**Project Title: Deploy Angular Application in Docker Container**

**Submitted by Aruna Annavajjula**

**Instructor: Anuj**

DESCRIPTION

Deploy the Angular application in Docker. The Angular application should be built with the Angular CLI along with Docker Compose for development and production.

**Problem Statement Scenario:**   
HTQual Technology Solutions hired you as a MEAN Stack Developer. The organization decided to implement DevOps to develop and deliver the products. Since HTQual is an Agile organization, they follow Scrum methodology to develop the projects incrementally. The Company decided to develop their website on Mean stack. Since you are the MEAN stack developer, you have to demonstrate that deploying the Angular application on Docker is always the best approach to develop a project and test it incrementally. You agreed upon the following:

    • Setting up an image for code development

    • Build the application in Docker and host it in Docker Hub

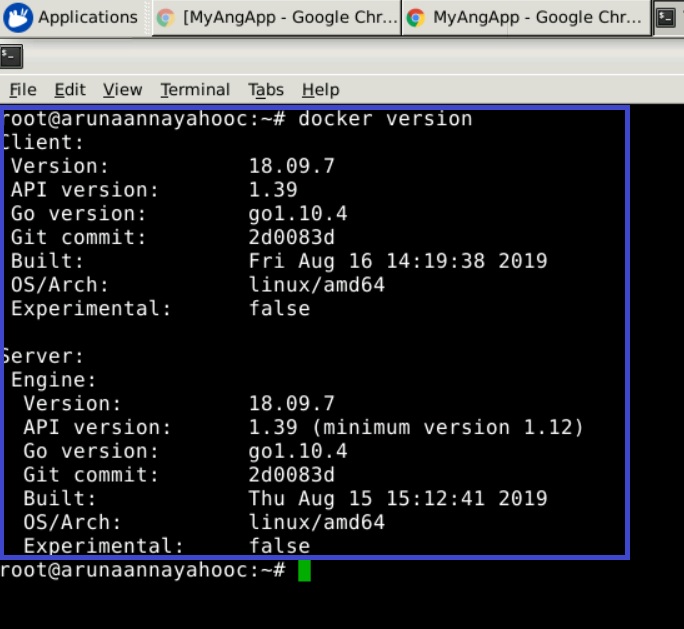
    • List the advantages, disadvantages, and document the tasks involved

Your goal is to demonstrate the Angular application and run it in Docker container.

**You must use the following tools:**   
    • Docker – To package the application in a Docker container   
    • Node.js – To support the Angular application with the required node modules  
    • Angular CLI – To execute and bundle the dependencies together  
    • Linux (Ubuntu) – As a base operating system to start and execute the project

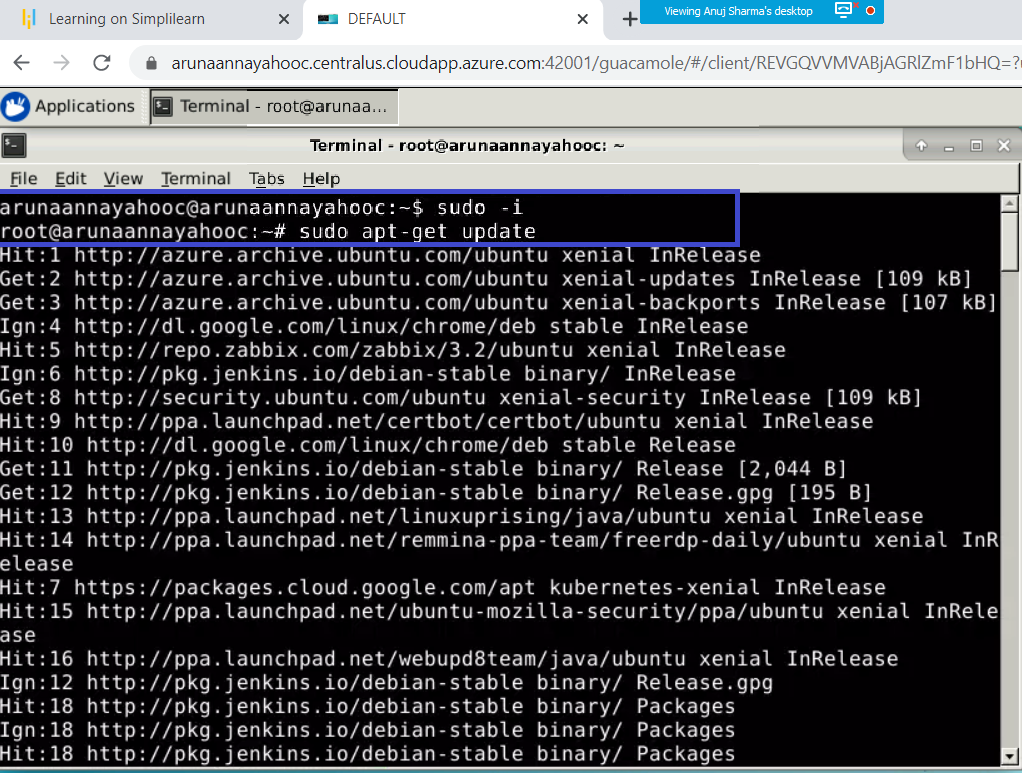
**Note: This is a solution document on how the demonstration is performed on Docker 18. + Version.**

