy and loss**

****

**Graph plotting accuracy and loss for VGG 16 (0.1 lr)**

**Conclusion:**Day has been concluded with training the model with pretrained architectures with different outputs and results.

**Report for 25.06.2023**

**Introduction:**

Day-7 Internship has continuation of previous day work which is to train the model with pretrained architecture.

**Activities assigned:**

* Wanted to run VGG or any pretrained model code and made a detailed report for it.

**Activities Completed:**

* To experiment with I have used MobileNet and ResNet101.
* I have got the lowest accuracy in both models with 0.001 and 0.01 learning rates and 100 epochs respectively with the batch sizes of 128.

**Challenges & Doubts:**

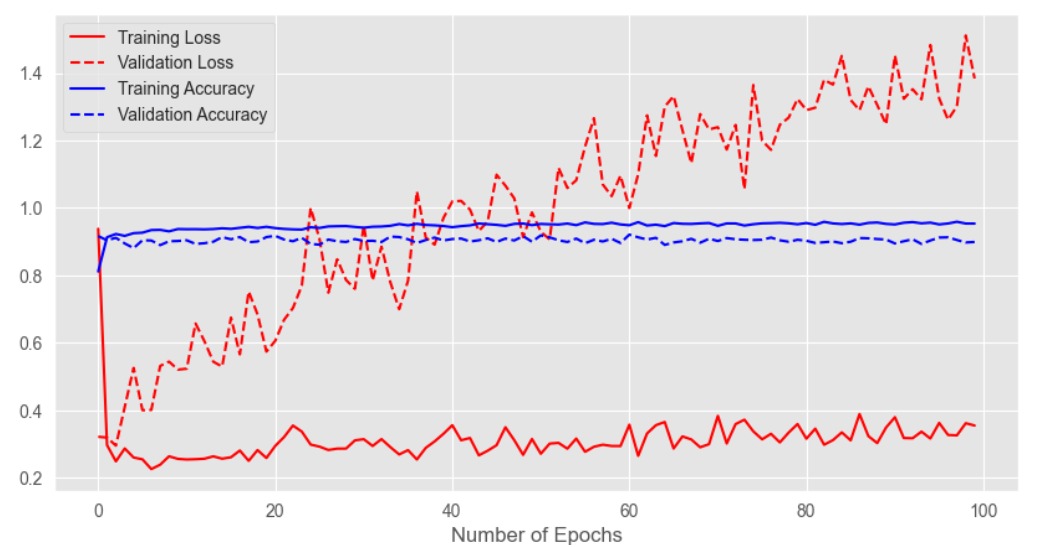
* It was too challenging to run ResNet model as it has run the whole day with the high progress and memory consumption for the model training.

