Business Requirements

• The team of developers working on new features will merge their code to a GitHub repo.

• As soon as the code reaches GitHub, using a CI (Continuous Integration) pipeline, setup in Jenkins, automated builds will be triggered.

• The automated builds will frequently deploy new features to the production website.

•Every build will prepare a Dockerfile and push docker images to docker-hub.

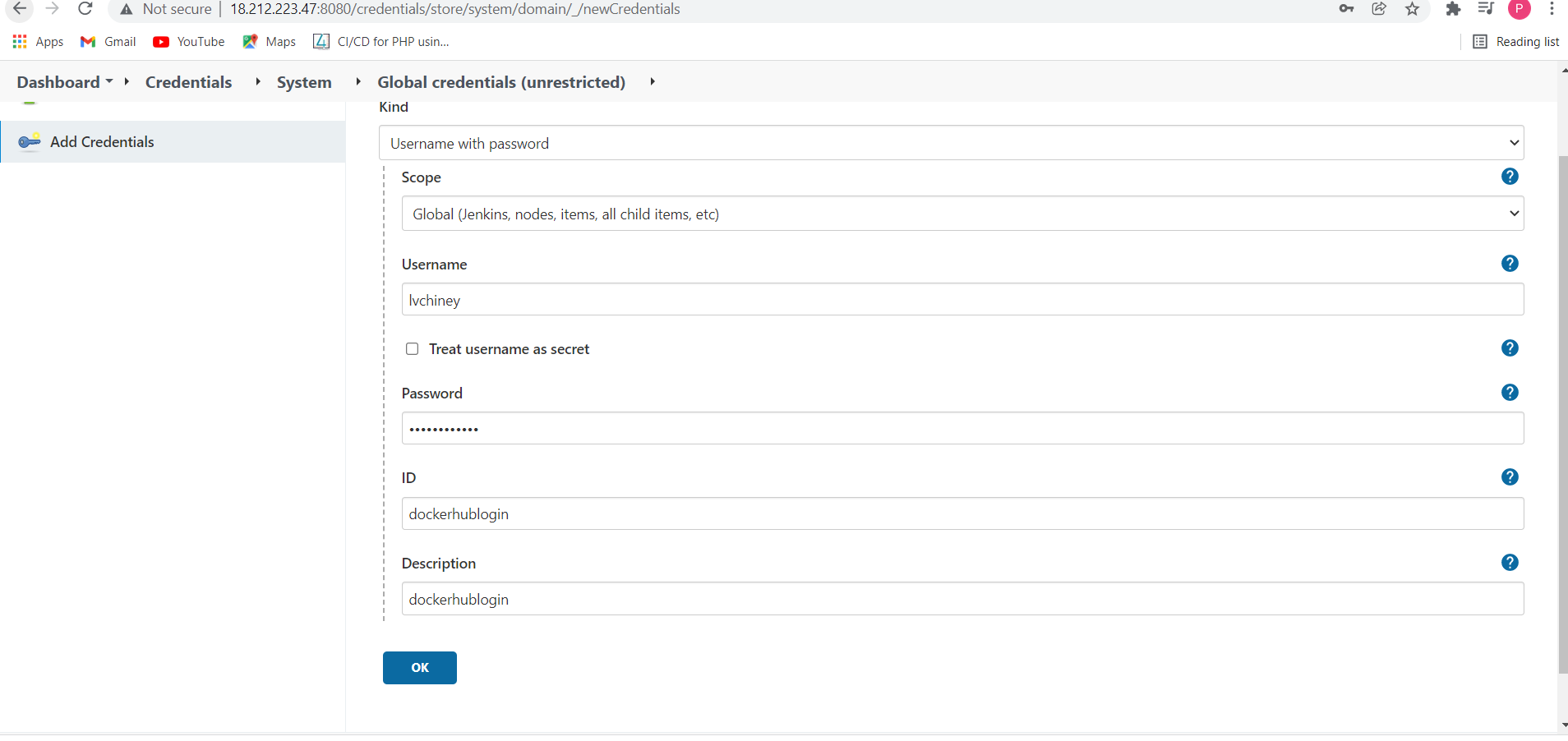
•Every docker image will be deployed (Continuous Deployment) to a kubernetes-cluster

GitHub: <https://github.com/lvchiney/cicd-pipeline-train-schedule-autodeploy.git>

Solution:

Install Kubernetes Continuous Plugin 1.0 version

Install Docker Pipeline Plugin



Set Up Kubernestes cluster (Master and Worker1)

Install docker in server

# This script is meant for quick & easy install via:

$ curl -fsSL https://get.docker.com -o get-docker.sh

$ sh get-docker.sh

sudo systemctl start docker

systemctl enable docker

Provide sudo access to Jenkins & current server :

sudo usermod -aG docker centos

sudo usermod -a -G docker jenkins

Pipeline:

pipeline {

agent any

stages {

stage('Checkout Source') {

steps {

git 'https://github.com/lvchiney/cicd-pipeline-train-schedule-autodeploy.git'

}

}

stage('Build image') {

steps{

script {

sh "docker build . -t lvchiney/project2"

}

}

}

stage('Pushing Image') {

steps{

withCredentials([string(credentialsId: 'dockerhubp', variable: 'dockerHubPwd')]) {

sh "docker login -u lvchiney -p ${dockerHubPwd}"

}

sh "docker push lvchiney/project2"

}

}

stage('Deploying App to Kubernetes') {

steps {

script {

kubernetesDeploy(configs: "deploymentservice.yml", kubeconfigId: "kubernetes")

}

}

}

}

}

deploymentservice.yml

|  |
| --- |
| apiVersion: apps/v1 |
|  | kind: Deployment |
|  | metadata: |
|  | name: nodeapp-deployment |
|  | labels: |
|  | app: nodeapp |
|  | spec: |
|  | replicas: 1 |
|  | selector: |
|  | matchLabels: |
|  | app: nodeapp |
|  | template: |
|  | metadata: |
|  | labels: |
|  | app: nodeapp |
|  | spec: |
|  | containers: |
|  | - name: nodeserver |
|  | image: lvchiney/project2:latest |
|  | ports: |
|  | - containerPort: 3000 |
|  |  |
|  |  |
|  | --- |
|  |  |
|  | apiVersion: v1 |
|  | kind: Service |
|  | metadata: |
|  | name: nodeapp-service |
|  | spec: |
|  | selector: |
|  | app: nodeapp |
|  | type: LoadBalancer |
|  | ports: |
|  | - protocol: TCP |
|  | port: 5000 |
|  | targetPort: 3000 |
|  | nodePort: 31110 |