Identify Skin Disorder using Dermoscopic Analysis through Federated Learning

In this project we have used YOLO8 to detect and extract skin disorder from uploaded image and then employ CNN algorithm to predict type of skin disease. To secure dataset we have used two clients to train on dataset and then each client will upload its trained weights to Federated server. Federated server will take average of all weights and then send to all requesting clients. To train all algorithms we have used HAM10000 dataset and then experiment with multiple algorithms such as CNN and VGG19. Among both algorithms CNN is giving best accuracy so we have used CNN to detect skin disorder.

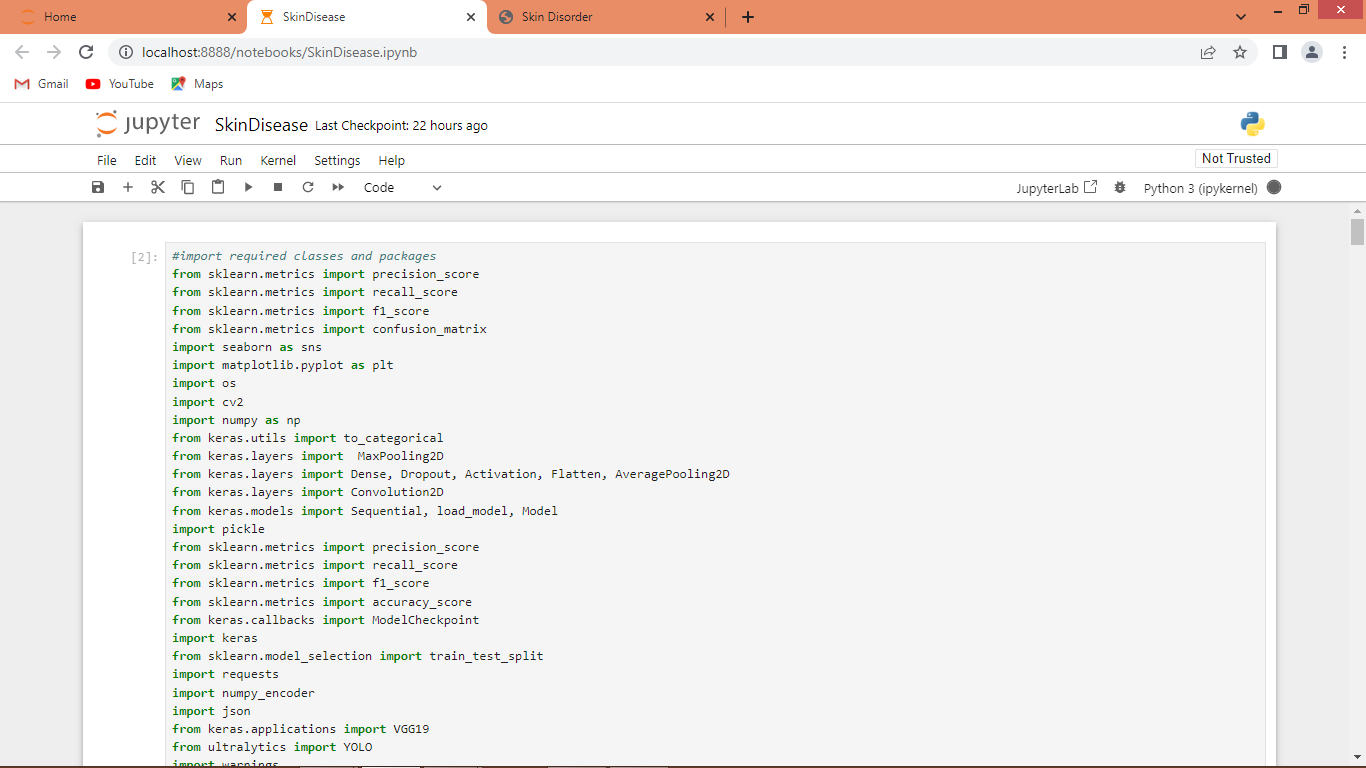
For algorithm training and testing we have used JUPYTER notebook and for prediction we have designed FLASK based web application

SCREEN SHOTS

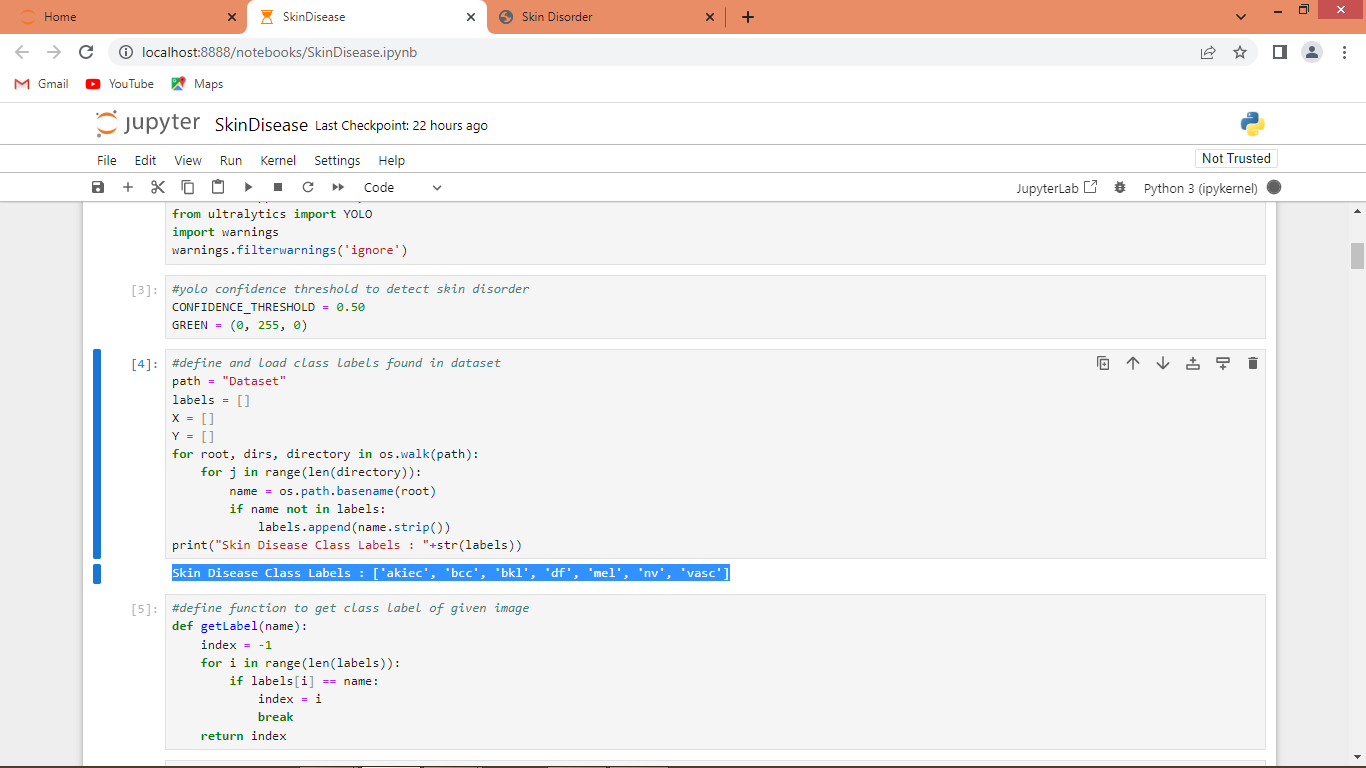
First double click on ‘runFLServer.bat’ file to start federated server and get below page



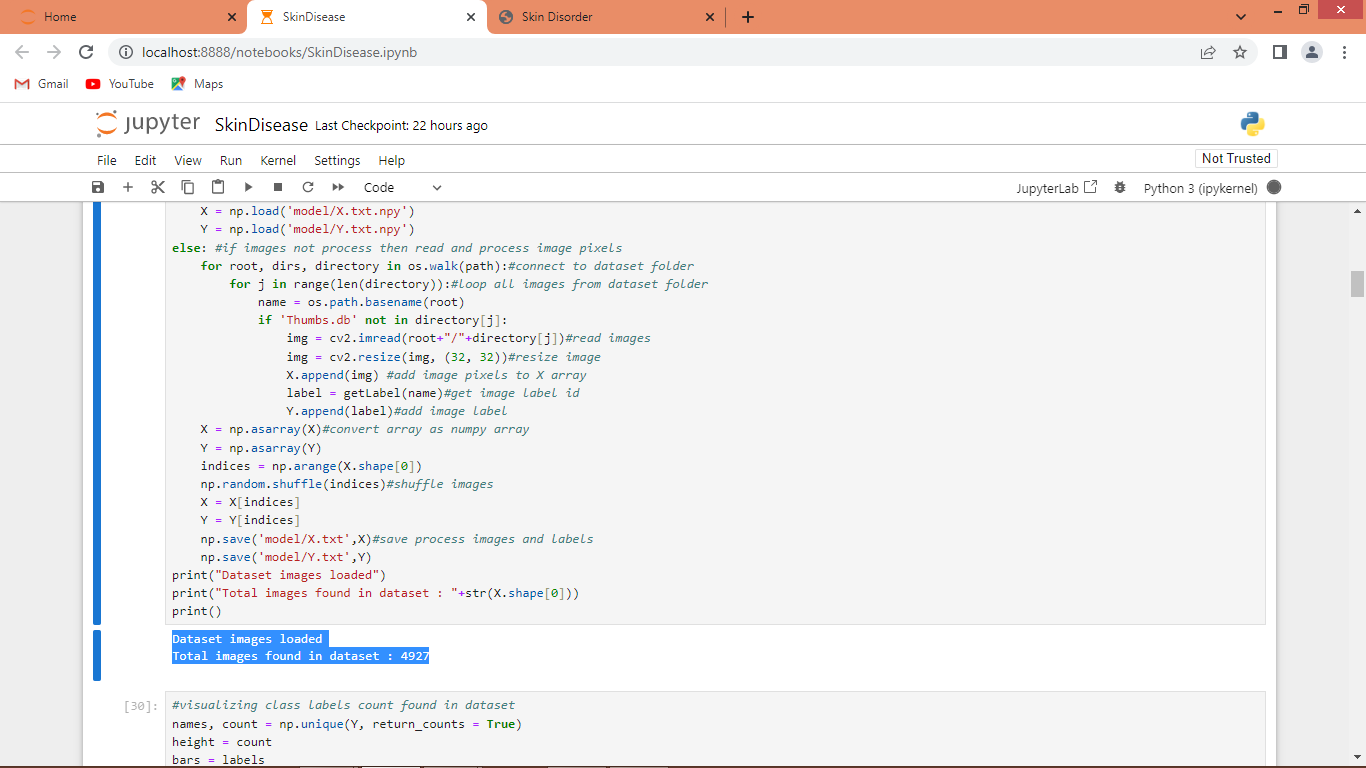
In above screen federated server started and now double clicks on ‘run.bat’ file to start JUPYTER notebook and will get below page