**Screenshots for Day – 7**

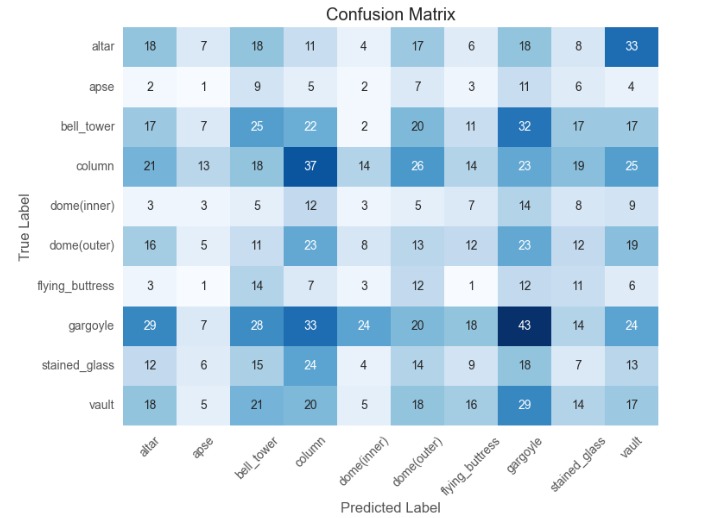
**MobileNet:**



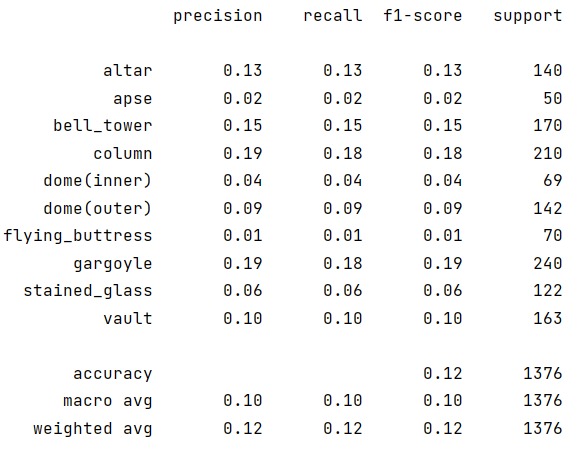
**Running Epochs with accuracy of 0.95 atlast**