**Docker Version on VM**

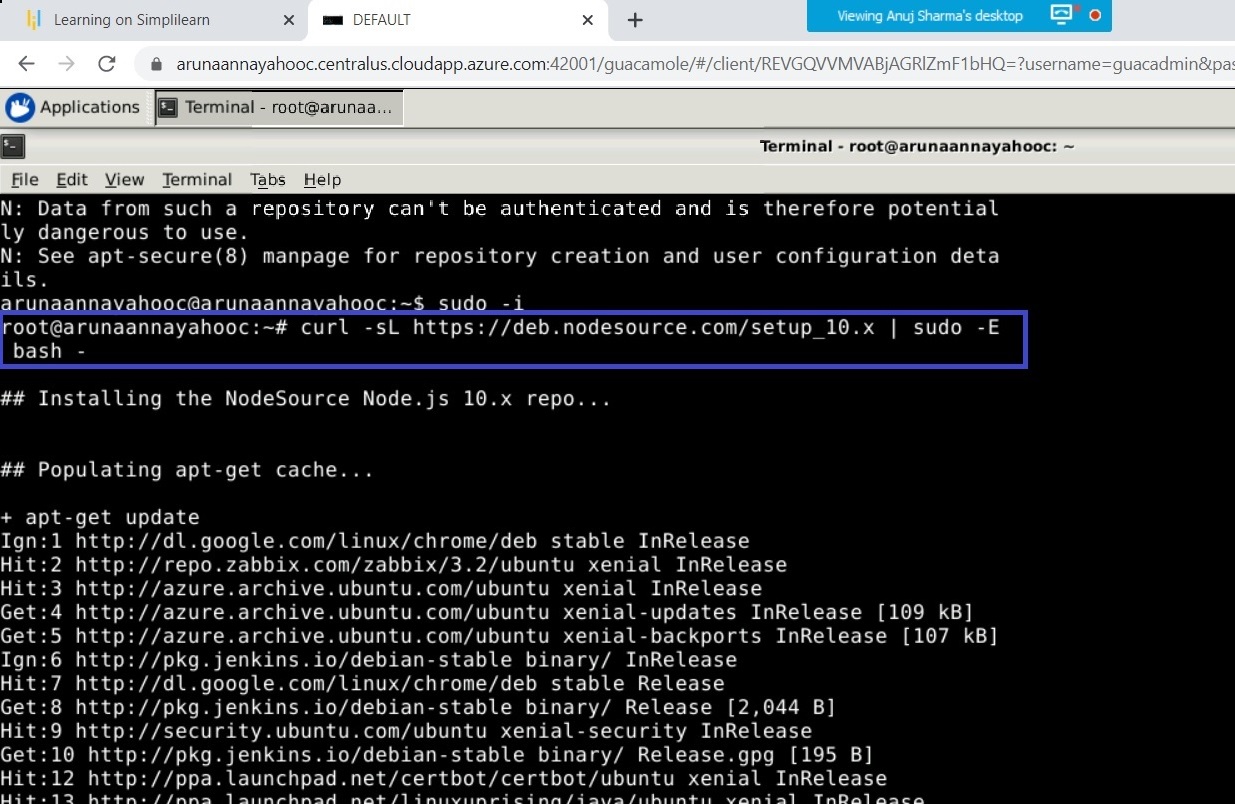
****

**Install Angular application on Ubuntu by running the commands mentioned below:**

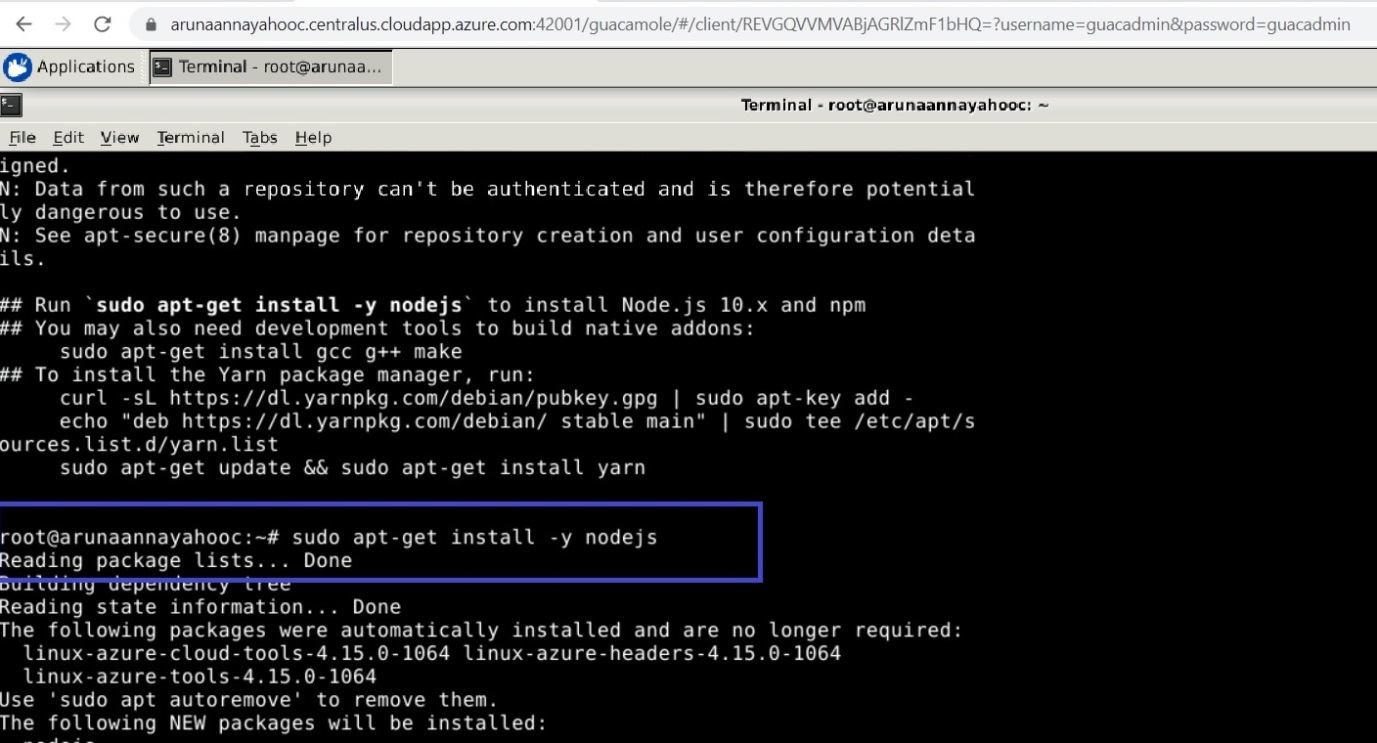
**sudo apt-get update**



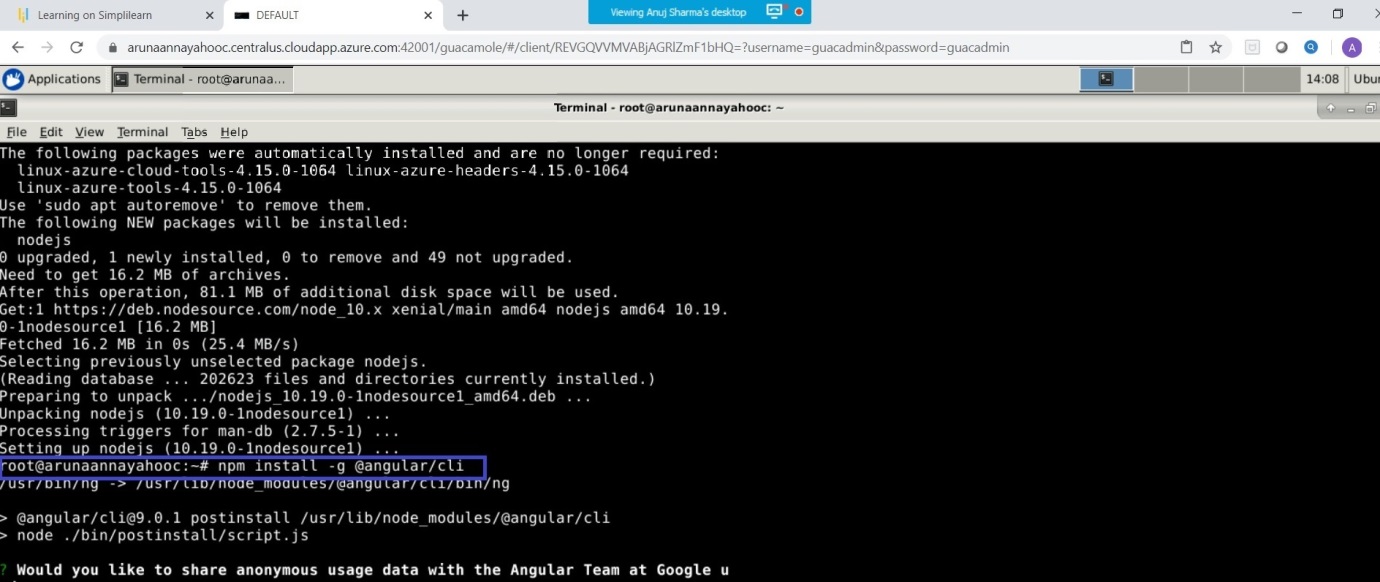
**curl -sL https://deb.nodesource.com/setup\_10.x | sudo -E bash –**