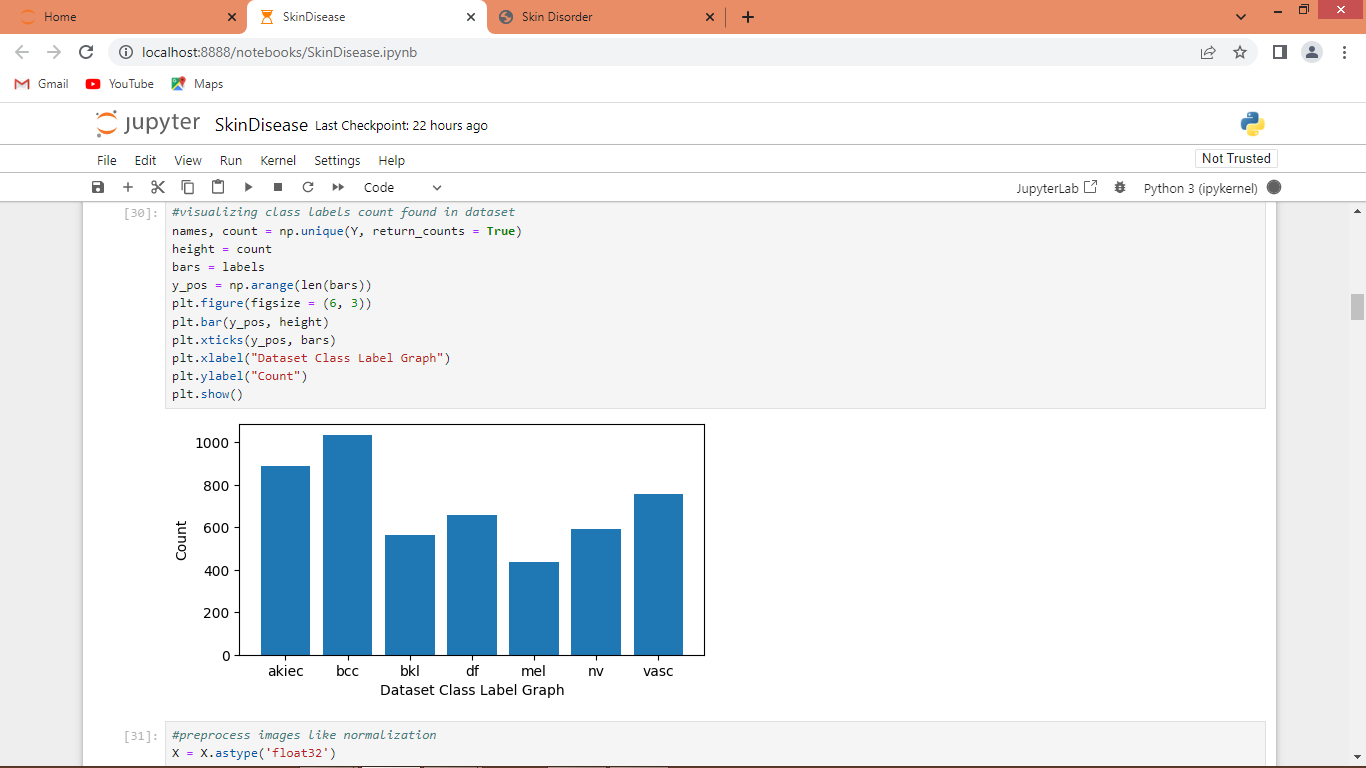
In above screen importing required packages and classes



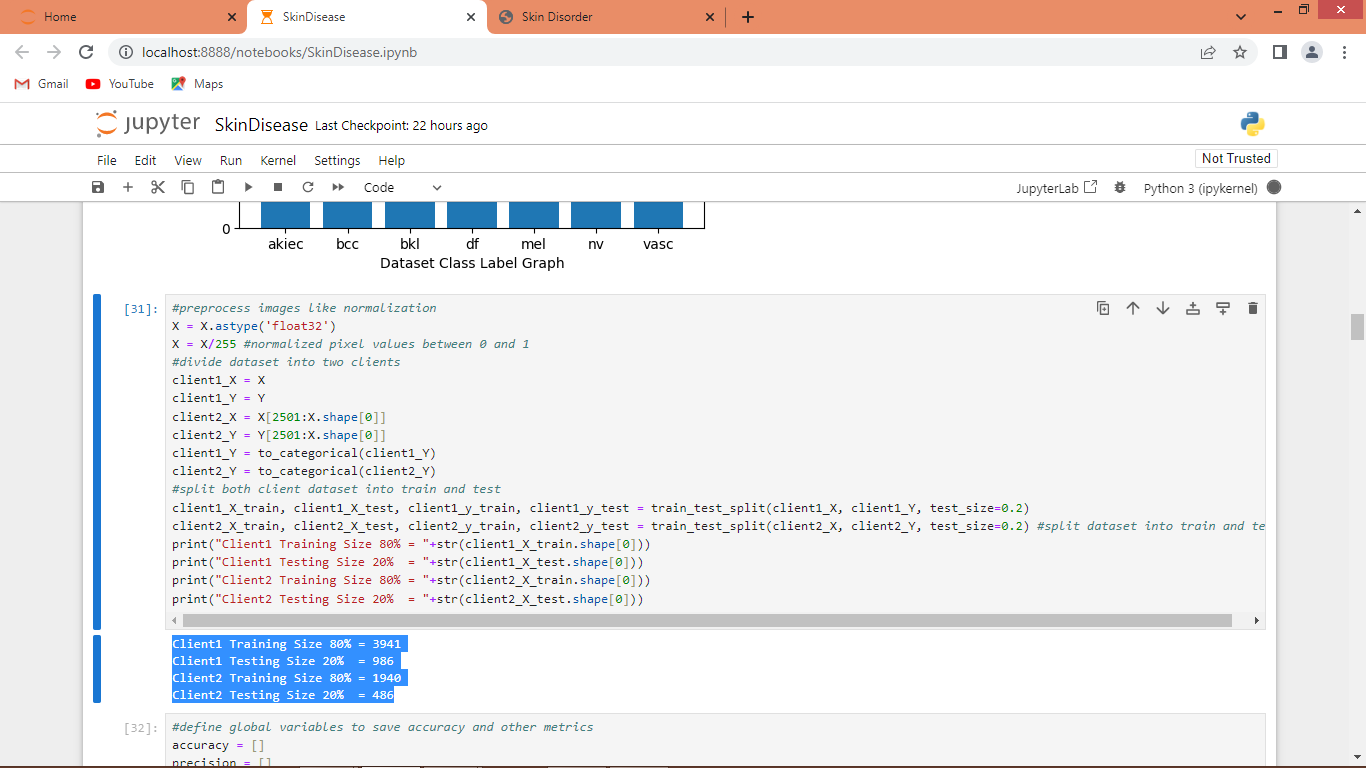
In above screen defining code to get labels of different skin disorders available in dataset and then in blue colour text can see all skin disorders labels



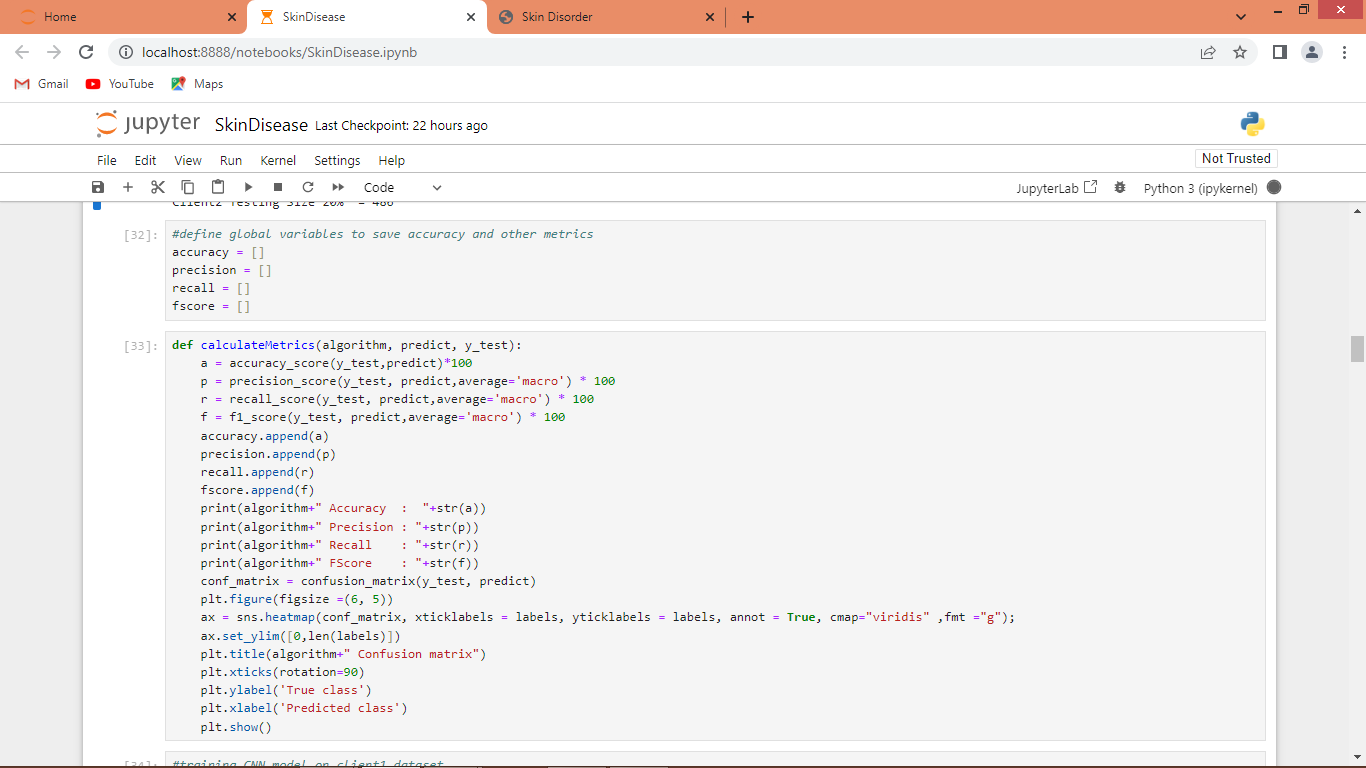
In above screen looping and reading each image from the dataset and then applying processing techniques such as resize and then extracting features and labels from each image and then adding X and Y training array and in blue colour text can see total number of loaded images