**Graph for validation and testing accuracy and loss**



**Confusion Matrix**

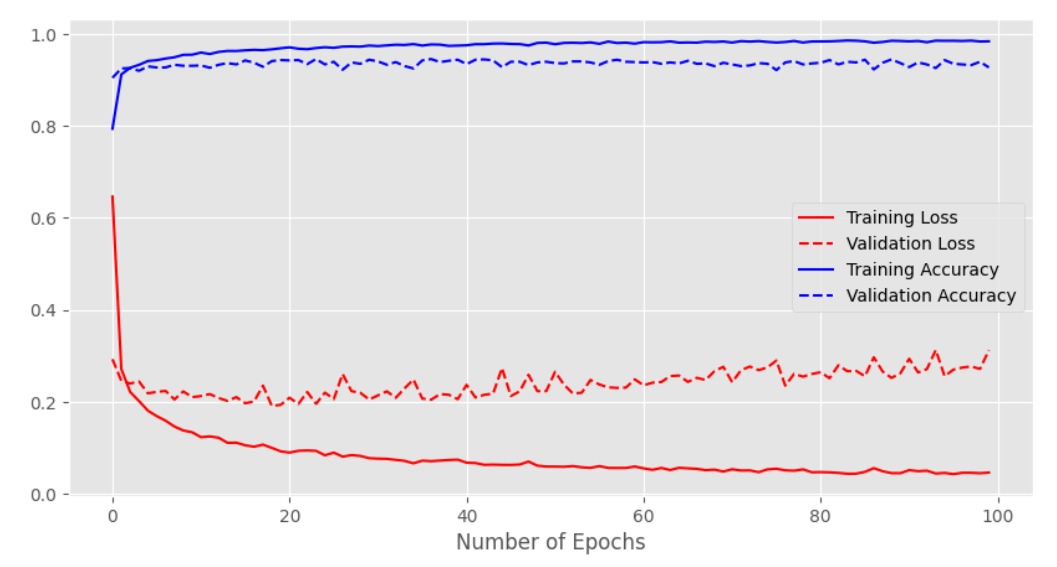


**Classification Report**

**ResNet101**

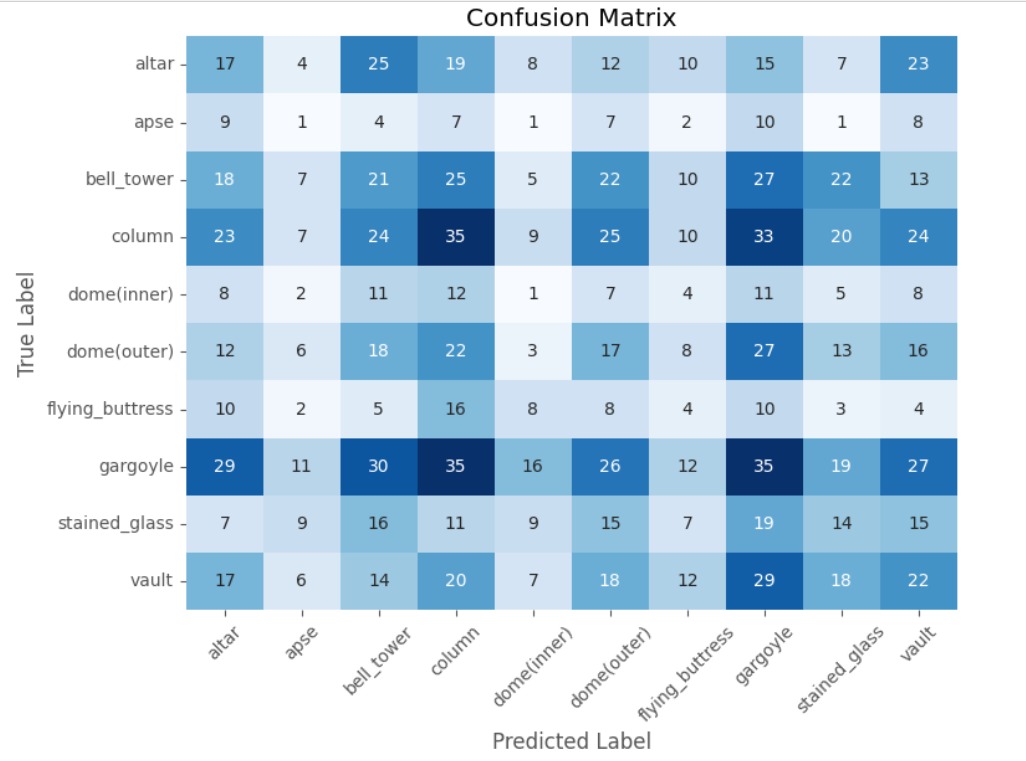


**Running Epochs with accuracy of 0.95 atlast**

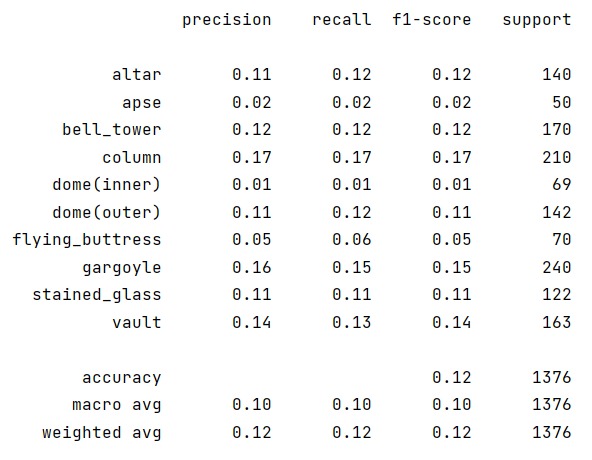


**Graph for validation and testing accuracy and loss**

\



**Confusion Matrix**



**Classification Report**

**Conclusion:**Concluding the works with the accuracy of minimal which must be fine-tuned to get better results to reach the overall better accuracy.

**Report for 26.06.2023**

**Introduction:**

Day-8 Internship has various takeaways and new concepts were made to know.

**Activities assigned:**

* Wanted to Complete Documentation for that week which is assigned previously.
* Need to Fine tune any one of the projects and need to make a detailed report on it.