**apt-get install -y nodejs**



**npm install -g @angular/cli**



**I had issues installing angular/cli**

**I issue I faced is after selecting CSS stylesheet.**

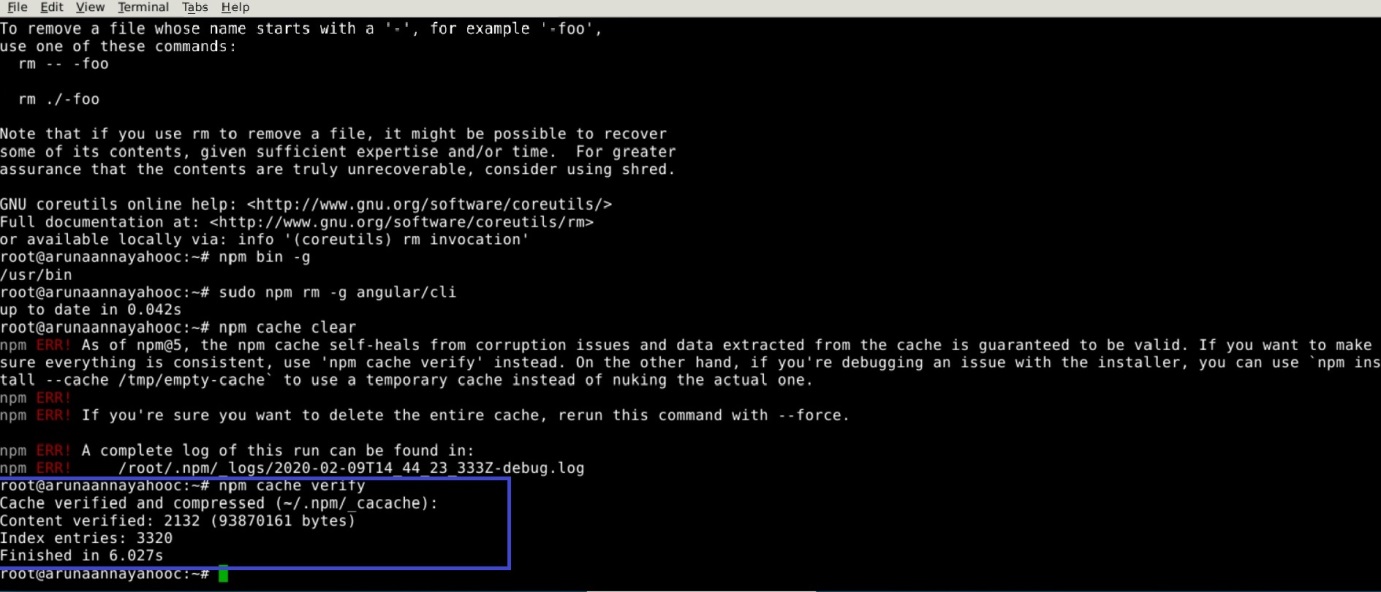
**Gave a path to root/user/….. logs.**

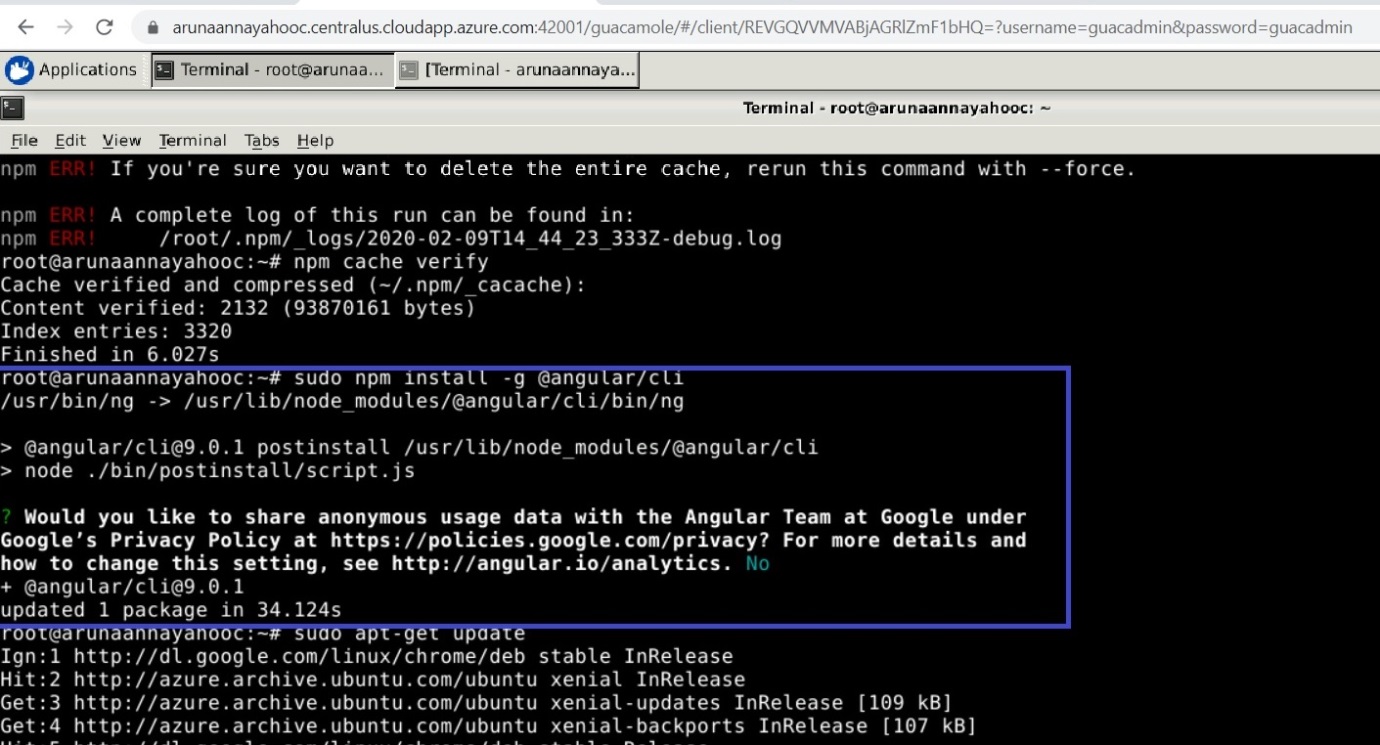
**Instead of digging into the logs, I removed the installed angular /cli ,**