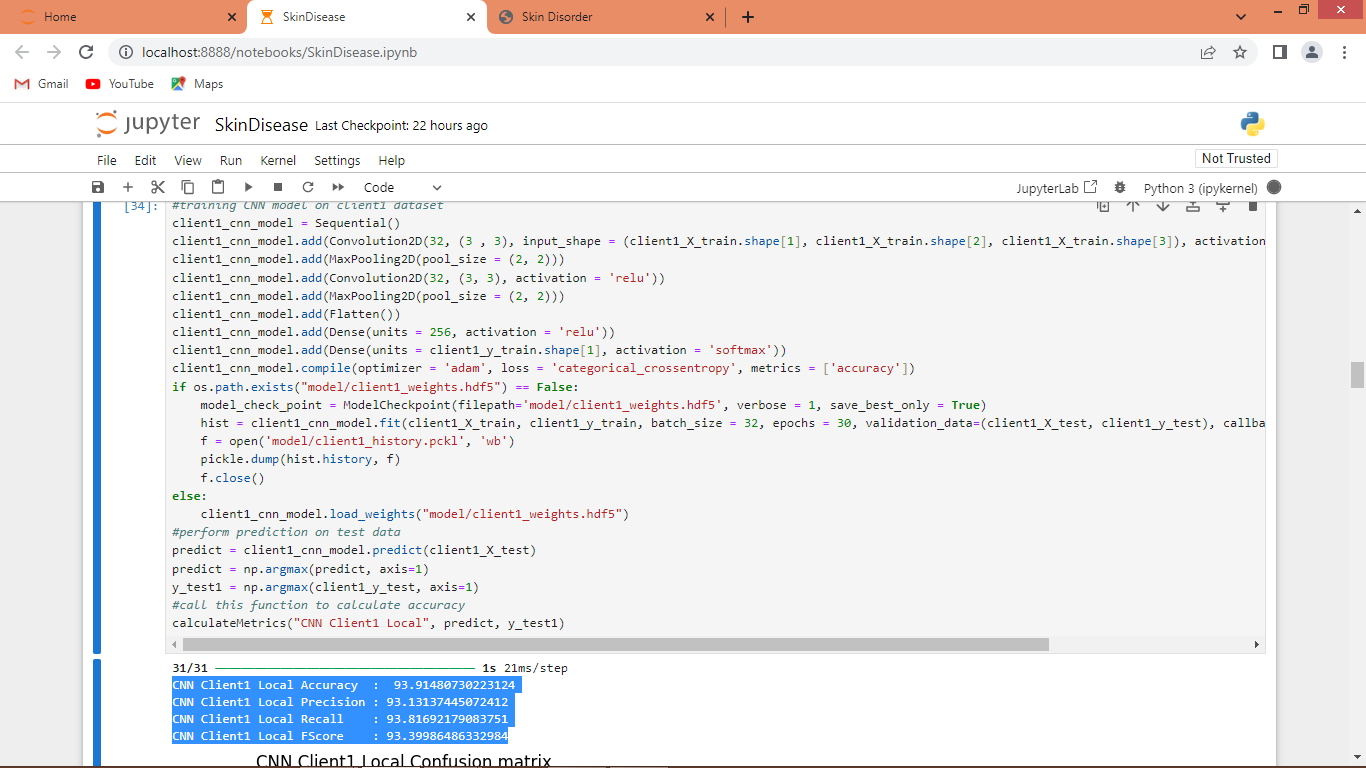
In above graph visualizing different class labels graph where x-axis represents ‘Skin Disorder’ class label and y-axis represents number of images



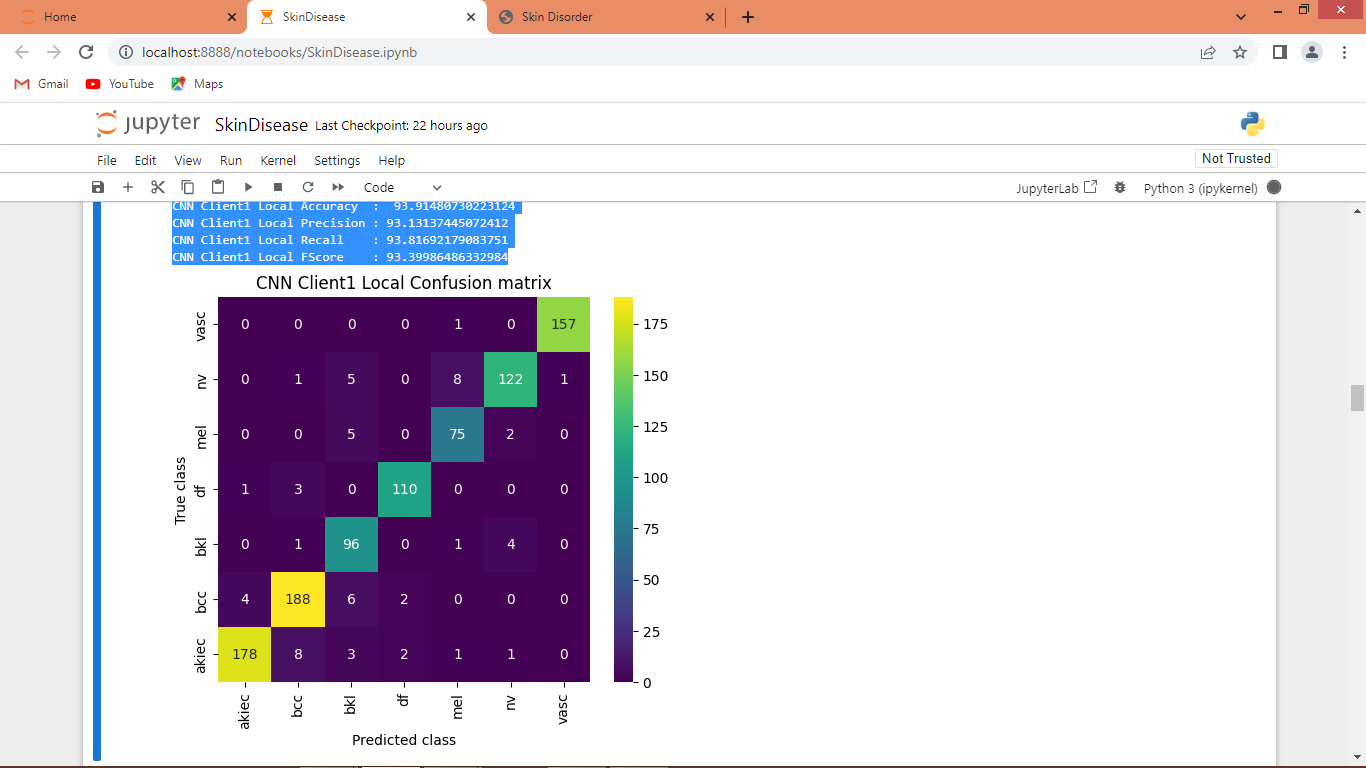
In above screen for client 1 we are taking full dataset and then for client2 we are taking half dataset and then splitting both datasets into train and test and then blue colour text can see train and test size for each client



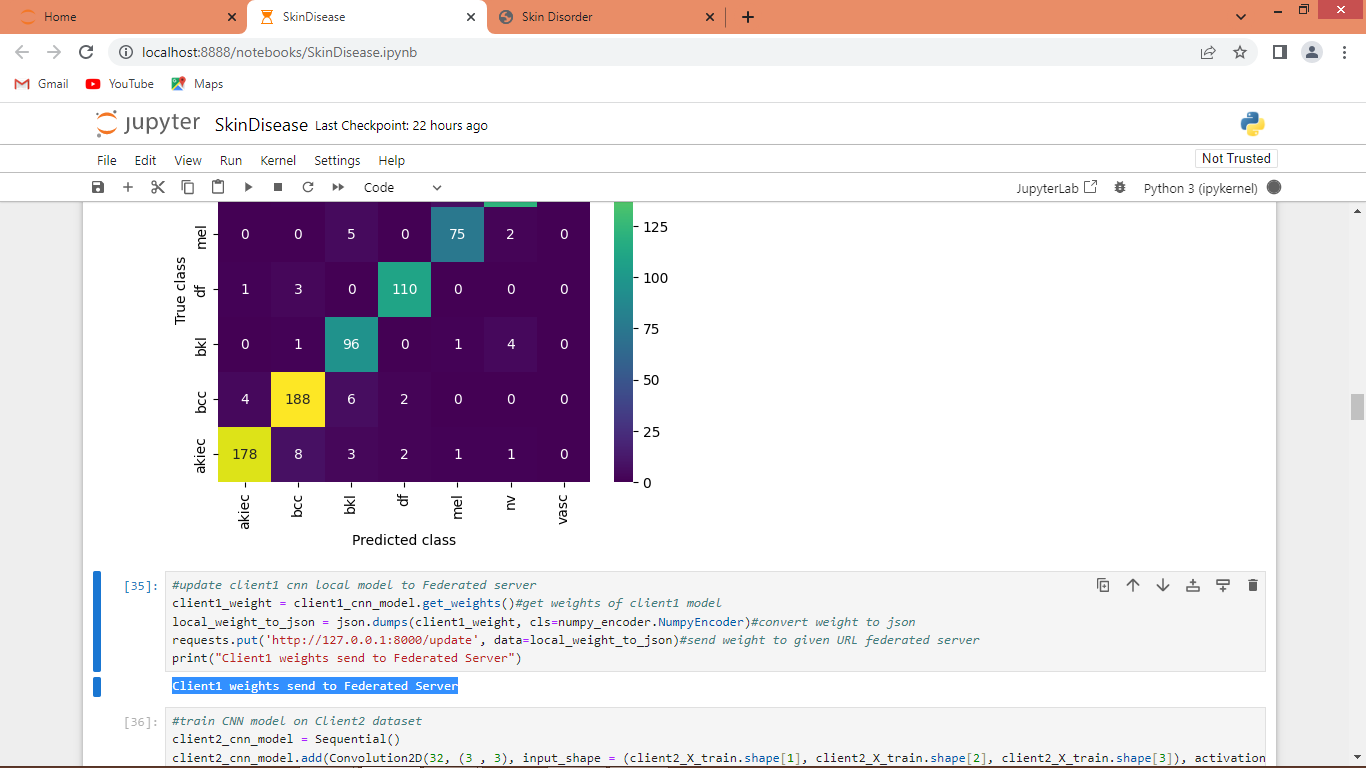
In above screen defining function to calculate accuracy and other metrics



