CLOUD COMPUTING LAB

CAT – 2

TEAM MEMBERS:

1934045 – SNEKA P

1934052 – TANUJA A

1934055 – VARSHA B

AIM:

To dockerize a web application and deploy it in the kubernetes cluster.

APPLICATION:

CIT REGISTRATION WEBSITE

PROBLEM STATEMENT:

This is to deploy ancit registration web portal for students know about the college infrastructure and brochure and to register for their interested courses.

DOCKER FILE:

Cclab > Dockerfile

```
1  FROM nginx:alpine
2  COPY . /usr/share/nginx/html
```

BUILDING DOCKER IMAGE:

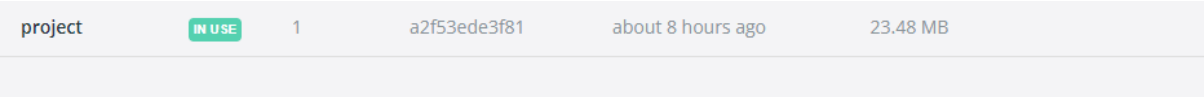
```
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\mmm\Downloads\cit-website\Cclab>docker build -t project:1 .
[+] Building 26.9s (7/7) FINISHED
 => [internal] load build definition from Dockerfile                                    2.0s
 => => transferring dockerfile: 84B                                                     0.4s
 => [internal] load .dockerignore                                                       1.6s
 => => transferring context: 2B                                                         0.2s
 => [internal] load metadata for docker.io/library/nginx:alpine                        23.1s
 => [1/2] FROM docker.io/library/nginx:alpine@sha256:da9c94bec1da829ebd52431a84502ec471c8e548ffb2cedbf36260fd9bd1  0.0s
 => [internal] load build context                                                       0.4s
 => => transferring context: 305B                                                       0.1s
 => CACHED [2/2] COPY . /usr/share/nginx/html                                           0.0s
 => exporting to image                                                                  0.5s
 => => exporting layers                                                                 0.0s
 => => writing image sha256:a2f53ede3f81c68fe55fe18a9339b8173f58a520a109e0333e5111ebdffece7b  0.1s
 => => naming to docker.io/library/project:1                                            0.0s
```

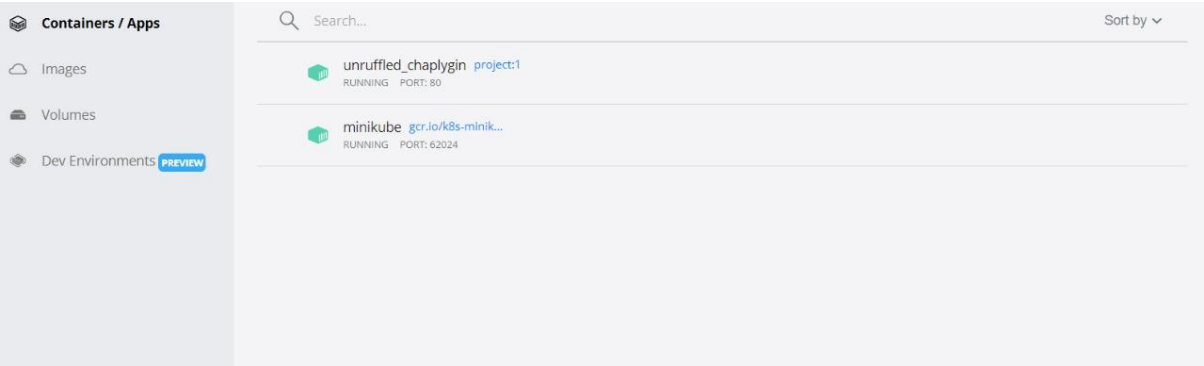
RUNNING DOCKER IMAGE:

```
C:\Users\mmm\Downloads\cit-website\Cclab>docker run -d -p 80:80 project:1
2caaf384f532e02f88802c526e7b004d1d0f6e6ffa406e752f6b70bd33caec8d
```

IMAGE CACHING:



LAYER CACHING:



CONTAINER SERVICE EXPOSURE:

