A screenshot of a computer

Description automatically generated

plugin

[Email Extension Template](https://plugins.jenkins.io/emailext-template)

A screenshot of a chat

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

Set project ให้ user

A screenshot of a computer

Description automatically generated with medium confidence

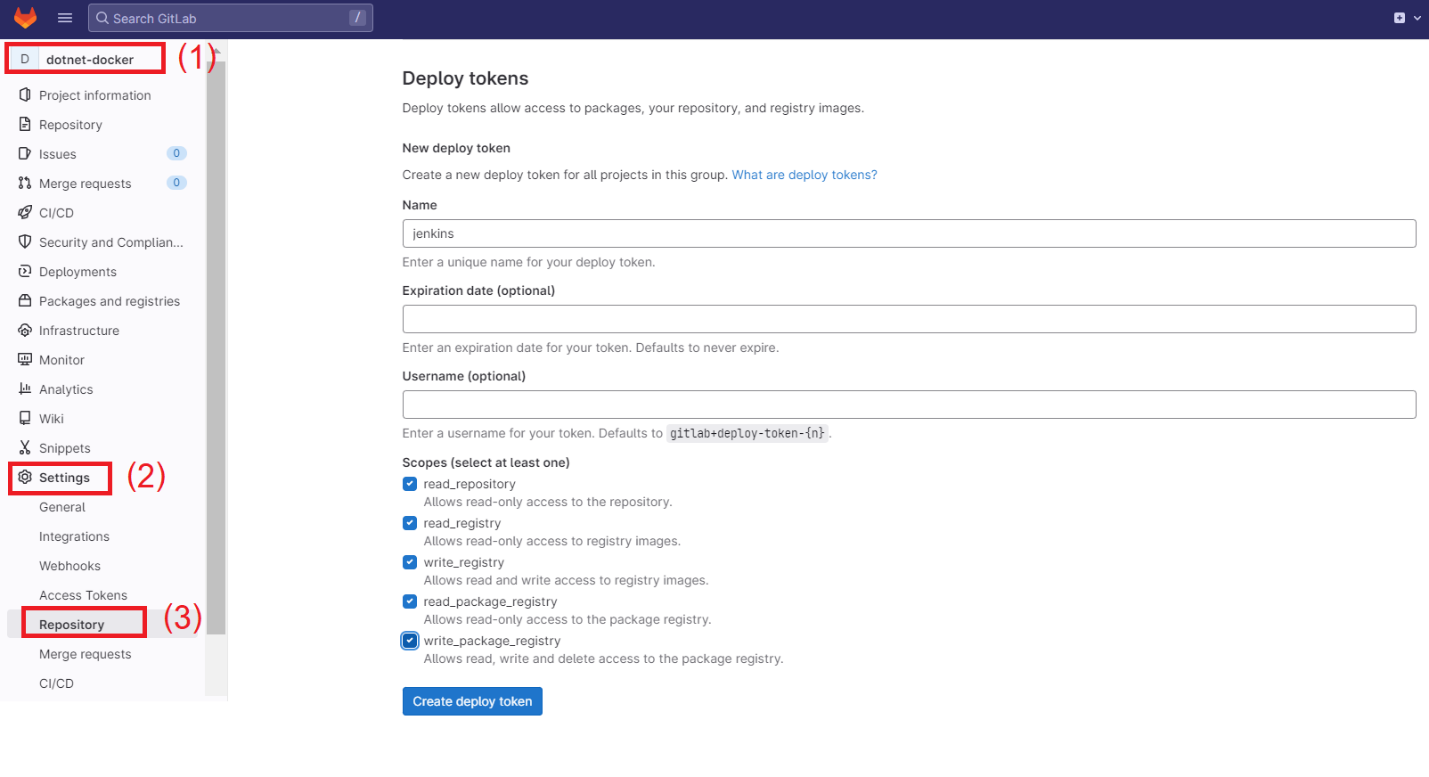
how2coding = admin

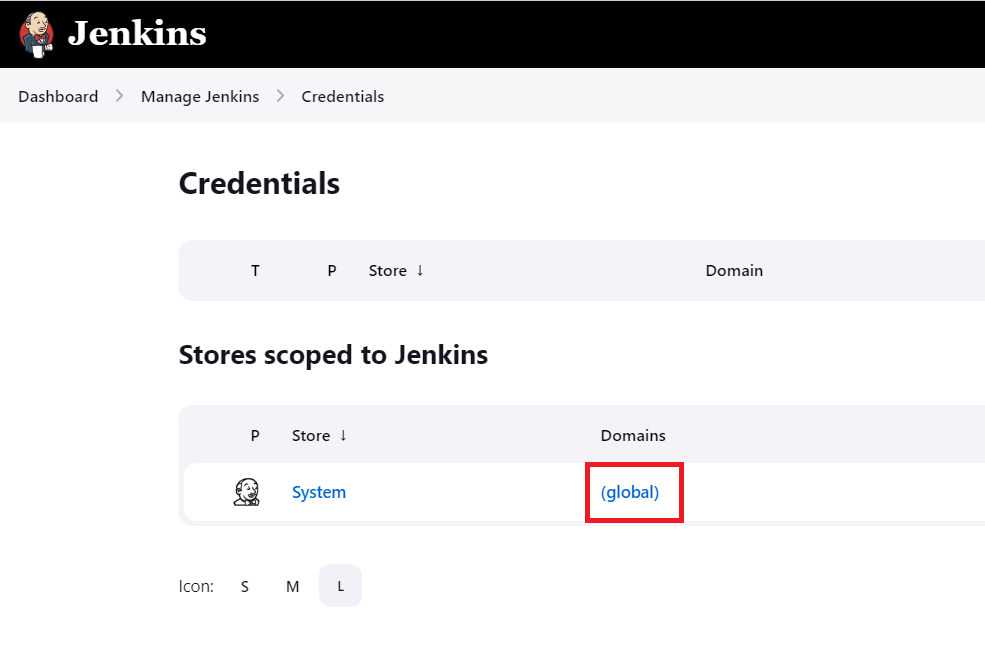
user1 = user

A screenshot of a computer

Description automatically generated

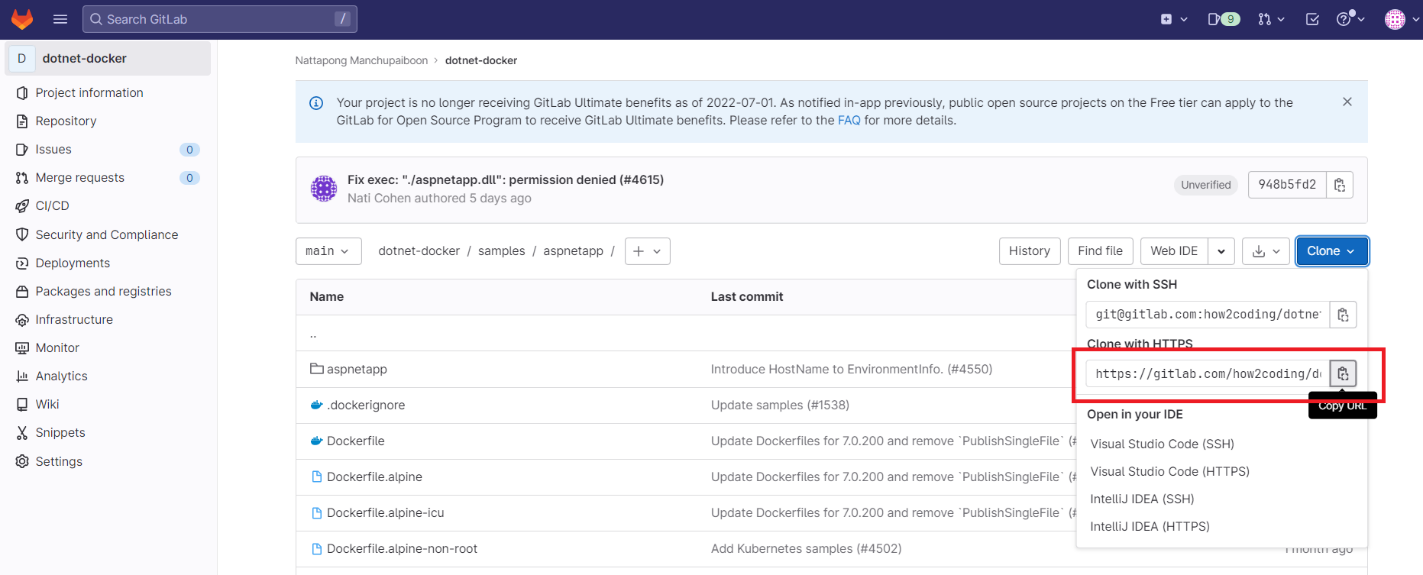
Gitlab





A screenshot of a computer

Description automatically generated



A picture containing text, line, font, screenshot

Description automatically generated

A screenshot of a computer

Description automatically generated

|  |
| --- |
| Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Post "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/build?buildargs=%7B%7D&cachefrom=%5B%5D&cgroupparent=&cpuperiod=0&cpuquota=0&cpusetcpus=&cpusetmems=&cpushares=0&dockerfile=Dockerfile&labels=%7B%7D&memory=0&memswap=0&networkmode=default&rm=1&shmsize=0&t=dotnetweb&target=&ulimits=null&version=1": dial unix /var/run/docker.sock: connect: permission denied |

