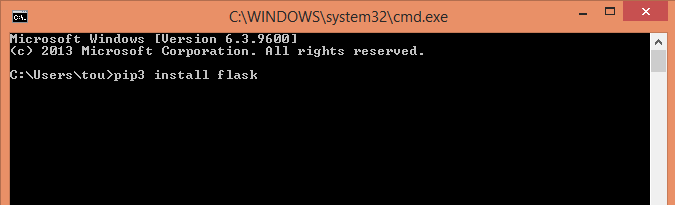
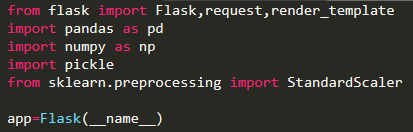
Creation of Web-API using Flask:-

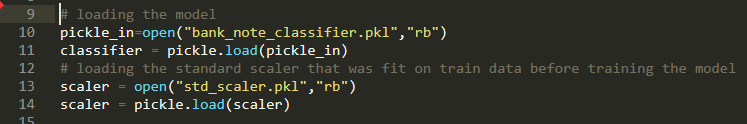


1. Install flask using the command: pip3 install flask



2. Import all necessary libraries required to run the app.

app = Flask(\_\_name\_\_) helps to run the app.



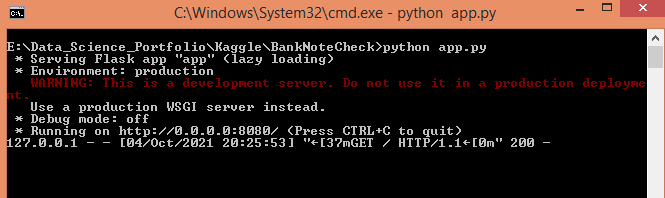
3. Loading the model pickle file and the standard scaler object (stored as a pickle file).



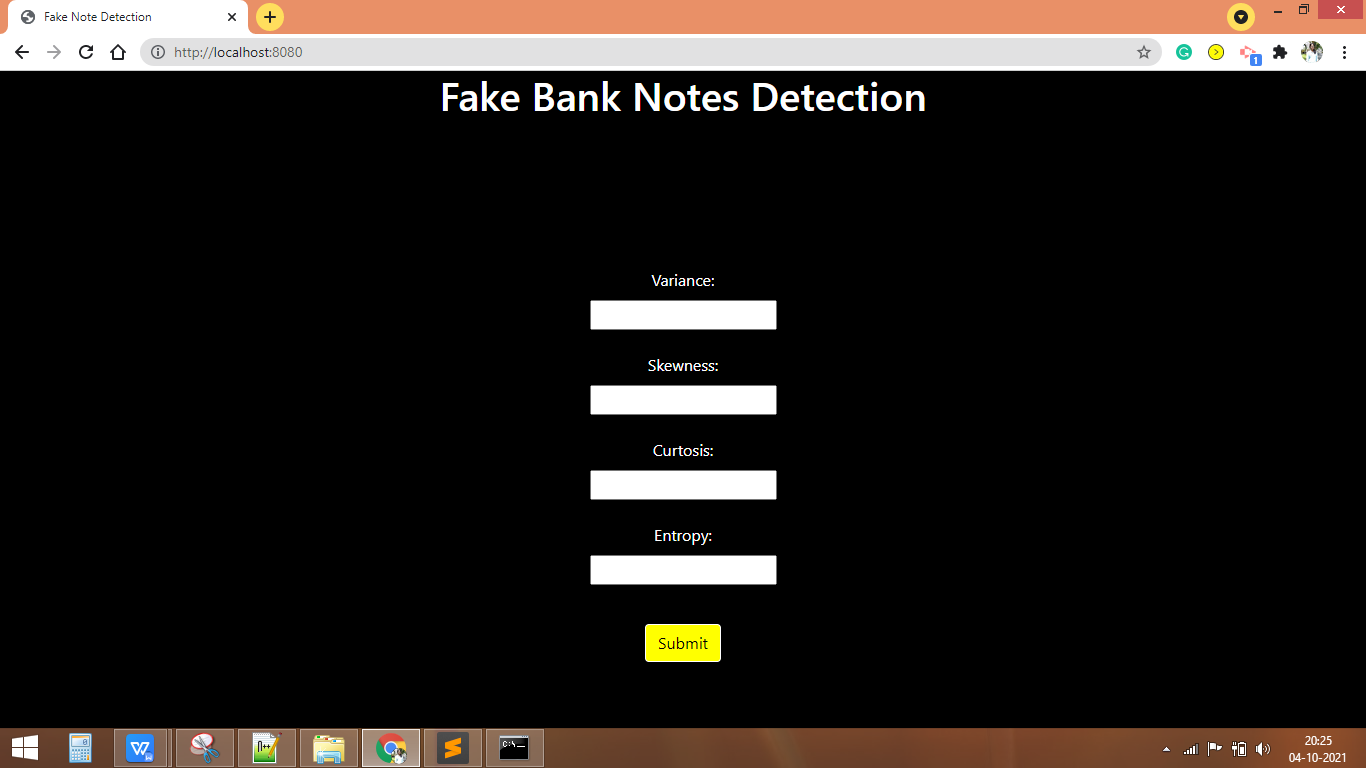
4. Define all relevant methods required to execute the model after deployment.