CONTAINER LOG FETCHING:

```
C:\Users\mmm\Downloads\cit-website\Cclab>docker logs 2caaf384f532
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2022/01/29 14:59:58 [notice] 1#1: using the "epoll" event method
2022/01/29 14:59:58 [notice] 1#1: nginx/1.21.6
2022/01/29 14:59:58 [notice] 1#1: built by gcc 10.3.1 20211027 (Alpine 10.3.1_git20211027)
2022/01/29 14:59:58 [notice] 1#1: OS: Linux 5.10.16.3-microsoft-standard-WSL2
2022/01/29 14:59:58 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2022/01/29 14:59:58 [notice] 1#1: start worker processes
2022/01/29 14:59:58 [notice] 1#1: start worker process 33
2022/01/29 14:59:58 [notice] 1#1: start worker process 34
2022/01/29 14:59:58 [notice] 1#1: start worker process 35
2022/01/29 14:59:58 [notice] 1#1: start worker process 36
172.17.0.1 - - [29/Jan/2022:15:00:47 +0000] "GET / HTTP/1.1" 200 1141 "-" "curl/7.79.1" "-"
172.17.0.1 - - [29/Jan/2022:15:01:09 +0000] "GET / HTTP/1.1" 200 1141 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) App
leWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.99 Safari/537.36" "-"
2022/01/29 15:01:10 [error] 35#35: *2 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory),
client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost", referrer: "http://localh
ost/"
172.17.0.1 - - [29/Jan/2022:15:01:10 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost/" "Mozilla/5.0 (Window
s NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.99 Safari/537.36" "-"
```

EXECUTING INTO DOCKER:

```
C:\Users\mmm\Downloads\cit-website\Cclab>docker exec -it 2caaf384f532 sh
/ # ls
bin                  etc                  mnt                  run                  tmp
dev                  home                 opt                  sbin                 usr
docker-entrypoint.d  lib                  proc                 srv                  var
docker-entrypoint.sh media                root                 sys
/ # exit
```

KUBERNETES:

STARTING MINIKUBE:

```
C:\Users\mmm\Downloads\cit-website\Cclab>minikube version
minikube version: v1.25.1
commit: 3e64b11ed75e56e4898ea85f96b2e4af0301f43d

C:\Users\mmm\Downloads\cit-website\Cclab>minikube start
* minikube v1.25.1 on Microsoft Windows 10 Home Single Language 10.0.19043 Build 19043
* Automatically selected the docker driver
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
* Downloading Kubernetes v1.23.1 preload ...
    > preloaded-images-k8s-v16-v1...: 504.42 MiB / 504.42 MiB  100.00% 3.93 MiB
    > gcr.io/k8s-minikube/kicbase: 378.98 MiB / 378.98 MiB  100.00% 2.13 MiB p/
* Creating docker container (CPUs=2, Memory=2200MB) ...
! Executing "docker ps -a --format {{.Names}}" took an unusually long time: 4.6958702s
* Restarting the docker service may improve performance.
! Executing "docker container inspect minikube --format={{.State.Status}}" took an unusually long time: 6.6983369s
* Restarting the docker service may improve performance.
! StartHost failed, but will try again: creating host: create host timed out in 360.000000 seconds
* docker "minikube" container is missing, will recreate.
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Failed to start docker container. Running "minikube delete" may fix it: recreate: creating host: create host timed out
 in 360.000000 seconds

X Exiting due to DRV_CREATE_TIMEOUT: Failed to start host: recreate: creating host: create host timed out in 360.000000
seconds
* Suggestion: Try 'minikube delete', and disable any conflicting VPN or firewall software
* Related issue: https://github.com/kubernetes/minikube/issues/7072
```

CREATING NAMESPACE:

```
F:\cit-website\Cclab>kubectl apply -f ns.yml
namespace/cccit created

F:\cit-website\Cclab>kubectl get namespaces
NAME              STATUS   AGE
cccit             Active   15s
default           Active   21h
kube-node-lease   Active   21h
kube-public       Active   21h
kube-system       Active   21h
```

ns.yml

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: cccit
```

CREATING POD IN THE NAME SPACE :

```
F:\cit-website\Cclab>kubectl apply -f deployment.yaml
deployment.apps/cccit-deployment created
```

deployment.yaml

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: cccit
  name: cccit-deployment
  namespace: cccit
spec:
  replicas: 2
  selector:
    matchLabels:
      app: cccit
  template:
    metadata:
      labels:
        app: cccit
    spec:
      containers:
        - image: project:1
          imagePullPolicy: Never
          name: cccit
          ports:
            - containerPort: 80
```

CONFIGURING DEPLOYMENT POD WITH 2 REPLICAS :

```
F:\cit-website\Cclab>kubectl apply -f deployment.yaml
deployment.apps/cccit-deployment created
```

```
F:\cit-website\Cclab>kubectl apply -f deployment.yaml
deployment.apps/cccit-deployment created

F:\cit-website\Cclab>kubectl get deployments -n=cccit
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
cccit-deployment  2/2     2            2           11m

F:\cit-website\Cclab>kubectl get pods -n=cccit
NAME                                READY   STATUS    RESTARTS   AGE
cccit-deployment-5f7fbb8946-gpl89   1/1     Running   0          12m
cccit-deployment-5f7fbb8946-hbp22   1/1     Running   0          12m
```