usermod -aG docker jenkins

usermod -aG root jenkins

chmod 777 /var/run/docker.sock

|  |
| --- |
| pipeline {      agent any        environment {         ENV = 'production'         REPOSITORY\_NAME='dotnetweb'      }      stages {          stage('Checkout'){              steps{                    cleanWs()                    checkout([$class: 'GitSCM', branches: [[name: '\*/main']], extensions: [],                  userRemoteConfigs: [[credentialsId: 'dotnet-docker',                  url: 'https://gitlab.com/how2coding/dotnet-docker.git']]])                    sh 'ls -lst '              }          }            stage("Docker build") {              steps{                  sh 'docker build -t ${REPOSITORY\_NAME} ./samples/aspnetapp'              }          }             stage("Docker run") {              steps{                  sh 'docker run -d -it --rm -p 80:80 --name aspnetcore\_sample ${REPOSITORY\_NAME}'                }          }        }      post {           always {               echo 'This will always run'           }             success {              echo 'This will run only if successful'              emailext attachLog: true, subject: 'jenkins build $PROJECT\_NAME success', recipientProviders: [                  [$class: 'DevelopersRecipientProvider'],                  [$class: 'RequesterRecipientProvider']],                  body: 'Hello Team,<br/><br/>Project: $PROJECT\_NAME  <br>Build Number: #$BUILD\_NUMBER is <font color="green">success</font> <br/> Best Regards,',                   to: 'it@abc.com'             }           failure {               emailext attachLog: true, subject: 'jenkins build $PROJECT\_NAME failure', recipientProviders: [                  [$class: 'DevelopersRecipientProvider'],                  [$class: 'RequesterRecipientProvider']],                  body: 'Hello Team,<br/><br/>Project: $PROJECT\_NAME  <br>Build Number: #$BUILD\_NUMBER  is <font color="red">failure!</font> <br/>please see attached file for detail <br/> <br/> Best Regards, <br/> ',                   to: 'it@abc.com'                    echo 'This will run only if the run was marked as unstable'           }             unstable {               echo 'This will run only if the run was marked as unstable'           }           changed {               echo 'This will run only if the state of the Pipeline has changed'               echo 'For example, if the Pipeline was previously failing but is now successful'           }       }  } |