**Activities Completed:**

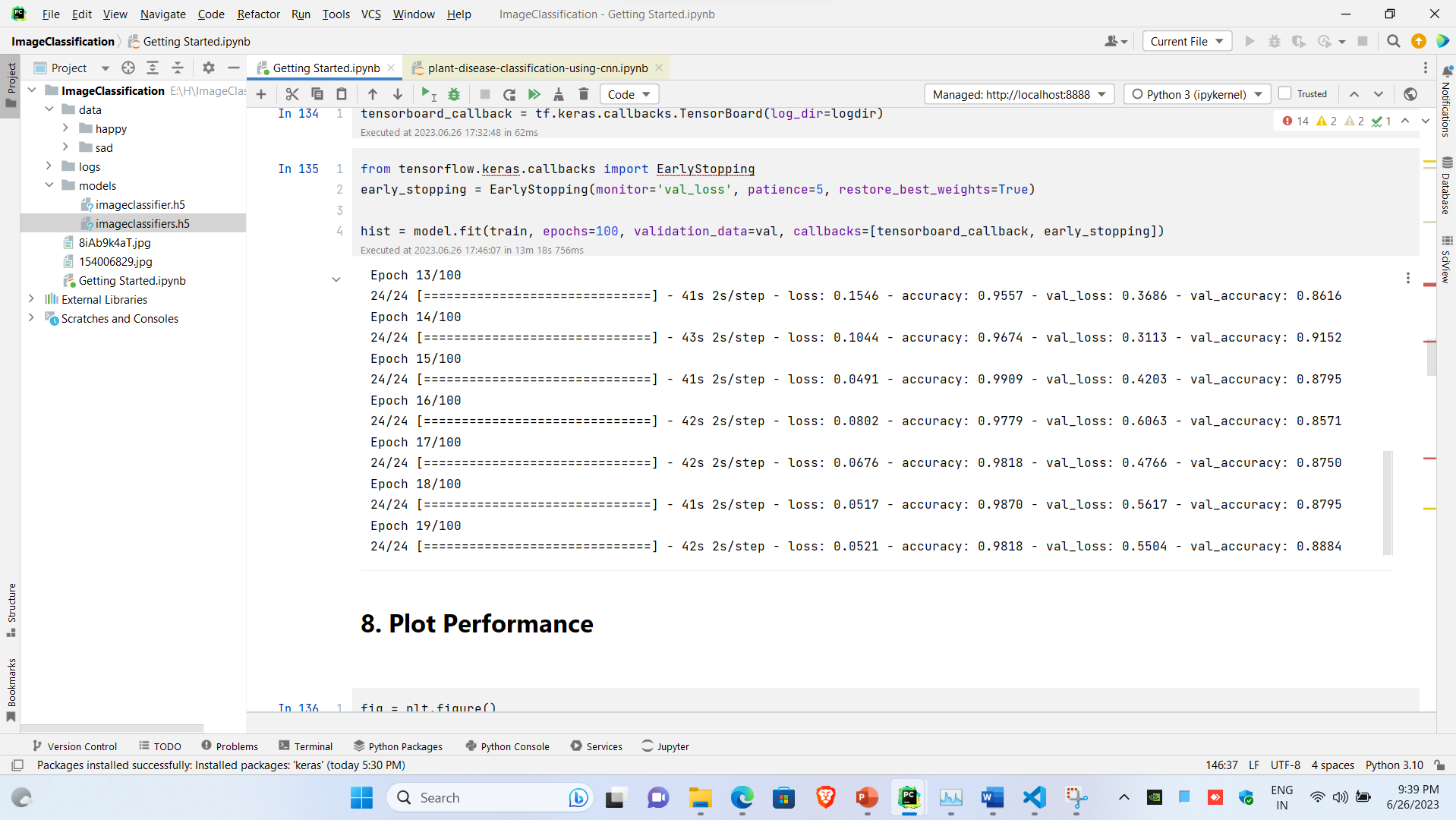
* Taken the Happy Sad image classification which is binary classification and made a CNN to work on it.
* First I had ran it on basic CNN and with VGG19 (currently running) model.
* Basic CNN ran faster and got an overall Test accuracy of 0.81.

**Challenges & Doubts:**

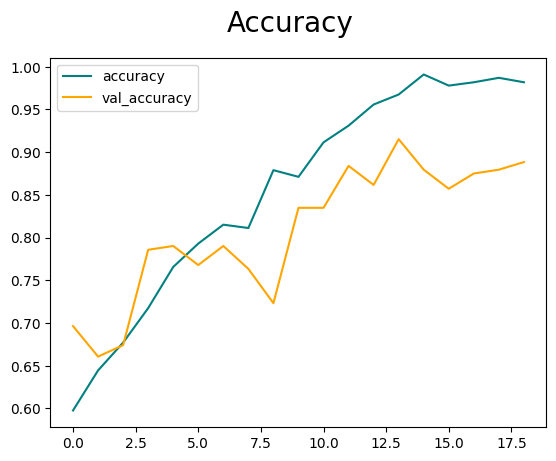
* It wasn’t a challenge as it ran successfully but didn’t got expected results as it needs to be still fine tuned to reach its accuracy as compared to previous test accuracies and graph it is quite good.
* It was difficult to stop at stage where accuracy has not increased so early stopping was introduced to train the model.
* Got to know about image recognition and its basics for annotation and all the basic ideas which are needed for image recognition like using image annotation tools and jobs for annotators and algorithms like Yolo which are used in real time applications.