**I removed angular/cli install and Cleared cache**. To make sure nothing is cached during uninstallation.

sudo npm rm –g angular/cli

npm cache verify



**Reinstalled angular/cli**

**After completion of installation of angular.cli**

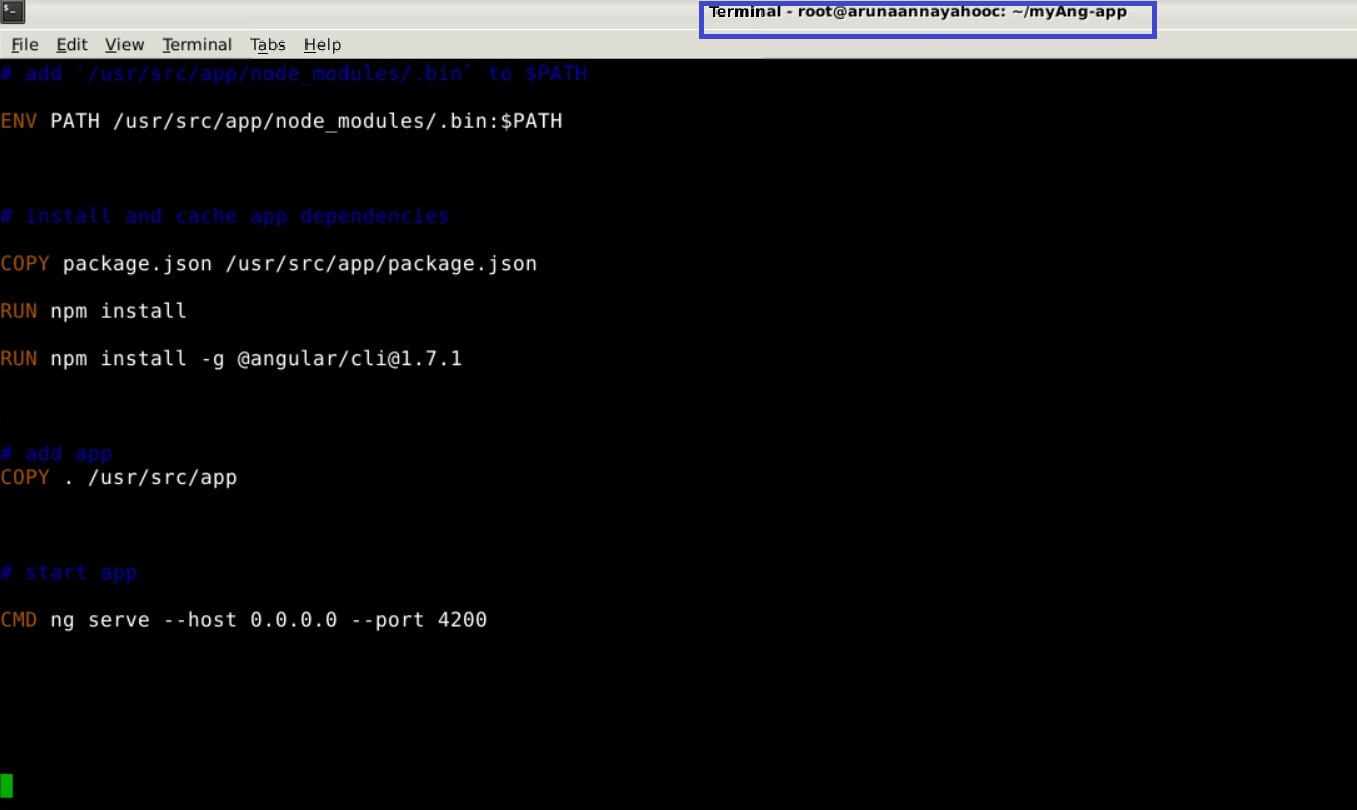
create a new folder

**ng new myAng-app**

cd myAng-app

create a **vi Docker file**

Navigate inside the project, and **create a Dockerfile.**



Now add the following content to it.

