Cloud Computing CAT-2 Project

E Jaya Suriya   - 1934012
S Sanjay Krishna   - 1934036
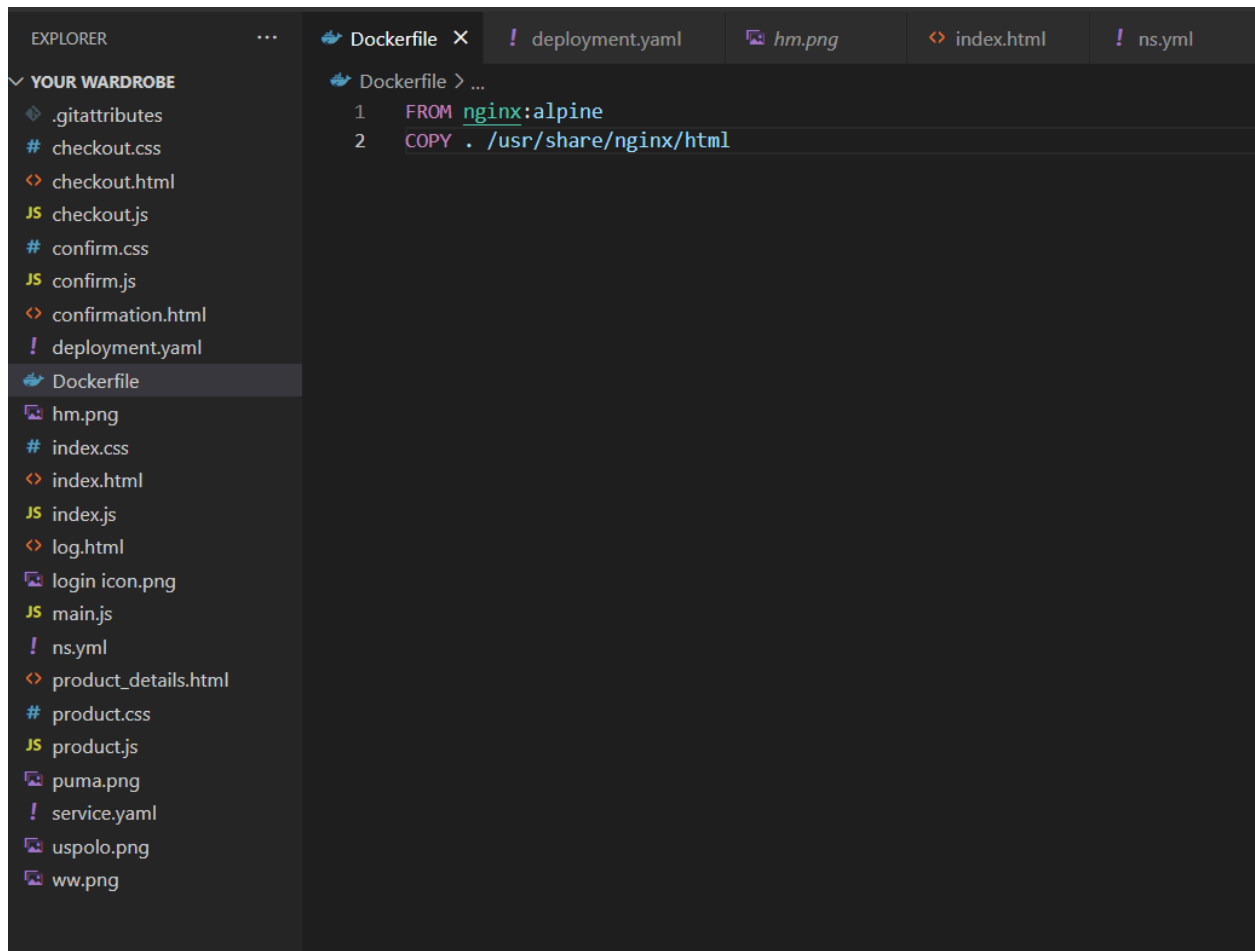S Sidharth   - 1934044

Problem Statement:
　　　　Many people around the world prefer to shop online and buy products from several brands and companies that they cannot find or are not available for purchase in their home countries.During this pandemic time purchasing at a local store comes in with the threat of Covid-19.

Proposed Solution:
　　　　Our's is an attempt to provide a prototype to an online shopping site for users to provide Convenience reducing the crowd and need for travel to purchase products using cloud services Docker containers and Kubernetes.