**Screenshots for Day – 8**

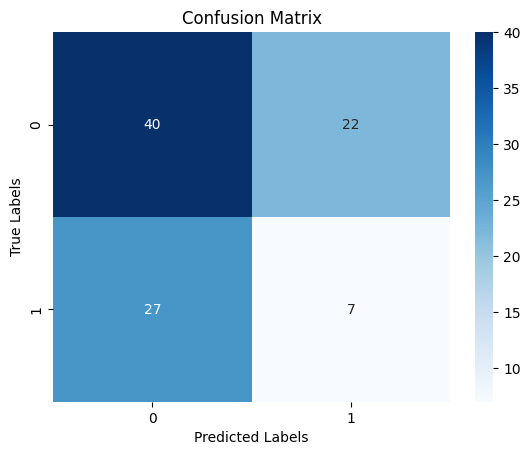
**BASIC CNN**



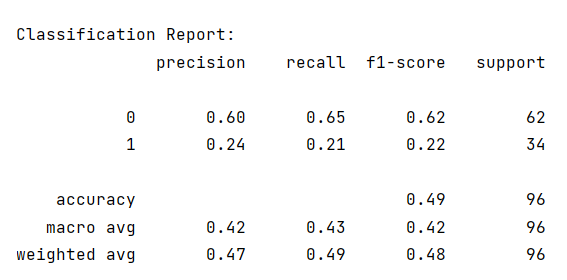
**Running Epochs**



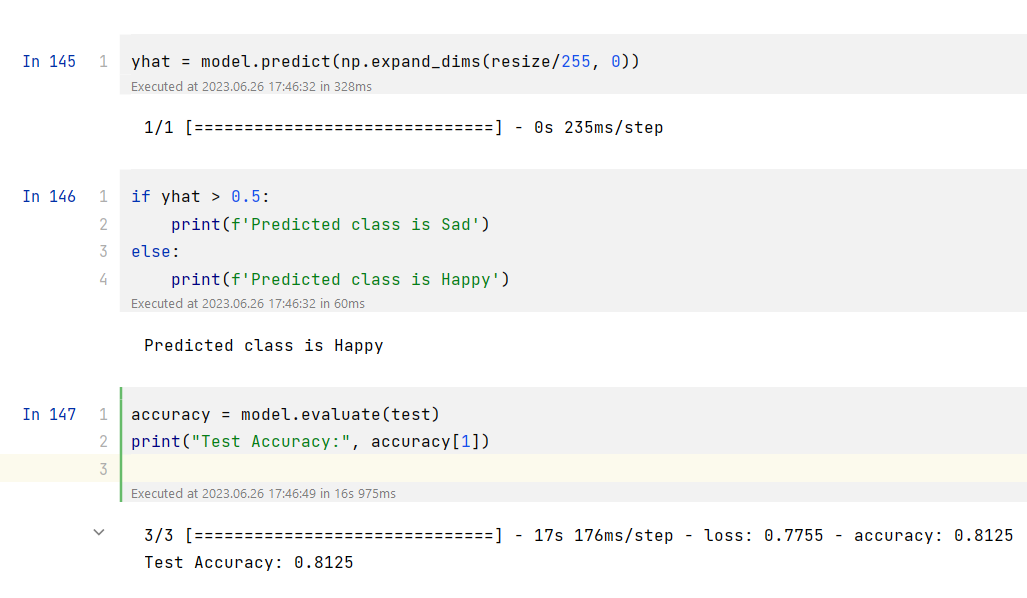
**Graph for validation and testing accuracy**



**Confusion Matrix**



**Classification Report**

****

**Test accuracy**

**Conclusion:**Concluding the works with the better accuracy for Happy or sad classification dataset and still working with vgg19 to get a better result and finetuning model.