# base image

FROM node:12

# set working directory

RUN mkdir /usr/src/app/node12

WORKDIR /usr/src/app/node12

# add `/usr/src/app/node\_modules/.bin` to $PATH

ENV PATH /usr/src/app/node\_modules/.bin:$PATH

# install and cache app dependencies

COPY package.json /usr/src/app/package.json

RUN npm install

RUN npm install -g @angular/cli@1.7.1

# add app

COPY . /usr/src/app

# start app

CMD ng serve --host 0.0.0.0 –port 4200

Exit from vi editor Dockerfile

**Esc :wq!**

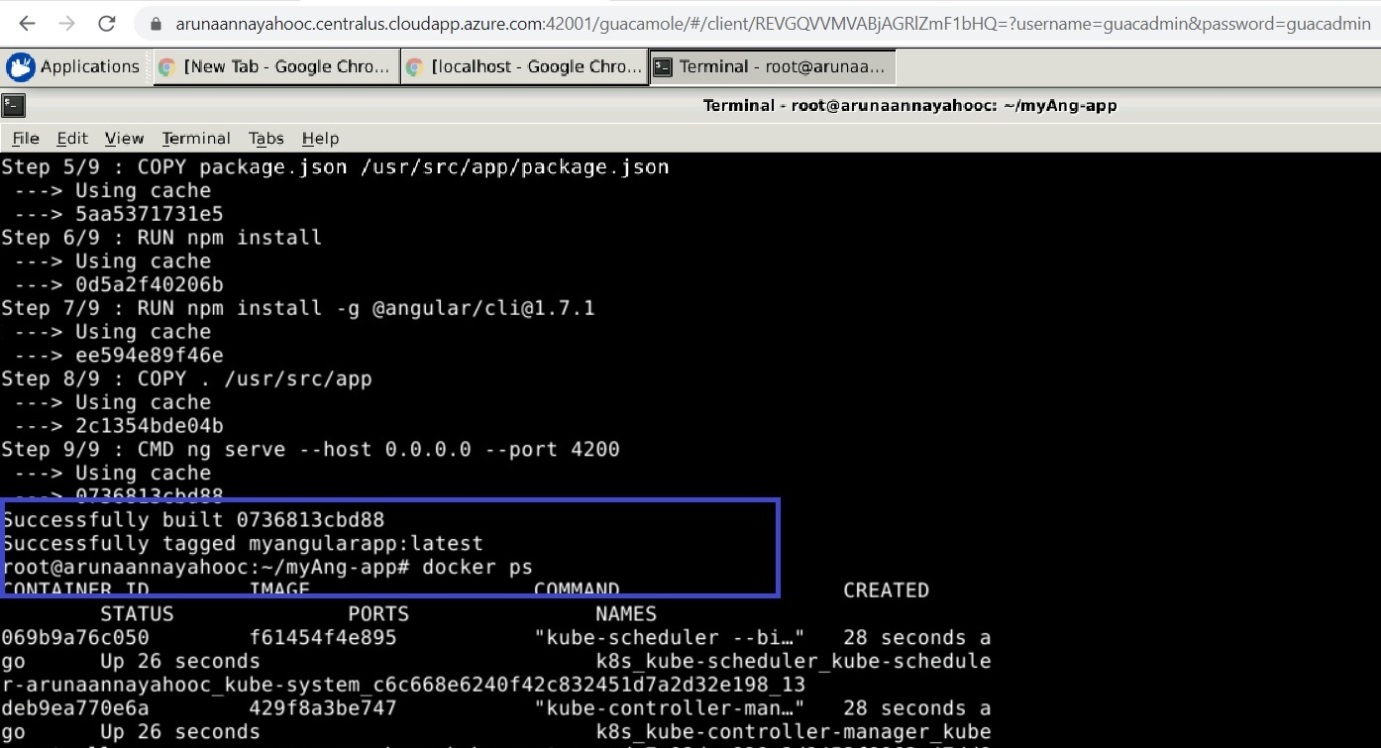
No protocol shows up

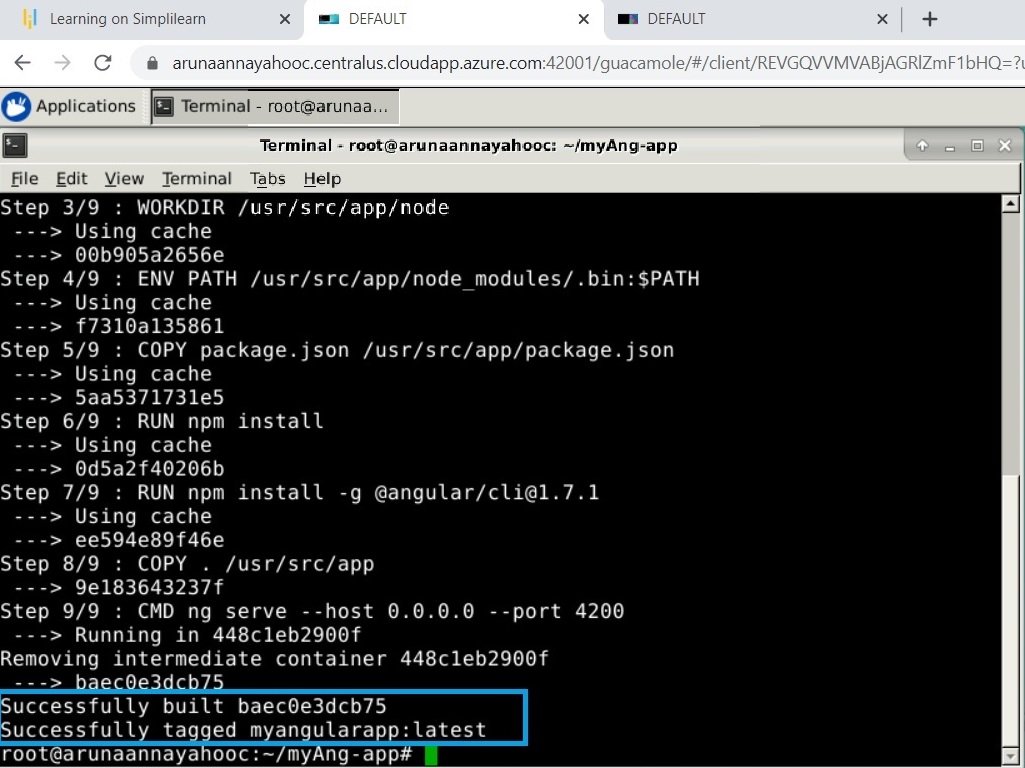
**From terminal**

**cat Dockerfile**

Build and tag the docker image

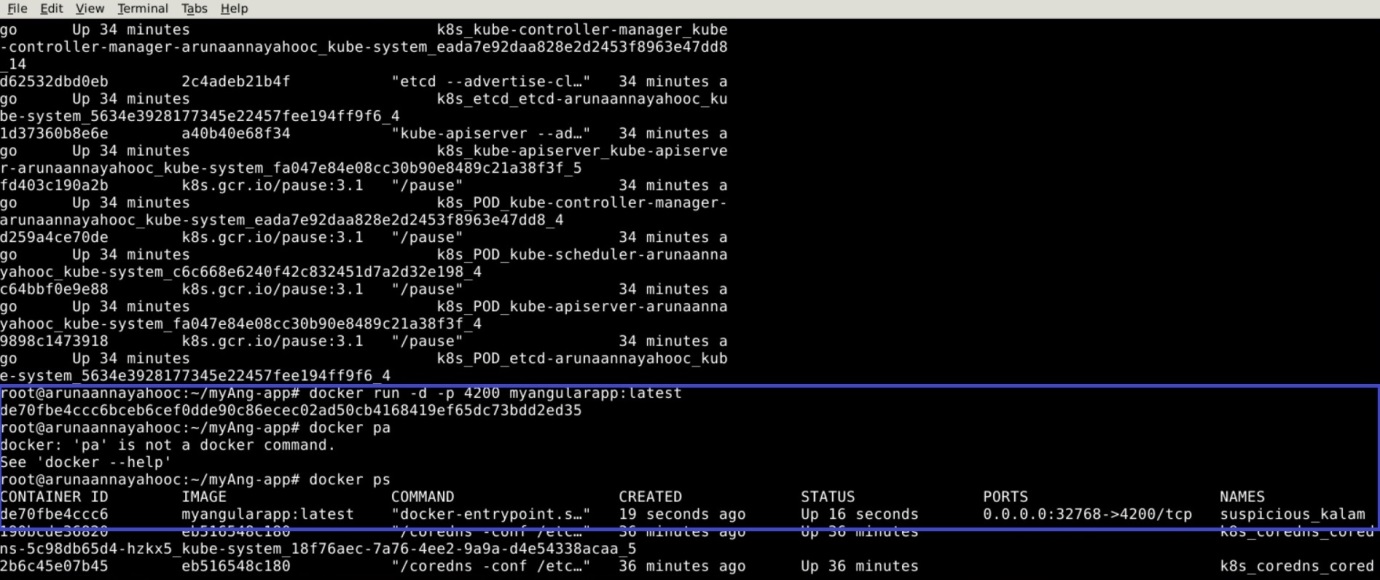
**docker build -t myangularapp .**

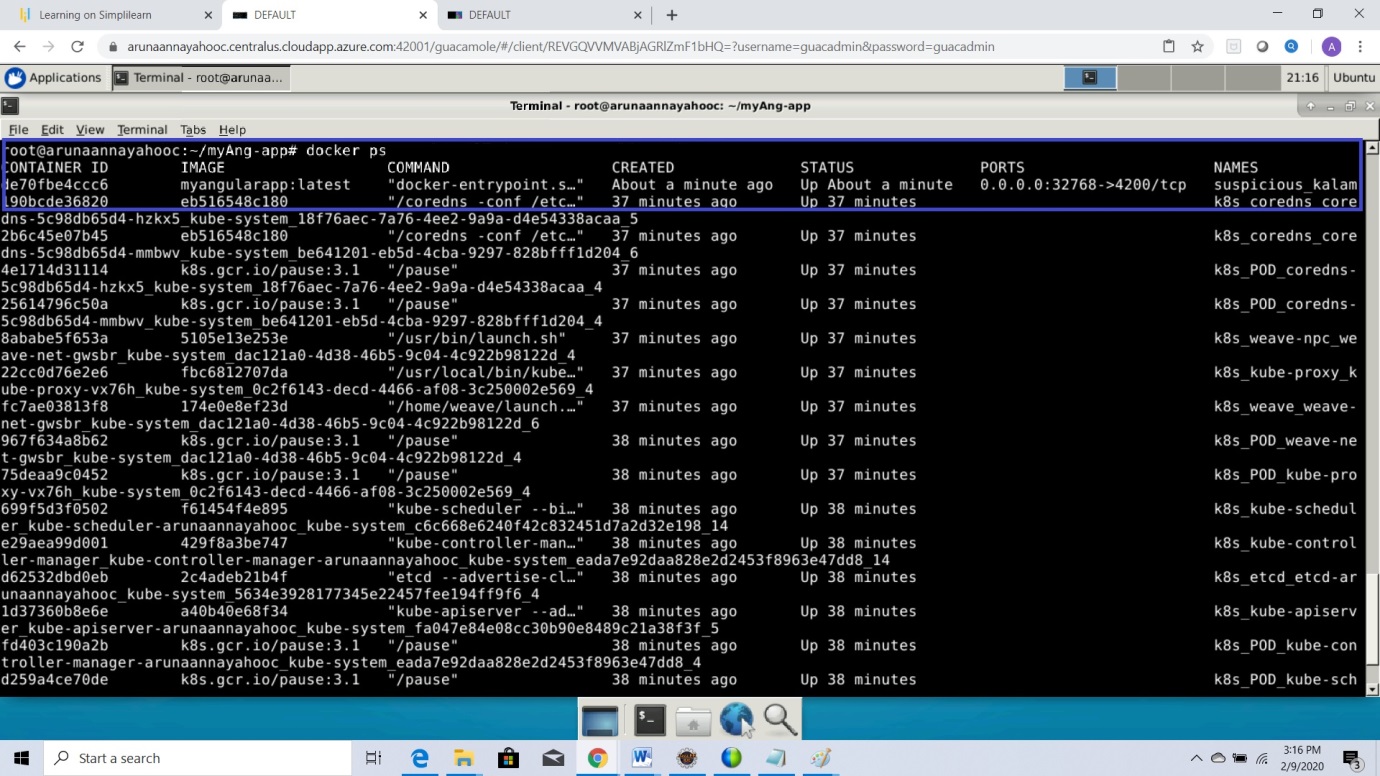