Dockerfile:


```dockerfile
FROM nginx:alpine
COPY . /usr/share/nginx/html
```

Building Docker Image:


```
91875@DRAGON MINGW64 ~/Downloads/FSWD PROJECT 44_17_51/copy/Your Wardrobe
$ docker build -t webserver-image:v1 .
[+] Building 7.0s (2/3)
 => [internal] load build definition from Dockerfile                                    0.0s
 => => transferring dockerfile: 84B                                                     0.0s
[+] Building 15.1s (8/8) FINISHED
 => => transferring dockerfile: 84B                        0.0s  => [internal] load build definition from Dockerfile    0.0s
 => => transferring context: 2B                            0.0s  => [internal] load .dockerignore                       0.0s
 => [auth] library/nginx:pull token for registry-1.docker.i 0.0s  => [internal] load metadata for docker.io/library/nginx:a  14.6s
 => [internal] load build context                         0.1s
 => => transferring context: 49.72kB                      0.1s
 => CACHED [1/2] FROM docker.io/library/nginx:alpine@sha256 0.0s
 => [2/2] COPY . /usr/share/nginx/html                    0.1s
 => exporting to image                                    0.2s
 => => exporting layers                                   0.1s
 => => writing image sha256:6aa9eb1b92e4a16b1a8e8fed07db9e3 0.0s
 => => naming to docker.io/library/webserver-image:v1     0.0s
```

Running Docker Image [Container Deployment] :


```
91875@DRAGON MINGW64 ~/Downloads/FSWD PROJECT 44_17_51/copy/Your Wardrobe
$ docker run -d -p 80:80 webserver-image:v1
7c3d23291d870daf5aa44a2e1bc60b3b9c339664e0ba0a7bb407b9b93b13ca04
```
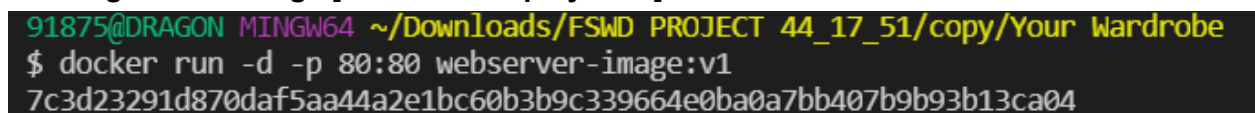
Image Caching:

webserver-image	IN USE	v1	6aa9eb1b92e4	19 minutes ago	23.51 MB	⋮	RUN ▶

Layer Caching:

‹ **webserver-image:...** IN USE

		IMAGE ID	CREATED	SIZE	More actions ⌄	RUN
		6aa9eb1b92e4	19 minutes ago	23.51 MB		

IMAGE HISTORY

0	COPY . /usr/share/nginx/html # buildkit	49 KB
1	/bin/sh -c #(nop) CMD ["nginx" "-g" "daemon off;"]	0 Bytes
2	/bin/sh -c #(nop) STOPSIGNAL SIGQUIT	0 Bytes
3	/bin/sh -c #(nop) EXPOSE 80	0 Bytes
4	/bin/sh -c #(nop) ENTRYPOINT ["/docker-entrypoint.sh"]	0 Bytes
5	/bin/sh -c #(nop) COPY file:09a214a3e07c919af2fb2d7c749ccbc446b8...	4.61 KB
6	/bin/sh -c #(nop) COPY file:0fd5fca330dcd6a7de297435e32af634f29f...	1.04 KB
7	/bin/sh -c #(nop) COPY file:0b866ff3fc1ef5b03c4e6c8c513ae014f691f...	1.96 KB
8	/bin/sh -c #(nop) COPY file:65504f71f5855ca017fb64d502ce873a31b2...	1.2 KB
9	/bin/sh -c set -x && addgroup -g 101 -S nginx && adduser -S -D -H -u ...	17.86 MB
10	/bin/sh -c #(nop) ENV PKG_RELEASE=1	0 Bytes
11	/bin/sh -c #(nop) ENV NJS_VERSION=0.7.1	0 Bytes

COMMAND

```
COPY . /usr/share/nginx/html # buildkit
```

Containers:[elegant_jackson;adoring_banach]