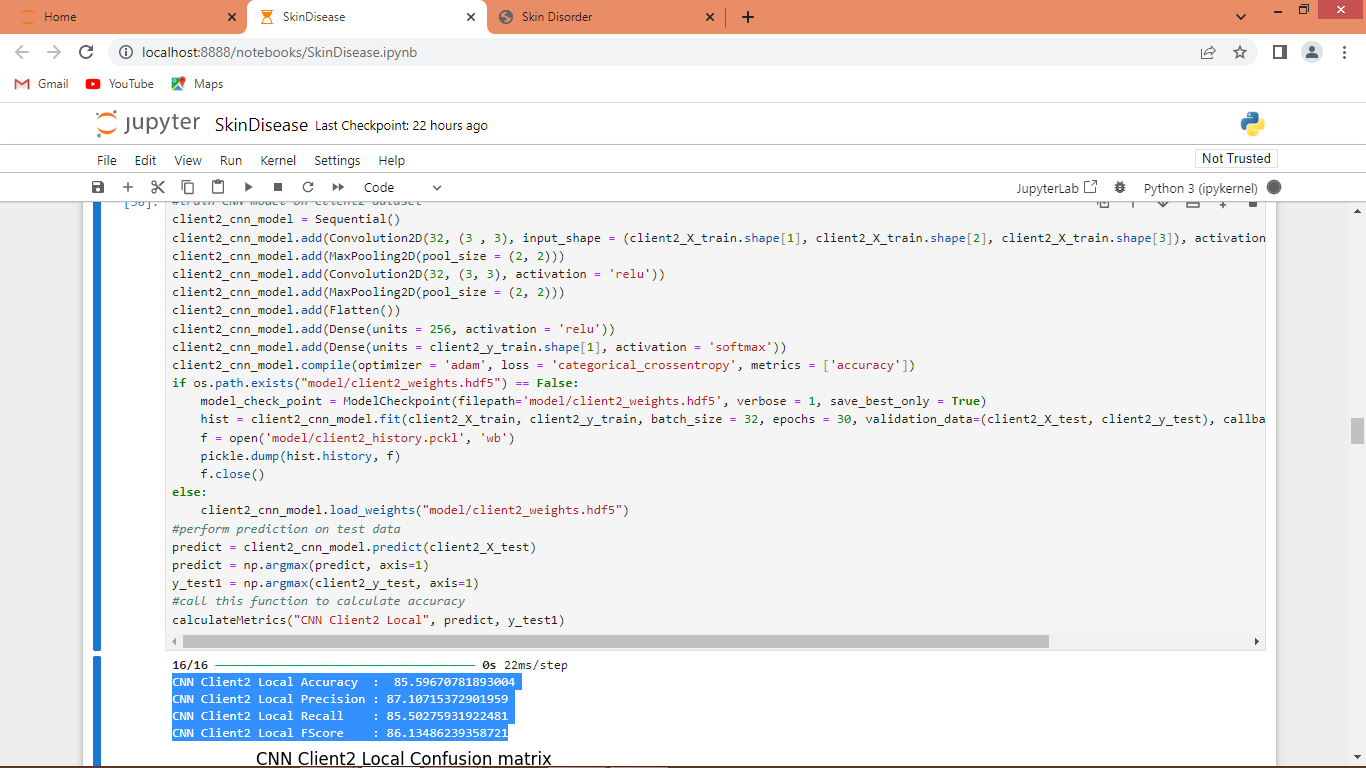
In above screen defining CNN for client1 which will get trained on training images and then its accuracy will be calculated based on prediction of test data and then in blue colour text can see CNN on Client1 got 93% accuracy and can see other metrics like precision, recall and FCSORE. In below screen can see confusion matrix graph



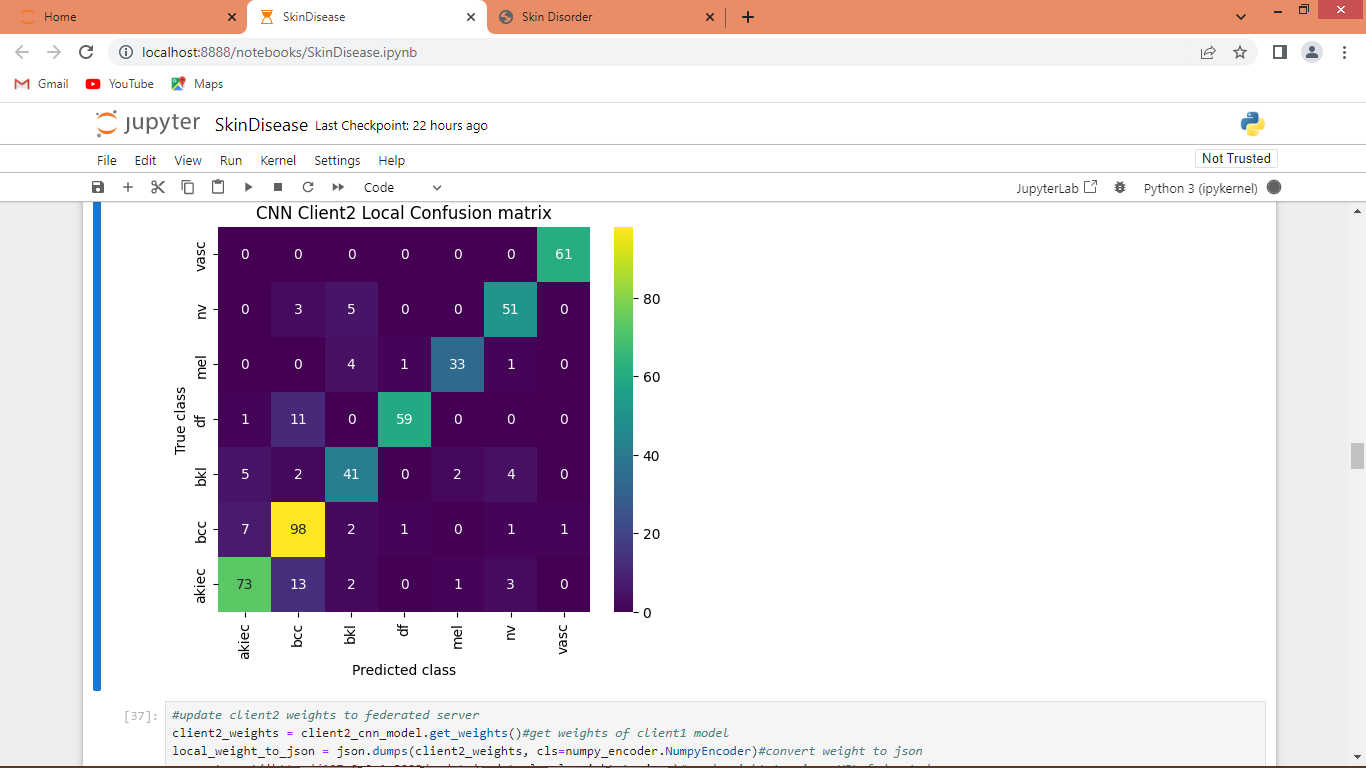
In above confusion matrix graph x-axis represents predicted class labels and y-axis represents True class labels and then all different colour boxes in diagnol represents correct prediction count and remaining blue boxes got incorrect prediction count.



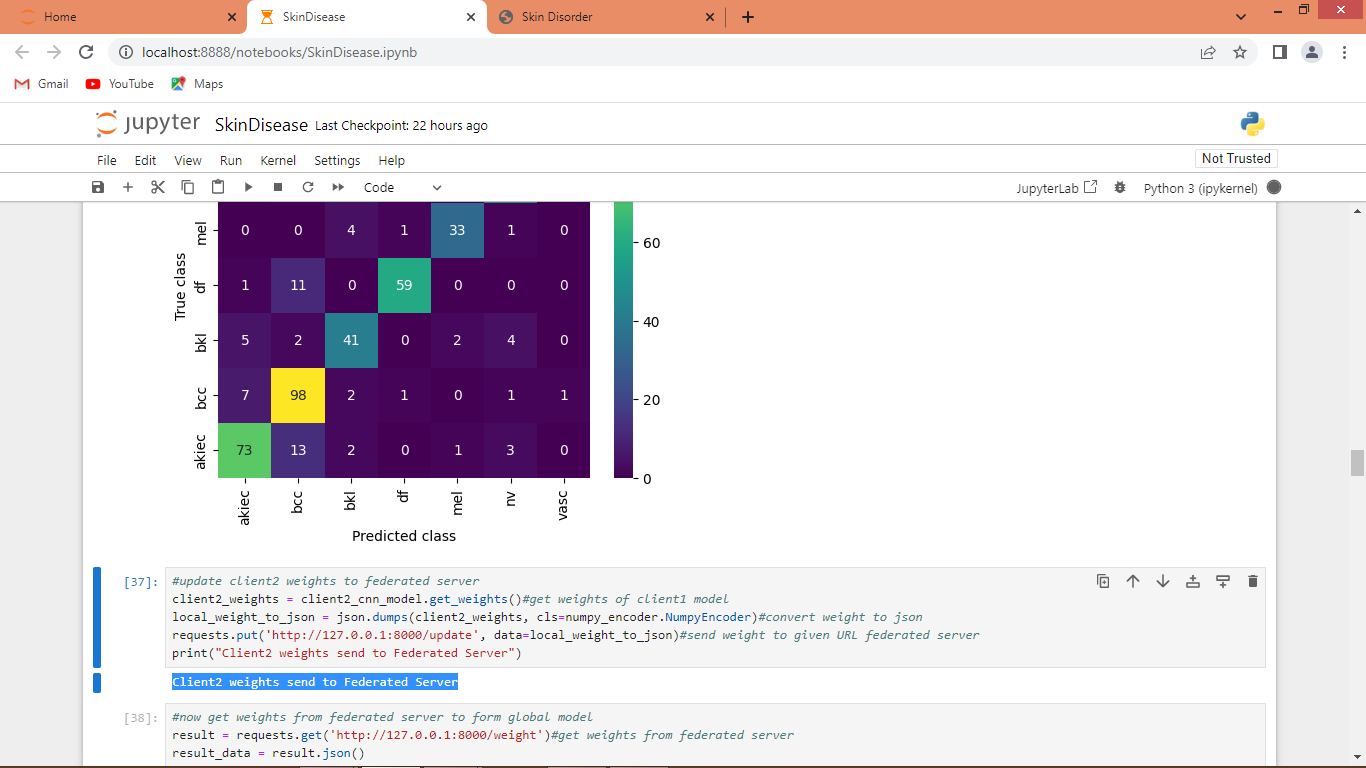
In above screen extracting weight of client1 CNN model and then sending that weight to federated server running on given URL