I have multiple builds successfully completed. 

**Exit the execution of application, and now run the Angular application in docker detach mode.**

**Docker run –d –p 4200 myangularapp.latest**

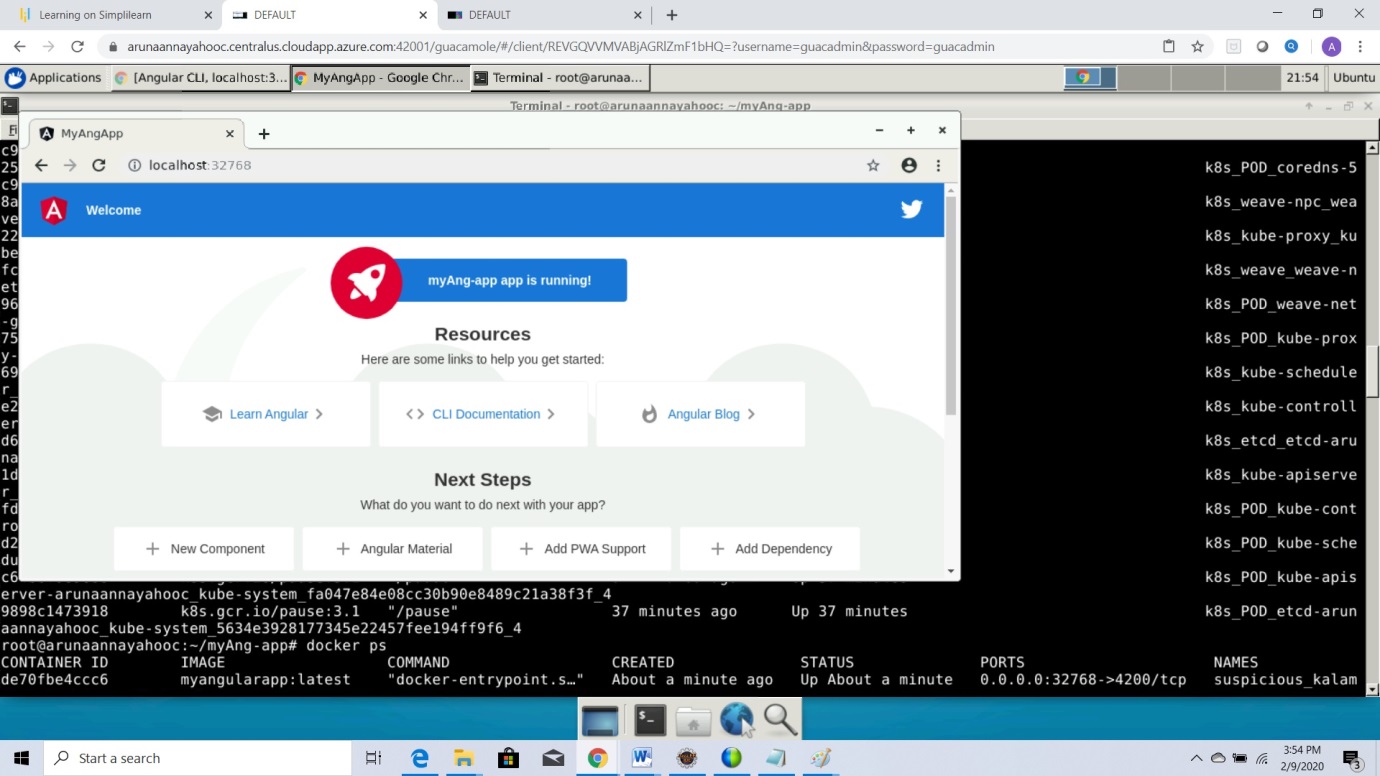
****



Now, open the browser and you should be able to run the application in the browser by navigating to the URL mentioned below.