(Please refer the index.html file inside templates folder in the repository to see the code for UI.)

5. Run the web-app using the command: python app.py to see the app in browser:-



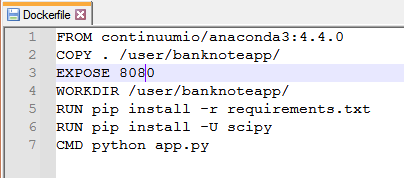
Note: The port number should match with the one used in app.py.



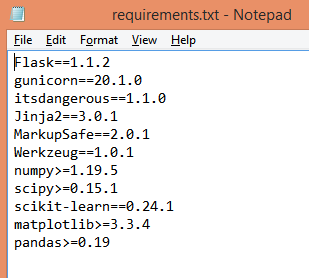
Containerization using Docker:-

1. Install Docker Desktop app.

2. Creation of Dockerfile and Saving in the folder containing your app.

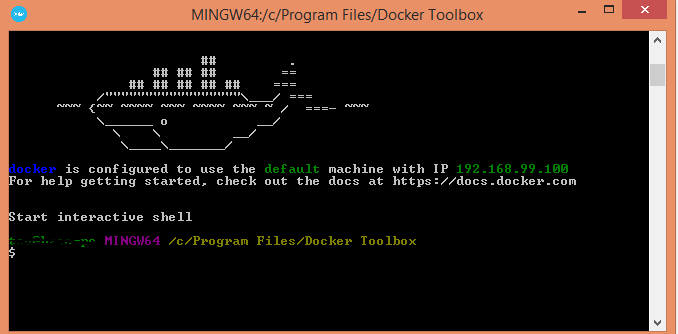


3. Creation of requirement.txt to inform Docker about the necessary libraries required and Saving in the folder containing your app.



4. After successful installation of Docker and Creation of above 2 files, go to Docker QuickStart Terminal.



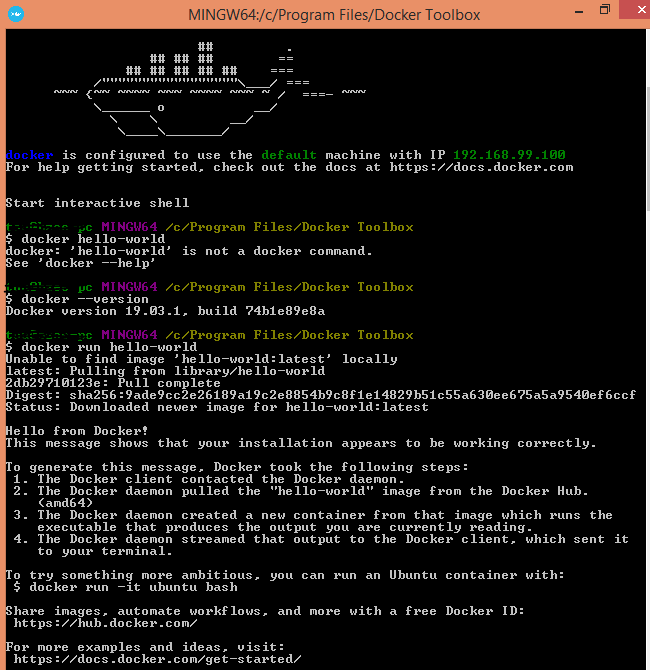


5. Check if everything is fine with Docker installation using the below commands:-

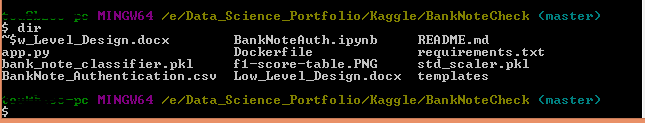
docker --version

docker run hello-world

docker

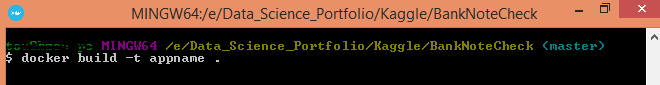


6. Navigate to the Working Directory of your app.



7. Building the Docker Image

Type the command: docker build -t appname .