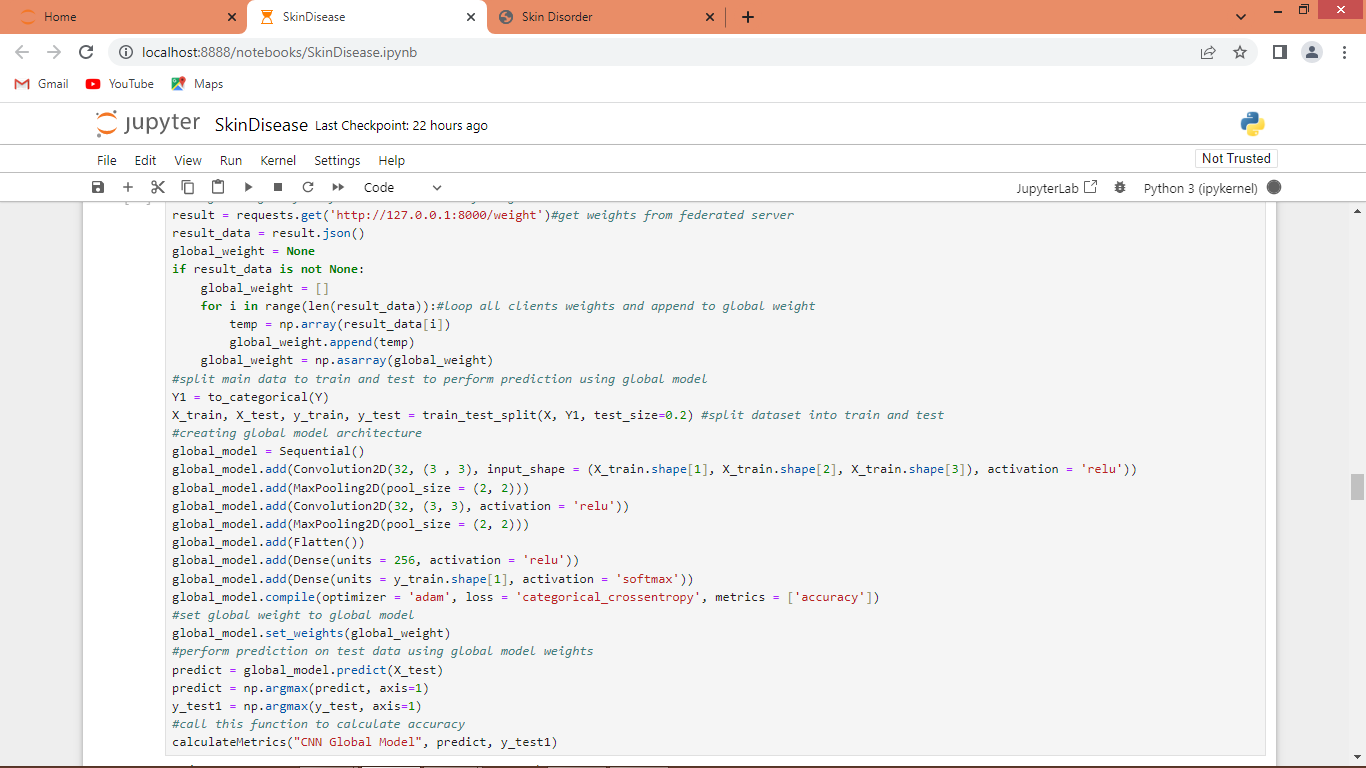
In above screen training CNN client2 model using another dataset and then in blue colour text can see client2 model got 87% accuracy



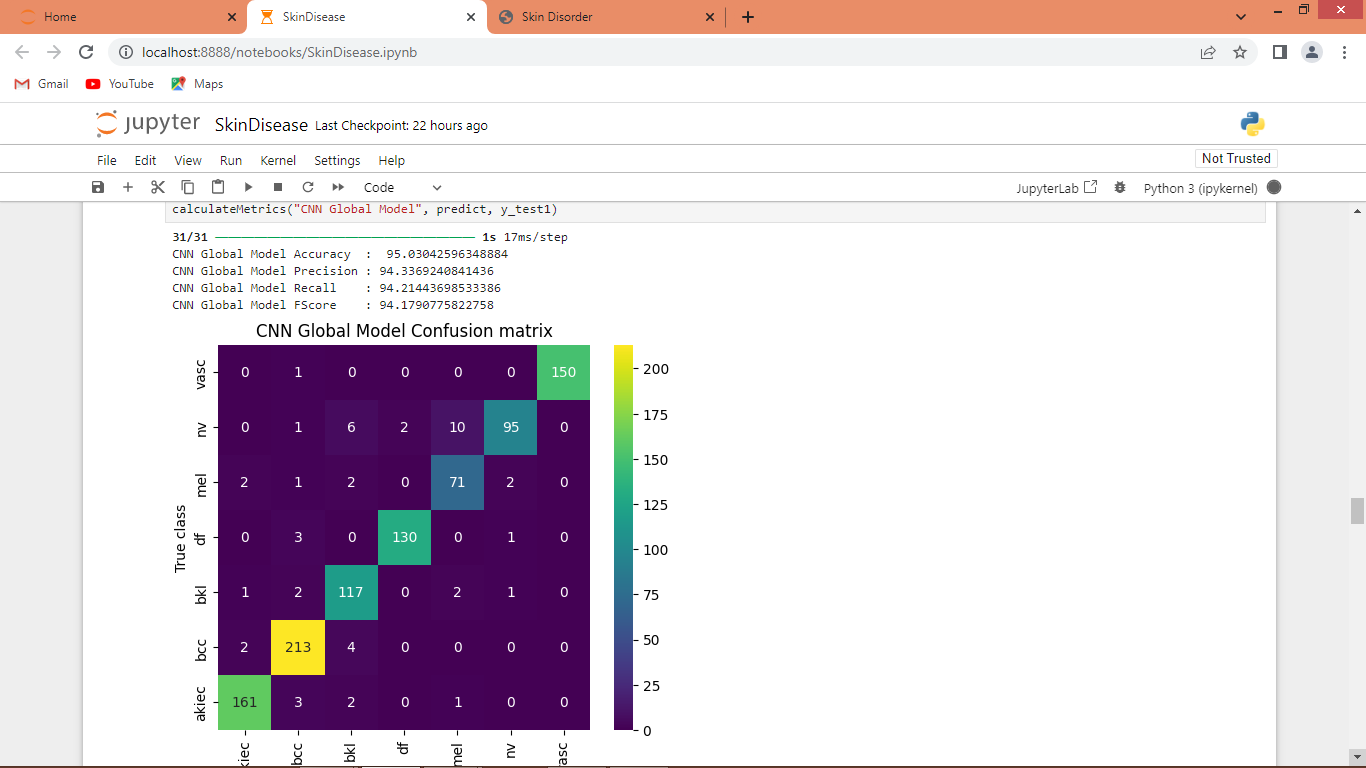
In above screen can see CNN client2 model confusion matrix graph



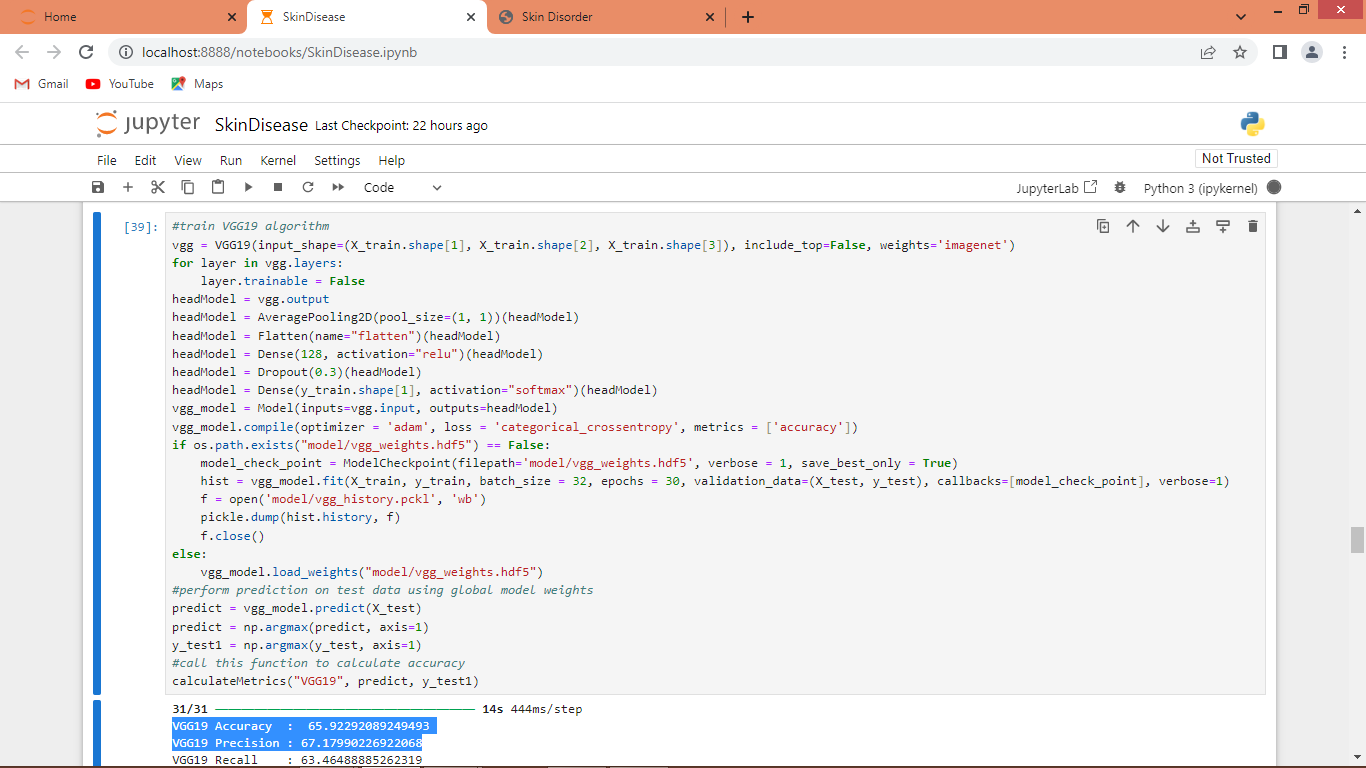
In above screen extracting weight of CNN client2 model and then sending to Federated server URL



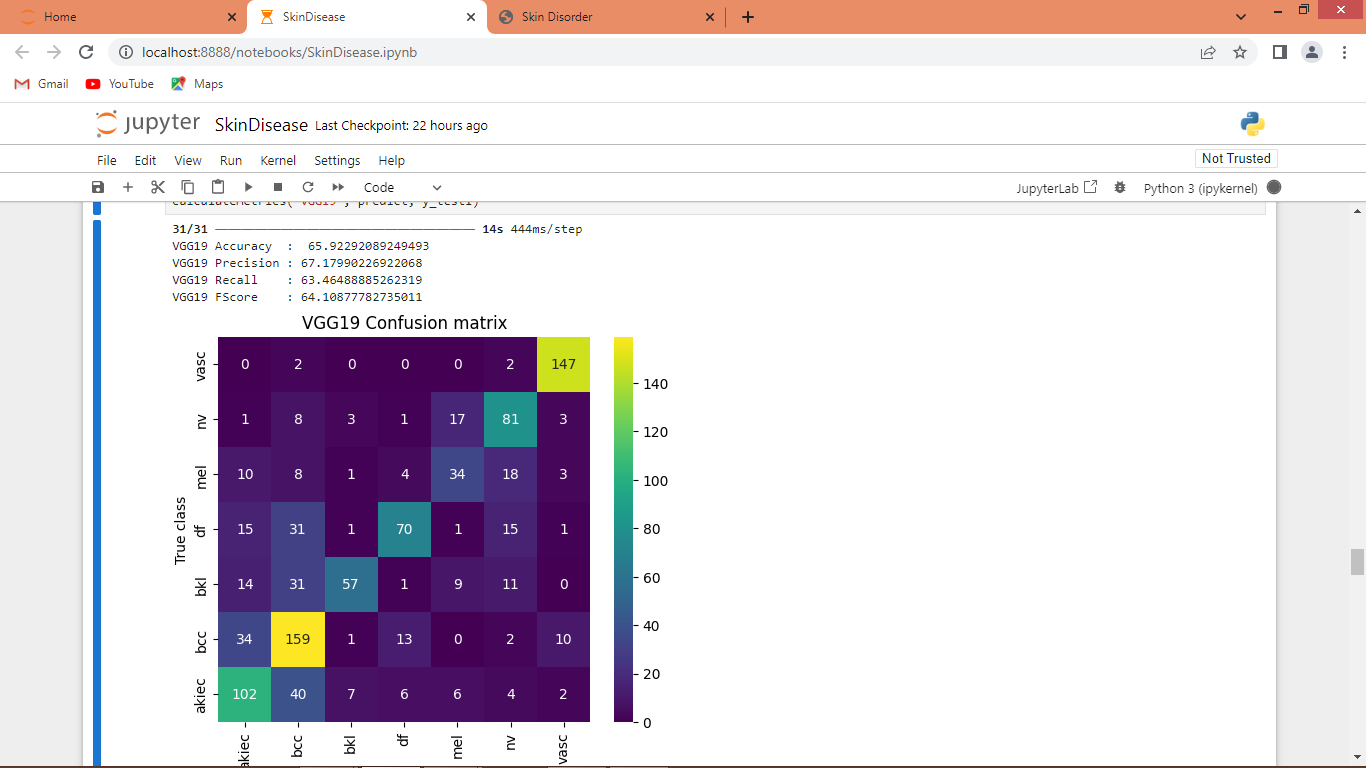
In above screen collecting weights from federated server and then generating Global model and after executing above model will get below weight