SERVICE EXPOSURE(v1):

service.yaml

```yaml
apiVersion: v1
kind: Service
metadata:
  name: cccit-service
  namespace: cccit
spec:
  selector:
    app: cccit
  type: LoadBalancer
  ports:
    - protocol: TCP
      port: 8080
      targetPort: 80
      nodePort: 30000
```

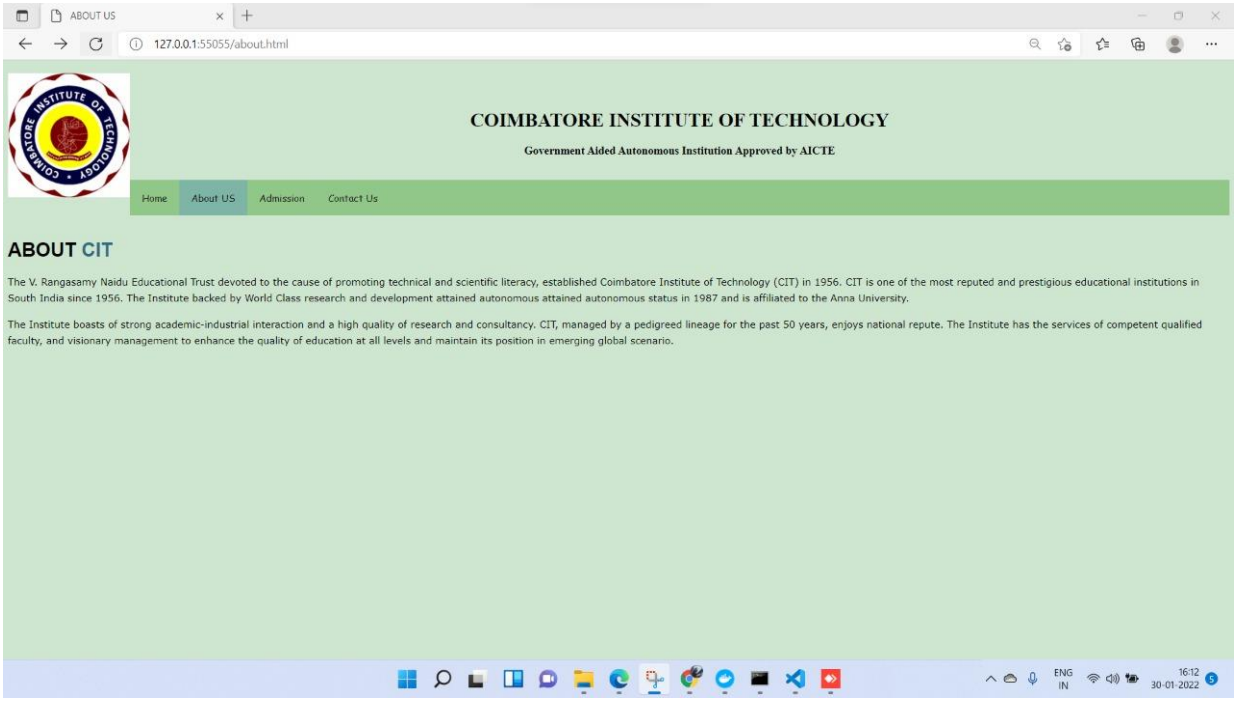
```
F:\cit-website\Cclab>kubectl apply -f service.yaml
service/cccit-service created

F:\cit-website\Cclab>kubectl get service
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   24h

F:\cit-website\Cclab>kubectl get service -n=cccit
NAME            TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
cccit-service   LoadBalancer   10.105.11.126   <pending>     8080:30000/TCP   33s
```

```
F:\cit-website\Cclab>minikube service cccit-service -n=cccit
! Executing "docker container inspect minikube --format={{.State.Status}}" took an unusually long time: 2.4886465s
* Restarting the docker service may improve performance.
|-----------|---------------|-------------|--------------------------|
| NAMESPACE |     NAME      | TARGET PORT |           URL            |
|-----------|---------------|-------------|--------------------------|
| cccit     | cccit-service |        8080 | http://192.168.49.2:30000 |
|-----------|---------------|-------------|--------------------------|
* Starting tunnel for service cccit-service.
|-----------|---------------|-------------|------------------------|
| NAMESPACE |     NAME      | TARGET PORT |          URL           |
|-----------|---------------|-------------|------------------------|
| cccit     | cccit-service |             | http://127.0.0.1:55055 |
|-----------|---------------|-------------|------------------------|
* Opening service cccit/cccit-service in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
```

