**MACHINE LEARNING OPERATIONS PROJECT**

**Deployment of Machine Learning Model on Docker, Kubernetes & Jenkins**

Submitted To: Mr. Chaitainya Soni

Submitted By: Harjot Singh (2110993784)

Shouary jain (2110993834)

Harsh Sharma (2110993787)

Tanuj Sharma (2110993846)

**DOCKER**

* Docker Installation:
* Visit the Official Docker website
* [www.docker.com/products/docker-desktop](http://www.docker.com/products/docker-desktop)
* Follow installation steps.
* Make a separate folder: a.py file containing a flask model and a docker file with no extension, and write the following in it.

FROM python:3.12.1

WORKDIR /app

COPY main\_file.py .

RUN pip install pandas

RUN pip install numpy

RUN pip install -U scikit-learn

COPY framingham.csv /app

EXPOSE 5000

CMD ["python","main\_file.py"]

* Image Building:
* Open docker desktop
* Open CMD in the folder, and run the following commands.
* Docker build –t image name.
* The built image image will be seen on docker desktop.
* Docker Hub: Log in to your docker hub online. Click on Create Repository and fill in the credentials.
* Image Pushing To Docker Hub:
* Run the following commands on the command prompt to push the image to the Docker hub.
* docker tag local-image:tagname new-repo:tagname
* docker push new-repo:tagname
* Now our image is pushed to the docker hub

**KUBERNETES**

* Kubernetes Installation:
* Run the following command in CMD
* curl.exe -LO <https://dl.k8s.io/release/v1.29.2/bin/windows/amd64/kubectl.exe>
* Check the version using the following command
* kubectl version: client
* Docker Settings:
* Open docker Hub
* Click on settings
* Choose container terminal as integrated
* Tick on (Expose daemon on tcp://localhost:2375 without TLS)
* Click Apply and restart
* Kubernetes Commands:
* kubectl config view
* Kubectl gets nodes
* Yaml File: Build a deployment. yaml in the same folder.
* Deployments:
* Use the following command to deploy the model:.
* kubectl apply -f deployments.yaml

**JENKINS**

* Log in to Jenkins using your local host.
* Click on the new Item & enter the item name to create a new pipeline.
* Click on pipeline.
* Now write the pipeline script.

pipeline {

agent any

stages {

stage('checkout') {

steps {

checkout scmGit(branches: [[name: '\*/main']], extensions: [], userRemoteConfigs: [[url: '<https://github.com/shouary/Final_mlops_project.git>']])

}

}

stage('Build'){

steps{

bat '''docker build -t hello-world-flask .'''

bat '''docker run -p 5000:5000 hello-world-flask'''

}

}

}

}

* Click on the “GitHub hook trigger for GITScm polling” and enable it.
* Enable the Groovy Sandbox.
* Now build the project.