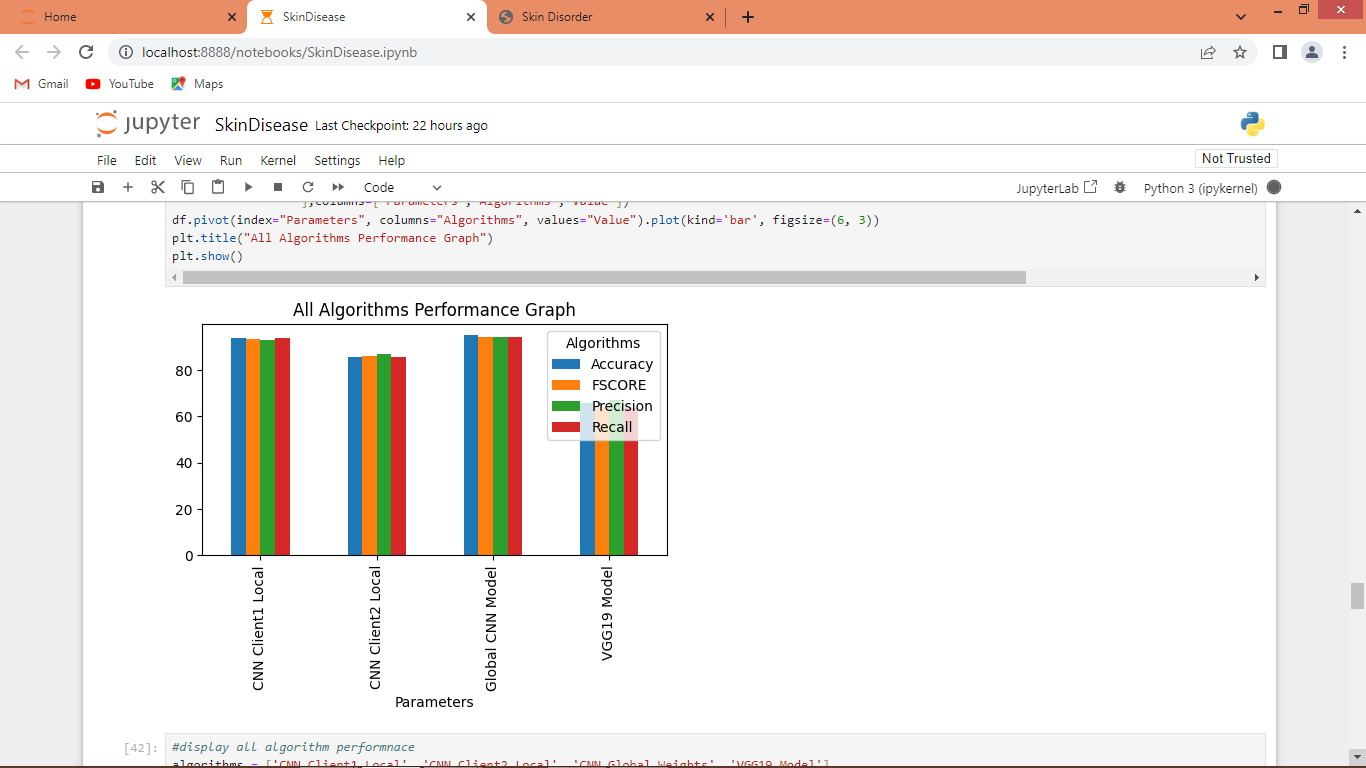
In above screen global model got 95% accuracy and can see other metrics and output for global model



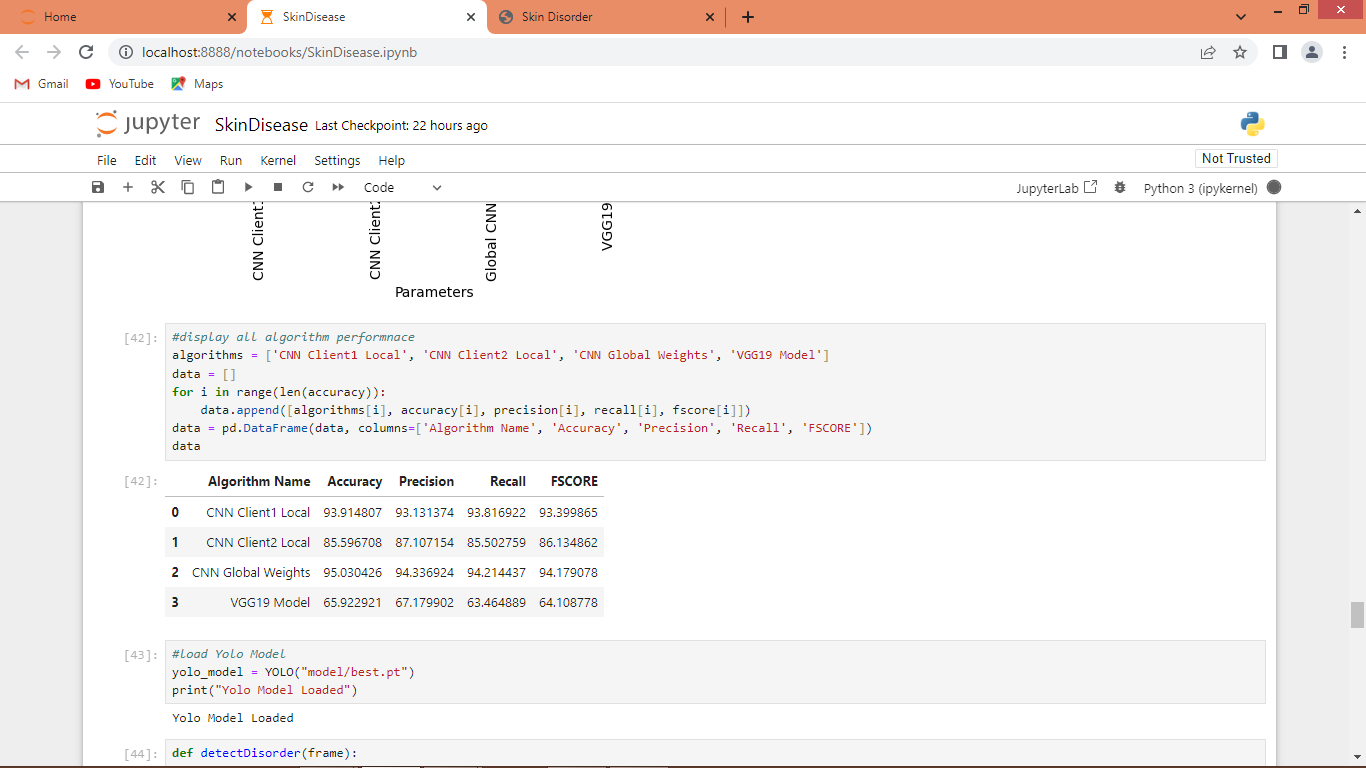
In above screen training VGG19 model and after executing this model we can see VGG19 got 65% accuracy and can see other metrics in below screen



In above screen can see VGG19 performance



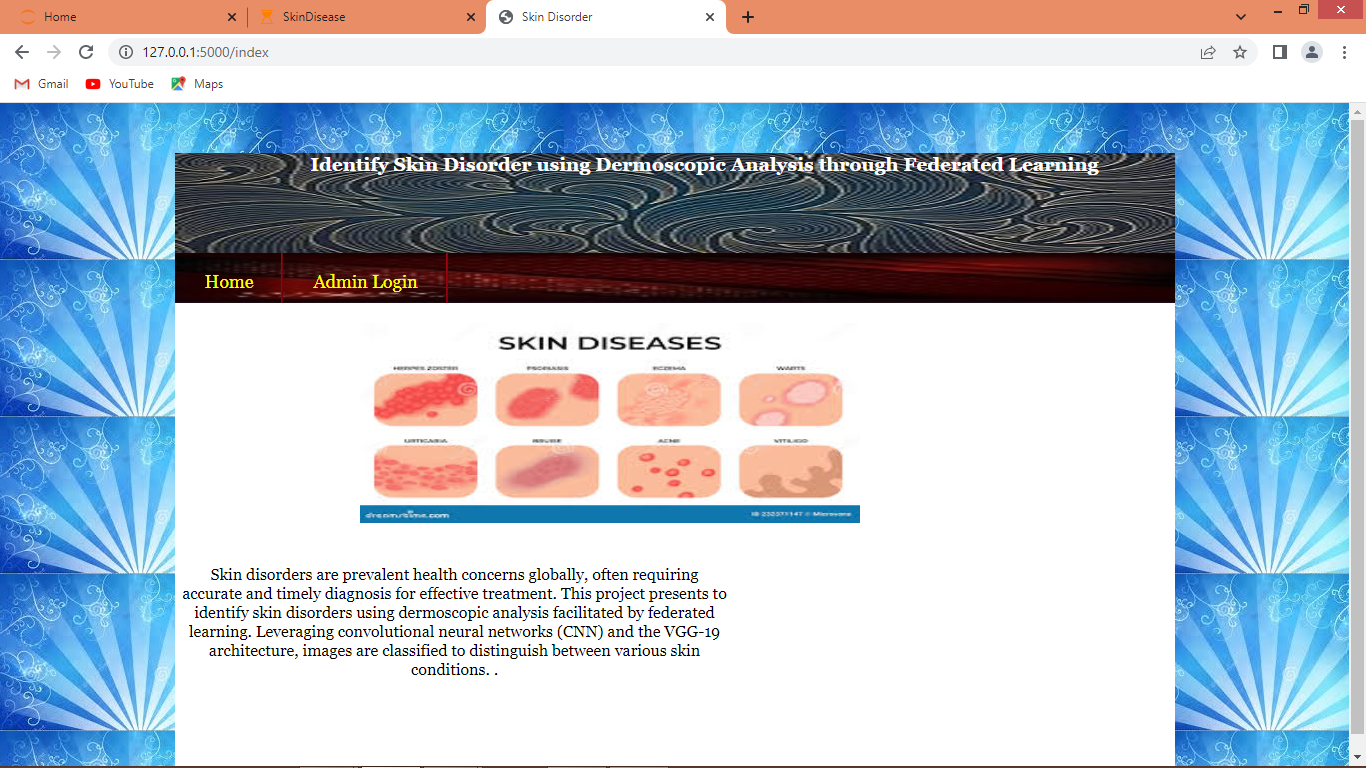
In above screen can see performance of each algorithm where x-axis represents algorithm names and y-axis represents accuracy and other metrics in different colour bars



