Contents

[Deploy an Angular application on Kubernetes 1](#_Toc130948805)

[Setup Agular Environment 1](#_Toc130948806)

[1. install angular app with following commands 1](#_Toc130948807)

[2. Need to install nodejs 14 with following commands 1](#_Toc130948808)

[3. Create a new Angular app by running the following command 1](#_Toc130948809)

[Application on Docker 2](#_Toc130948810)

[1. Build and push the Docker image: Use the command 2](#_Toc130948811)

[2. Publish the Docker image to Docker Hub 2](#_Toc130948812)

[Deployment on Kubernetes 3](#_Toc130948813)

[1. Run following command to create Pod 3](#_Toc130948814)

[Deployment 3](#_Toc130948815)

[Deploy an Vue.js application on Kubernetes 4](#_Toc130948816)

[Setup node.js Environment 4](#_Toc130948817)

[Run following commands: 4](#_Toc130948818)

[Install npm 4](#_Toc130948819)

[Installing Vue.js by Using CLI 5](#_Toc130948820)

[Create a Vue.js Application using CLI 5](#_Toc130948821)

[application on Docker 5](#_Toc130948822)

[Build and push the Docker image: Use the command 6](#_Toc130948823)

[Publish Docker image to Docker Hub 6](#_Toc130948824)

[Create Pod and Service 6](#_Toc130948825)

[Verify that the Pod and Service are running by running the following commands: 7](#_Toc130948826)

# 

# Deploy an Angular application on Kubernetes

## Setup Agular Environment

### install angular app with following commands

sudo apt-get update

sudo apt-get install nodejs npm

npm install -g @angular/cli

when this command is not run correctly use below.

npm install -g @angular/cli

### Need to install nodejs 14 with following commands

1: curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh

2: curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash

3: source ~/.bashrc

4: nvm list-remote

5: nvm install v16.14.0

6: nvm list

7: nvm install lts/fermium

8: node -v

successfully install nodejs 14

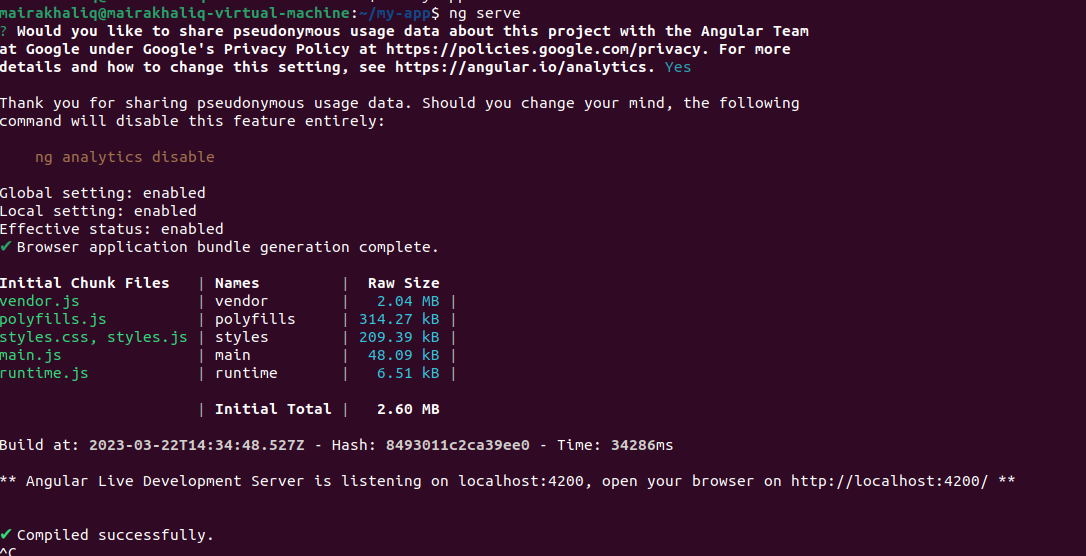
### Create a new Angular app by running the following command

ng new my-app

cd my-app

ng serve

app should now be accessible at <http://localhost:4200>.



## Application on Docker

|  |
| --- |
| # base image  FROM node:14.16.0-alpine3.13  # set working directory  WORKDIR /app  # install and cache app dependencies  COPY package.json /app/package.json  RUN npm install  # add app  COPY . /app  # start app  CMD ["npm", "start"] |

### Build and push the Docker image: Use the command

docker build -t my-angularapp .

### Publish the Docker image to Docker Hub

* docker login
* **tag image** (sudo docker tag my-angularapp mairakhaliq/angularapp :latest)
* docker push mairakhaliq/angularapp
* image successfully push

## Deployment on Kubernetes

Create this file.

## 

|  |
| --- |
| apiVersion: v1  kind: Pod  metadata:  name: webapp  spec:  containers:  - name: webapp  image: mairakhaliq/angularapp:latest  imagePullPolicy: Always  resources: {}  ports:  - containerPort: 4200 |

### Run following command to create Pod

kubectl create -f pod.yml

### Deployment

|  |
| --- |
| apiVersion: apps/v1  kind: Deployment  metadata:  name: angular-webapp  spec:  replicas: 1  selector:  matchLabels:  app: angular-webapp  template:  metadata:  labels:  app: angular-webapp  spec:  containers:  - name: angular-webapp  image: mairakhaliq/angular:latest  ports:  - containerPort: 4200 |

kubectl create -f deployment.yml

|  |
| --- |
| apiVersion: v1  kind: Service  metadata:  name: angular-webapp  spec:  type: NodePort  selector:  app: angular-webapp  ports:  - name: http  port: 4200  targetPort: 4200 |

* kubectl create -f service.yml
* kubectl get svc
* kubectl get pods
* kubectl get deployment

# Deploy an Vue.js application on Kubernetes

# Setup node.js Environment

## Run following commands:

* sudo apt install nodejs
* nodejs –version

### Install npm

* sudo apt install npm
* npm –version
* npm install vue@next

### Installing Vue.js by Using CLI

* sudo npm install -g @vue/cli
* vue –version
* sudo npm update -g @vue/cli

### Create a Vue.js Application using CLI

* vue create my-app
* cd my-app
* npm run serve

## application on Docker

|  |
| --- |
| # base image  FROM node:lts-alpine  # set working directory  WORKDIR /app  # copy package.json and package-lock.json  COPY package\*.json ./  # install dependencies  RUN npm install  # copy project files and folders to the container  COPY . .  # build for production with minification  RUN npm run build  # expose the port the app runs on  EXPOSE 8080  # start the app  CMD [ "npm", "run", "serve" ] |

### Build and push the Docker image: Use the command

* docker build -t my-vue-app.

### Publish Docker image to Docker Hub

* docker login
* **tag image** (docker tag my-vue-app:latest mairakhaliq/my-vue-app:latest)
* docker push mairakhaliq/my-vue-app
* image successfully push

## Create Pod and Service

|  |
| --- |
| apiVersion: v1  kind: Pod  metadata:  name: my-vue--app  spec:  containers:  - name: my-vue-app  image: mairakhaliq/my-vue-app:latest  imagePullPolicy: Always  ports:  - containerPort: 8000  ---  apiVersion: v1  kind: Service  metadata:  name: my-vue-app  spec:  type: NodePort  selector:  app: my-vue-app  ports:  - name: http  port: 8000  targetPort: 8080 |

kubectl apply -f pod.yml

### Verify that the Pod and Service are running by running the following commands:

* kubectl get pods
* kubectl get services