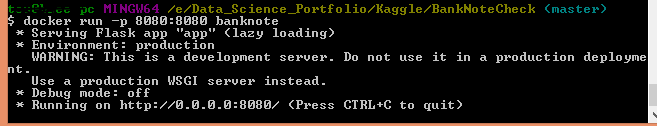


Use the app name that you like.

This will create your docker image with all the necessary libraries installed.

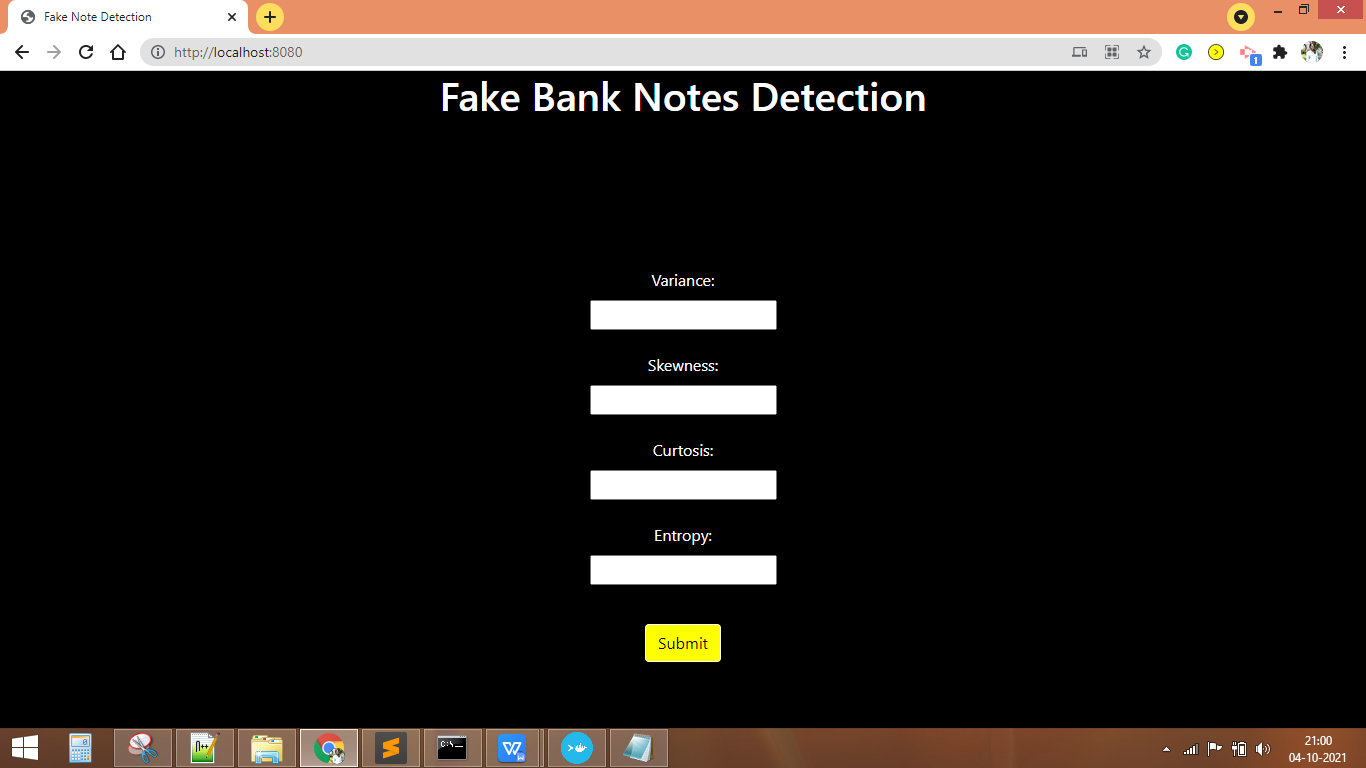
8. Running the docker

Type the command: docker run -p 8080:8080 appname



My app’s name is banknote.