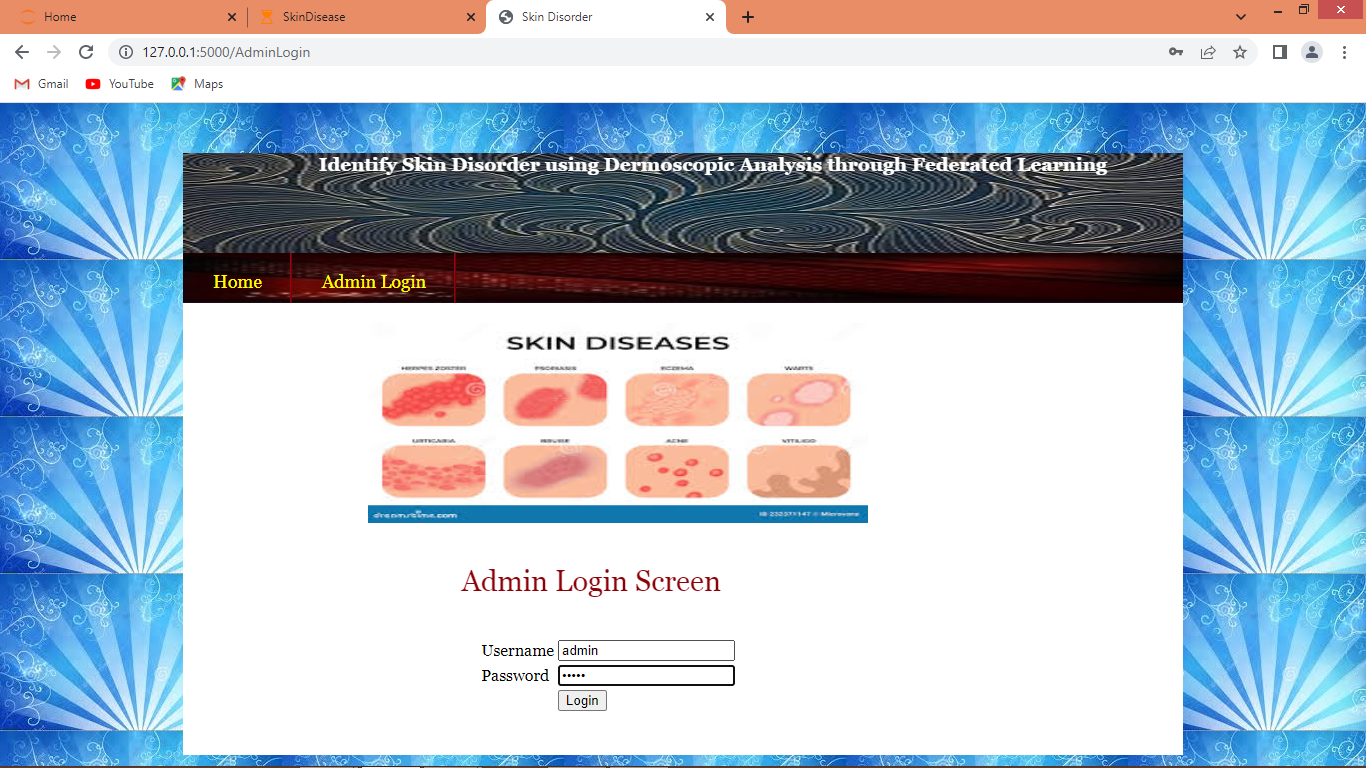
In above screen in tabular format can see performance of each algorithm



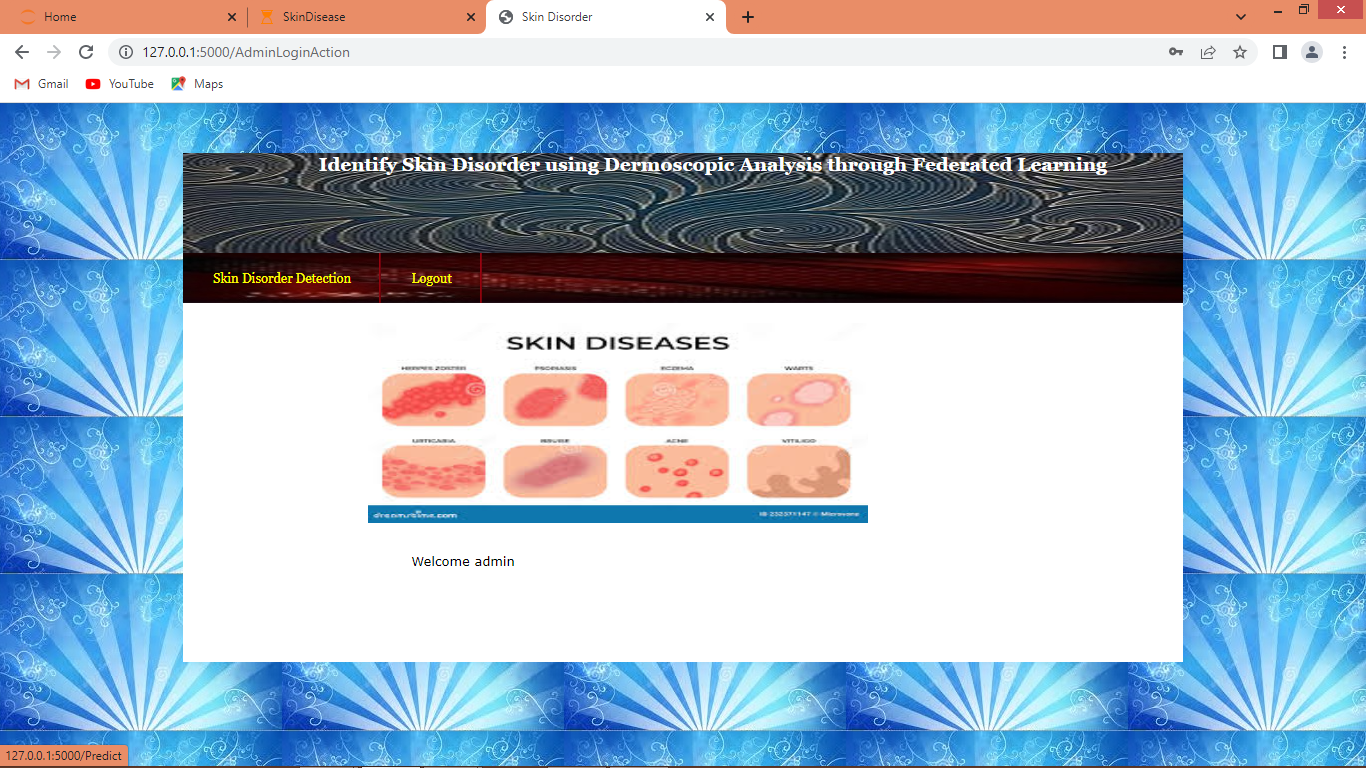
In above screen run FLASK code block to start flask server and then open browser and enter URL as <http://127.0.0.1:5000/index> and press enter key to get below page



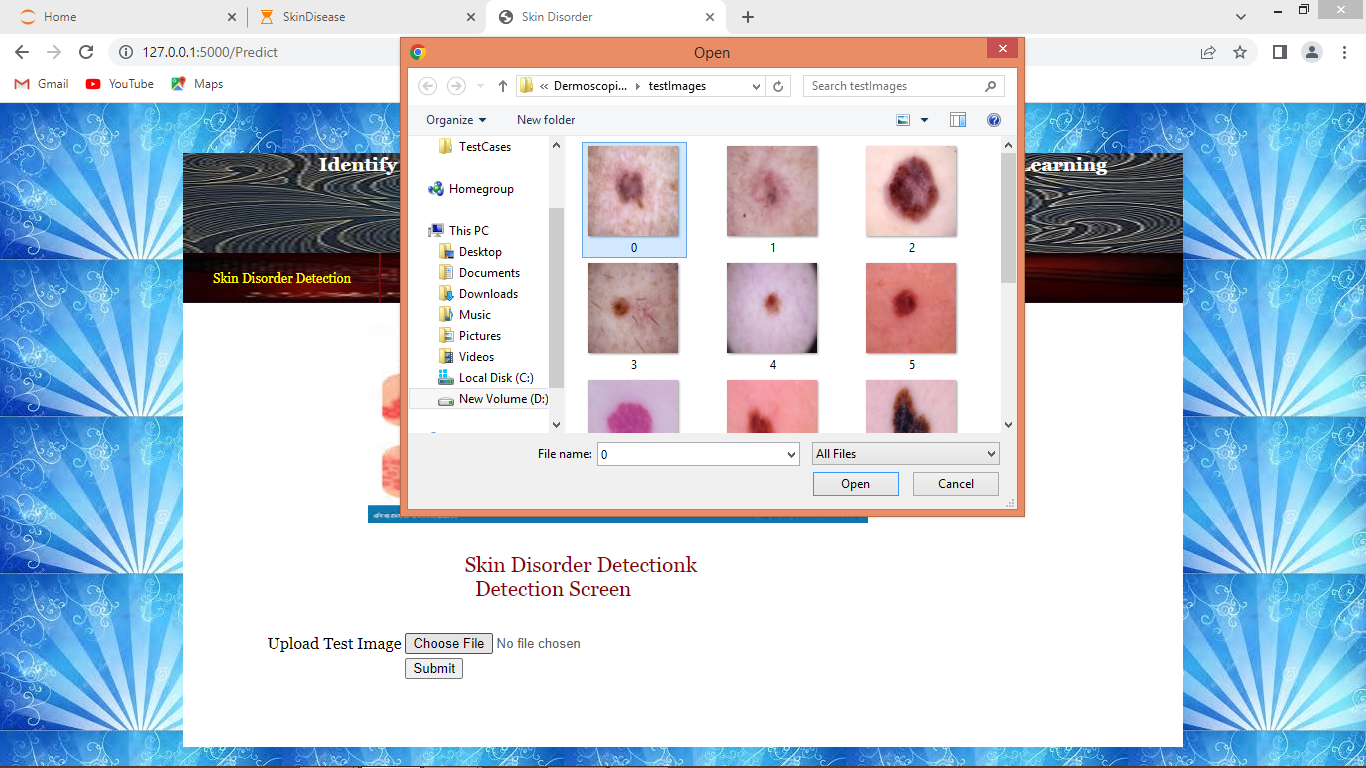
In above screen click on ‘Admin Login’ link to get below page