<http://localhost:4200>

URL: http://localhost:32768/



You will be able to load the Angular application page in your browser.

Now, check if the container is running.

**docker login**

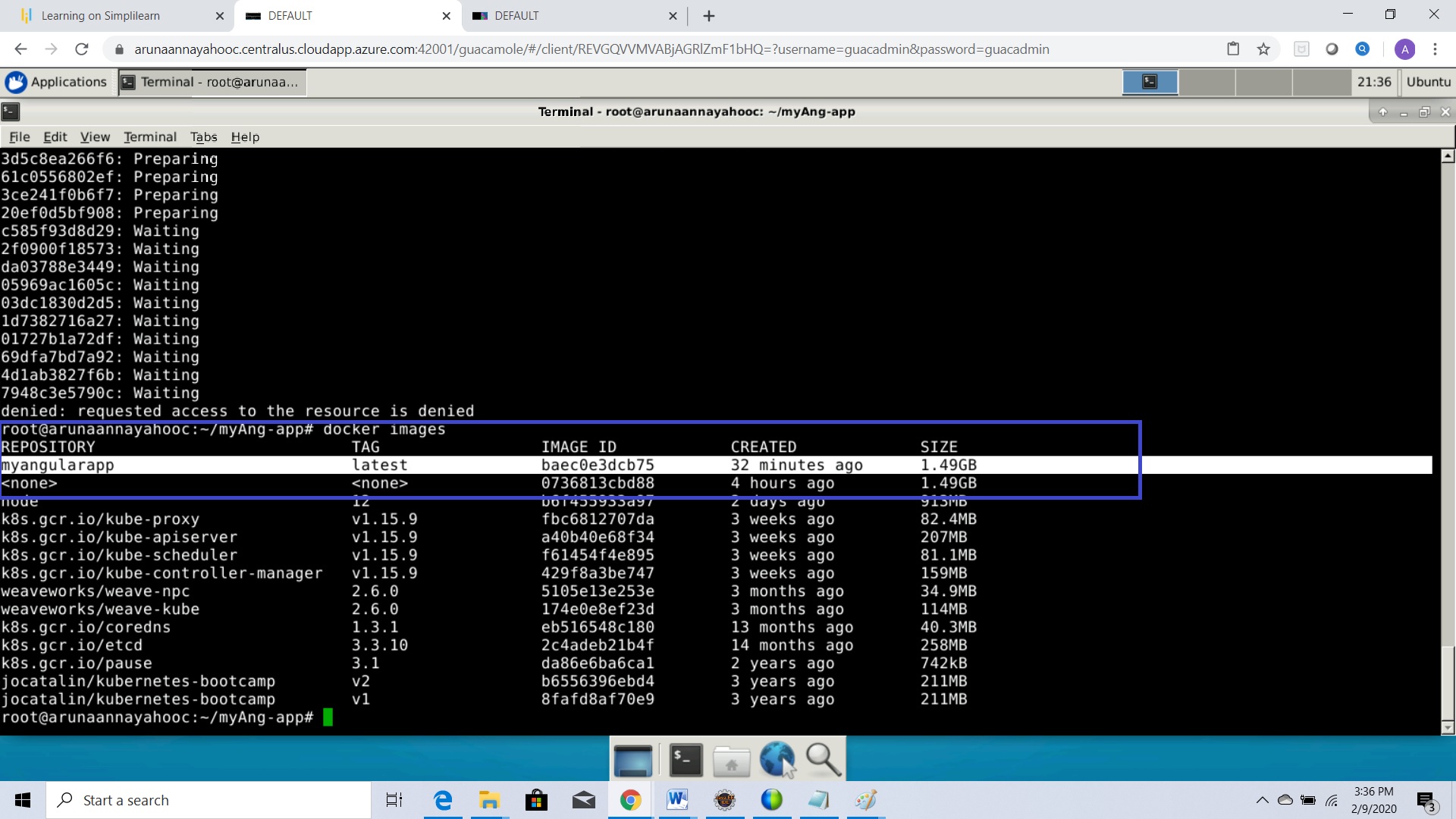
**Username**

pw

**root@arunaannayahooc:~/myAng-app# docker push myangularapp:latest**

requested access to the resource denied.

**docker images**



Myangularapp folder shows as 1.49GB size when verified from Docker Images.

Docker is a containerization tool used to streamline application, development, and deployment workflows across various environments. The container allows a developer to wrap up an application with all essential parts, libraries, dependencies together as one package. Docker is used as a continuous integration tool for developers and system administrators.

Benefits of docker: 1. Reproducibility: Docker container is guaranteed to be identical on any system that can run Docker. The exact specifications are stored in a Dockerfile. This file can be distributed among the team members or org so that the functions built on the images are identical and easier to track the problems and keep track of the application.

2. Avoid conflicting dependency.

3. Security is not compromised when one container gets affected.

4. For simple use cases and images, the files can be saved and pushed to docker hub and anyone from your org can pull and run the same container image with exactly the same npm package.

Unlike every tool, docker has cons and pros. Cons are when complicated apps( like large images) are developed and no sysadmin, Security will be a concern for complicated apps and performance might be critical as the docker containers share the host kernel and do not emulate a full operating system.

**Challenges faced during this project: I am unable to install angular /cli.**

I did some research and found that sometimes, when we uninstall and reinstall – installation completes as expected.

I had hard time, creating a docker image initially. I included build step also in the vi Dockerfile and when I entered build command, fetching started and failed. Second time it failed because I did not enter(“.”) after the path in build command

After figuring in out both, successful build with ID showed up. docker run command with tag container showed with port number.

<http://localhost:4200>

URL: http://localhost:32768/

I was trying with 4200 but it did not work. I finally gave a shot with port: 32768 it worked and all the SS are included in this document. Finally at the end verified docker images and verified the image with 1.4 GB in size.

From the mistakes, I have learnt little bit in depth about the docker, its advantages and how to implement. If I would have completed without errors, my knowledge on this might be minimal.