HOSTED WEBPAGE :



RESOURCE LIMITATION :

```
F:\cit-website\Cclab>kubectl apply -f deployment.yaml
deployment.apps/cccit-deployment configured

F:\cit-website\Cclab>kubectl get deployment -n cccit
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
cccit-deployment   2/2     1            2           89m

F:\cit-website\Cclab>kubectl get pods -n cccit
NAME                               READY   STATUS              RESTARTS   AGE
cccit-deployment-568947ccdb-6j4gw   0/1     CreateContainerError   0          51s
cccit-deployment-5f7fbb8946-gpl89   1/1     Running             0          89m
cccit-deployment-5f7fbb8946-hbp22   1/1     Running             0          89m

F:\cit-website\Cclab>kubectl describe pod cccit-deployment-568947ccdb-6j4gw -n cccit
Name:         cccit-deployment-568947ccdb-6j4gw
Namespace:    cccit
Priority:     0
Node:         minikube/192.168.49.2
Start Time:   Sun, 30 Jan 2022 17:21:04 +0530
Labels:       app=cccit
              pod-template-hash=568947ccdb
Annotations:  <none>
Status:       Pending
IP:           172.17.0.5
IPs:
  IP:           172.17.0.5
Controlled By:  ReplicaSet/cccit-deployment-568947ccdb
```

```
Containers:
  cccit:
    Container ID:
    Image:          project:1
    Image ID:
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Waiting
      Reason:       CreateContainerError
    Ready:          False
    Restart Count:  0
    Limits:
      cpu:     100m
      memory:  1Mi
    Requests:
      cpu:        100m
      memory:     1Mi
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-4257p (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             False
  ContainersReady   False
  PodScheduled      True
Volumes:
  kube-api-access-4257p:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   Guaranteed
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age                    From               Message
  ----     ------     ----                   ----               -------
  Normal   Scheduled  4m27s                  default-scheduler  Successfully assigned cccit/cccit-deployment-568947ccdb-6j4gw to minikube
  Warning  Failed     2m12s (x12 over 4m13s)  kubelet            Error: Error response from daemon: Minimum memory limit allowed is 6MB
  Normal   Pulled     117s (x13 over 4m14s)  kubelet            Container image "project:1" already present on machine
```

deployment.yaml

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: cccit
  name: cccit-deployment
  namespace: cccit
spec:
  replicas: 2
  selector:
    matchLabels:
      app: cccit
  template:
    metadata:
      labels:
        app: cccit
    spec:
      containers:
        - image: project:1
          imagePullPolicy: Never
          name: cccit
          ports:
            - containerPort: 80

          resources:
            requests:
              cpu: 0.1
              memory: 1Mi
            limits:
              cpu: 0.1
              memory: 1Mi
```

ROLLBACK STRATEGY

VERSION 2 :

```
F:\cit-website\Cclab>docker build -t project:1 .
[+] Building 13.4s (7/7) FINISHED
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 31B
 => [internal] load metadata for docker.io/library/nginx:alpine
 => [internal] load build context
 => => transferring context: 1.22kB
 => CACHED [1/2] FROM docker.io/library/nginx:alpine@sha256:da9c94bec1da829ebd52431a84502ec471c8e548ffb2cedbf36260fd9bd1d4d3
 => [2/2] COPY . /usr/share/nginx/html
 => exporting to image
 => => exporting layers
 => => writing image sha256:fd4908ffabfdde68e51727ed5971a7996164ceb04c7ea8e9ce749f3d176f2ae2
 => => naming to docker.io/library/project:1

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

F:\cit-website\Cclab>kubectl apply -f deployment.yaml
deployment.apps/cccit-deployment configured

F:\cit-website\Cclab>kubectl get deployments -n cccit
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
cccit-deployment   2/2     1            2           99m

F:\cit-website\Cclab>minikube service cccit-service -n cccit
|-----------|---------------|-------------|---------------------------|
| NAMESPACE |     NAME      | TARGET PORT |            URL            |
|-----------|---------------|-------------|---------------------------|
| cccit     | cccit-service |        8080 | http://192.168.49.2:30000 |
|-----------|---------------|-------------|---------------------------|
* Starting tunnel for service cccit-service.
|-----------|---------------|-------------|------------------------|
| NAMESPACE |     NAME      | TARGET PORT |          URL           |
|-----------|---------------|-------------|------------------------|
| cccit     | cccit-service |             | http://127.0.0.1:56269 |
|-----------|---------------|-------------|------------------------|
* Opening service cccit/cccit-service in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
```