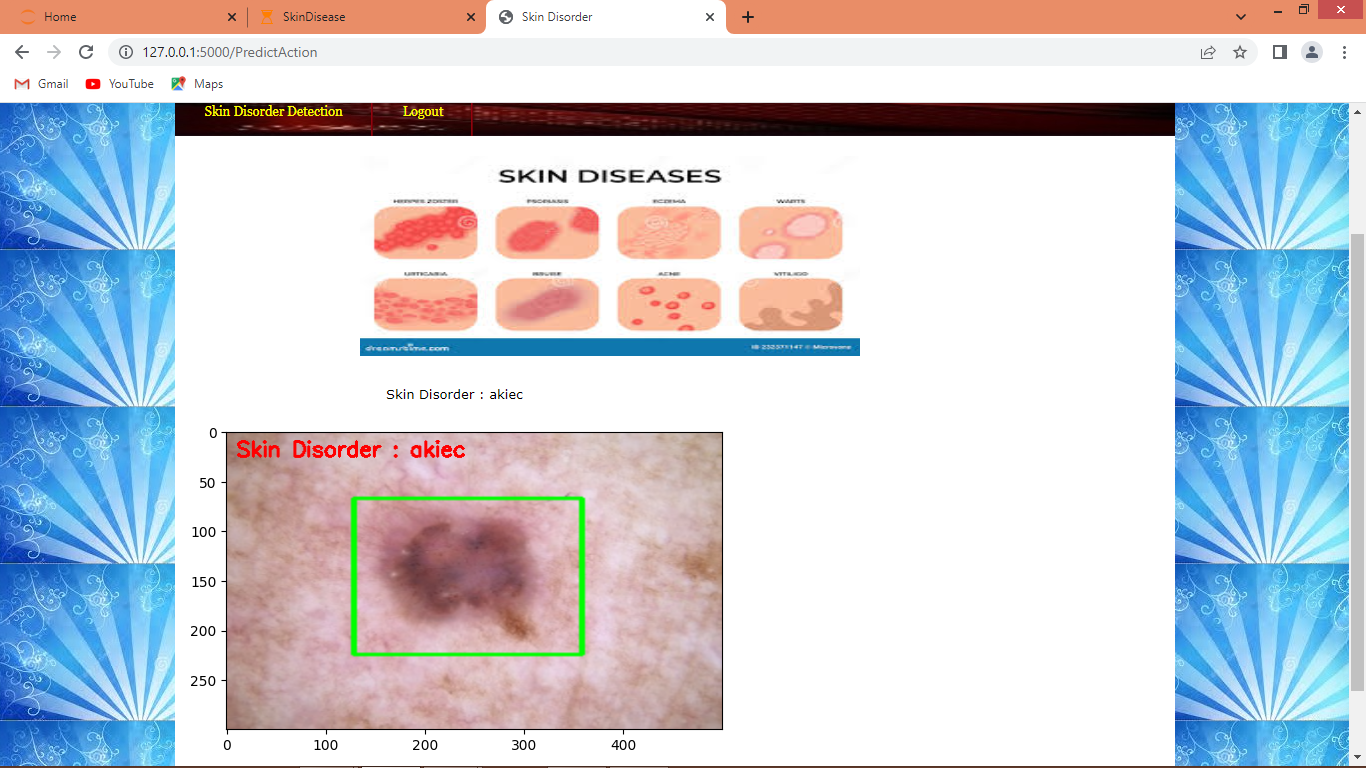
In above screen admin is login by using username and password as ‘admin and admin’ and then press button to get below page



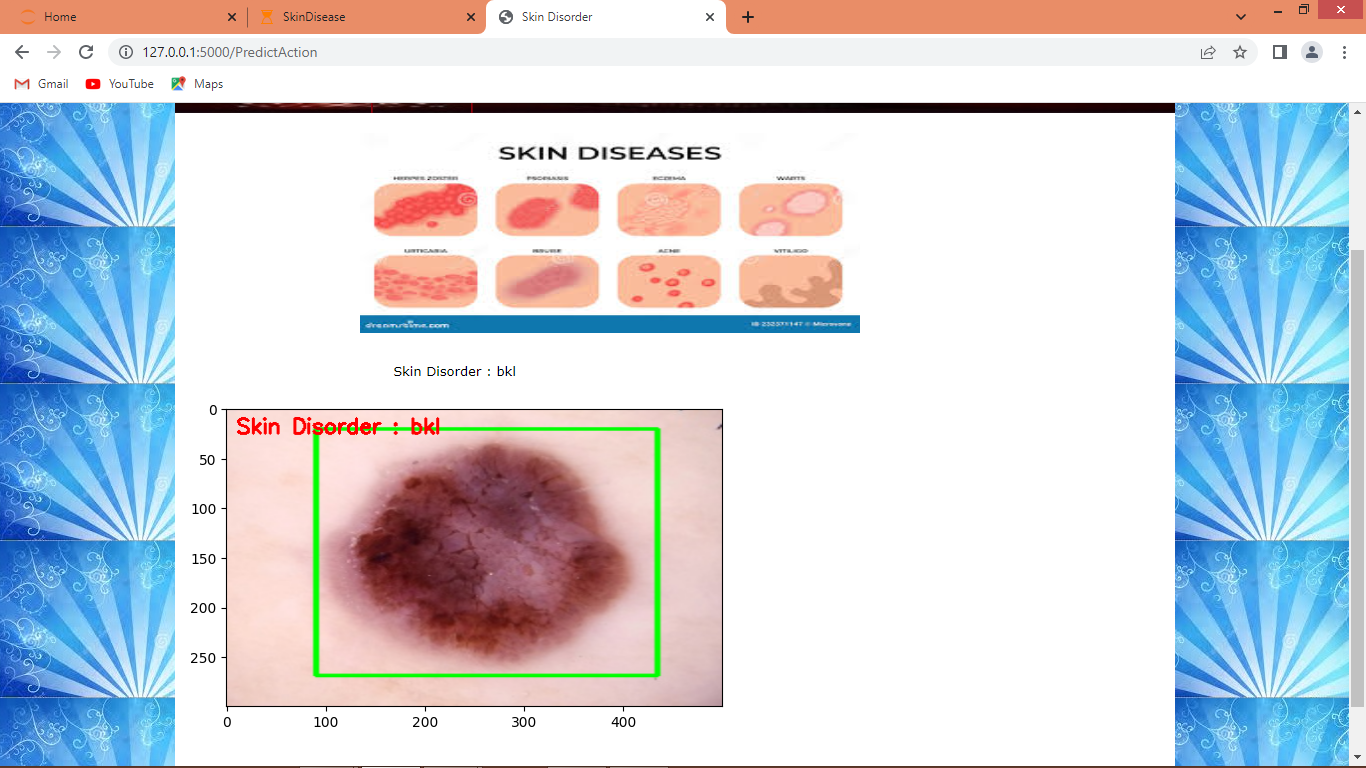
In above screen click on ‘Skin Disorder Detection’ link to get below page



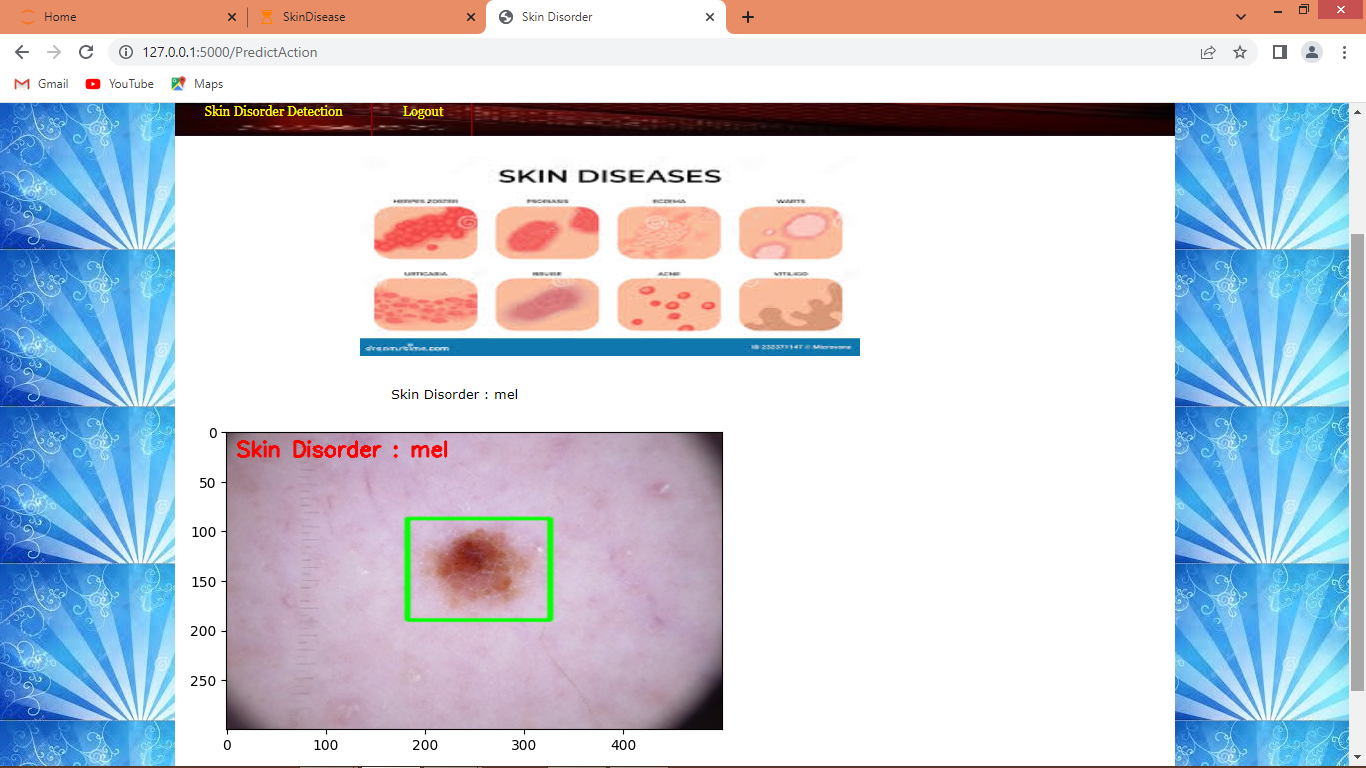
In above screen click on ‘Choose File’ button and then upload test image from ‘test Images’ folder available inside code folder and then press ‘Submit’ button to get below page



In above screen Yolo8 will detect skin disorder region and surrounded with green rectangle and then CNN will predict type of skin disorder and in above screen AKIEC disorder detected. Similarly you can upload and test other images and below are few other samples



In above screen BKL disorder detected



In above screen MEL disorder detected and similarly you can test other images also.