This port number will depend upon the port number you wish to use while creating the API. I chose 8080.



Docker image successfully Created !!

Deployment into AWS EC2 instance:-

1. Create an AWS account, and log in to you AWS console.

2. Create an AWS instance of your choice.

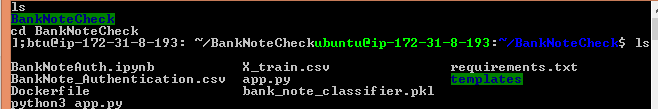
3. Setup relevant Security groups based on the port you wish to run your app in.

4. Connect to your AWS instance using SSH key-pair.

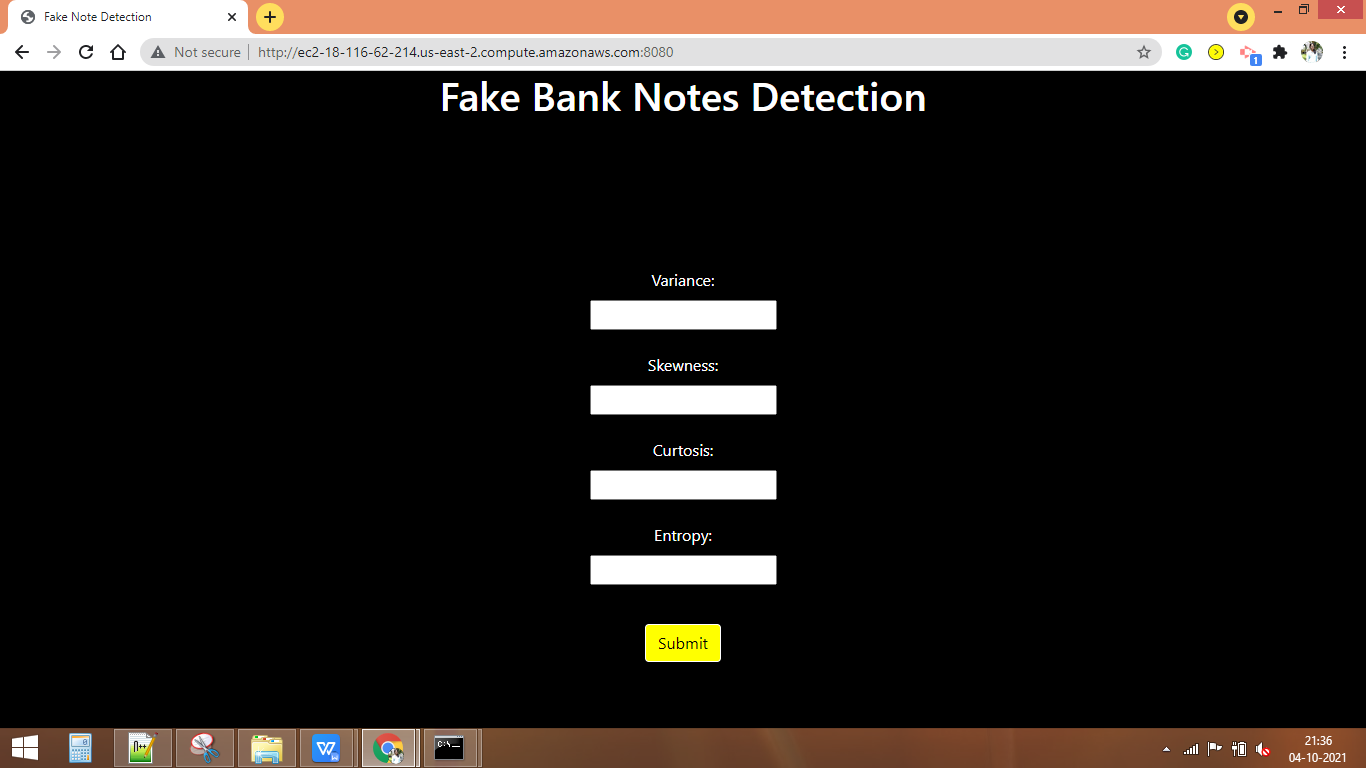
5. Transfer all your app files using scp.

6. Install all relevant libraries using pip command.

6. Once transfer is done. Type python3 app.py to run the app (just like in localhost).



7. This app is now up and running !



Production Deployment Done!