Containers / Apps

🔍 Search...   Sort by ⌄

- Images
- Volumes
- Dev Environments PREVIEW

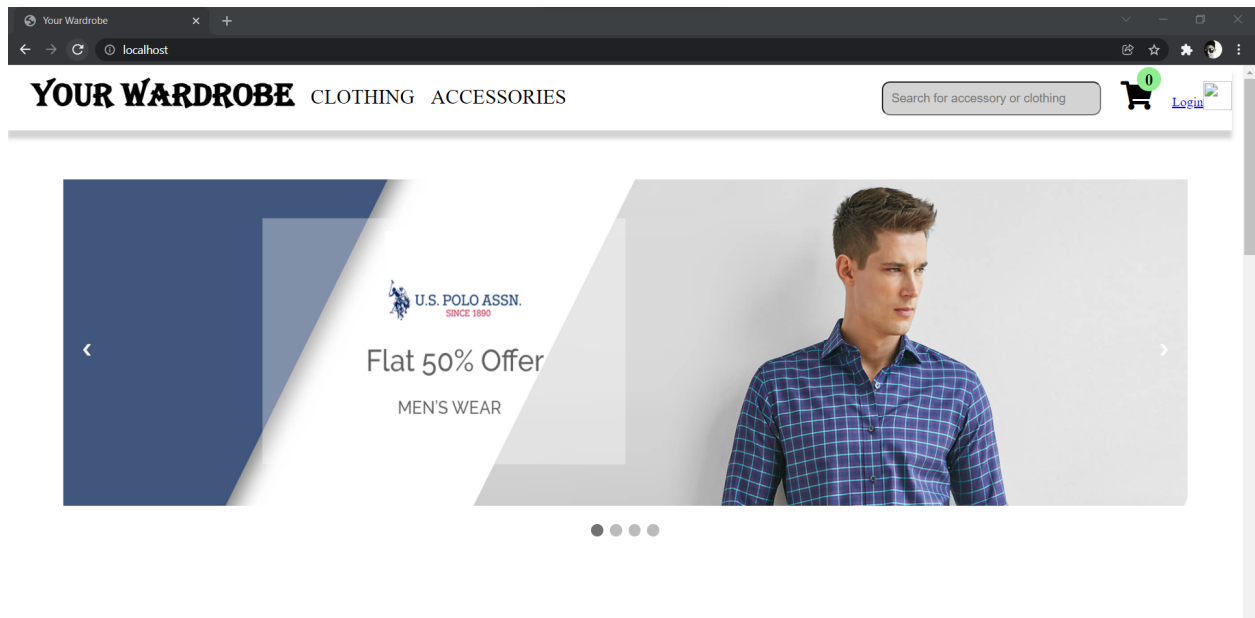|	hopeful_feistel d13c942271d6
	EXITED (127)

	elastic_lamarr hello-world
	EXITED (0)

	docker-tutorial docker101tutorial
	EXITED (255) PORT: 80

	repo alpine/git
	EXITED (128)

	elegant_jackson webserver-imag...
	RUNNING PORT: 80

	adoring_banach webserver-imag...
	RUNNING

Container Service Exposure:

We have hosted and exposed our container in localhost 80:80 port. The exposed server image is attached above.

Container Log Fetching:

```
91875@DRAGON MINGW64 ~/Downloads/FSWD PROJECT 44_17_51/copy/Your Wardrobe
$ docker logs f1e76c7a1d0b
/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
2022/01/25 06:50:21 [notice] 1#1: using the "epoll" event method
2022/01/25 06:50:21 [notice] 1#1: nginx/1.21.5
2022/01/25 06:50:21 [notice] 1#1: built by gcc 10.3.1 20211027 (Alpine 10.3.1_git20211027)
2022/01/25 06:50:21 [notice] 1#1: OS: Linux 5.10.16.3-microsoft-standard-WSL2
2022/01/25 06:50:21 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2022/01/25 06:50:21 [notice] 1#1: start worker processes
2022/01/25 06:50:21 [notice] 1#1: start worker process 33
2022/01/25 06:50:21 [notice] 1#1: start worker process 34
2022/01/25 06:50:21 [notice] 1#1: start worker process 35
2022/01/25 06:50:21 [notice] 1#1: start worker process 36
2022/01/25 06:50:21 [notice] 1#1: start worker process 37
2022/01/25 06:50:21 [notice] 1#1: start worker process 38
2022/01/25 06:50:21 [notice] 1#1: start worker process 39
2022/01/25 06:50:21 [notice] 1#1: start worker process 40
172.17.0.1 - - [25/Jan/2022:06:50:59 +0000] "GET / HTTP/1.1" 200 9912 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/97.0.4692.71 Safari/537.36" "-"
172.17.0.1 - - [25/Jan/2022:06:51:00 +0000] "GET /index.css HTTP/1.1" 200 4824 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/53
7.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
172.17.0.1 - - [25/Jan/2022:06:51:00 +0000] "GET /index.js HTTP/1.1" 200 2132 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537
.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
2022/01/25 06:51:06 [error] 33#33: *3 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhos
t, request: "GET /favicon.ico HTTP/1.1", host: "localhost", referrer: "http://localhost/"
172.17.0.1 - - [25/Jan/2022:06:51:06 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5
37.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
172.17.0.1 - - [25/Jan/2022:07:03:55 +0000] "GET / HTTP/1.1" 200 9912 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/97.0.4692.71 Safari/537.36" "-"
172.17.0.1 - - [25/Jan/2022:07:03:55 +0000] "GET /index.css HTTP/1.1" 200 4824 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/53
7.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
172.17.0.1 - - [25/Jan/2022:07:03:55 +0000] "GET /index.js HTTP/1.1" 200 2132 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537
.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
2022/01/25 07:03:59 [error] 33#33: *12 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localho
st, request: "GET /favicon.ico HTTP/1.1", host: "localhost", referrer: "http://localhost/"
172.17.0.1 - - [25/Jan/2022:07:03:59 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5
37.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36" "-"
```

Execute Into A Docker:

```
91875@DRAGON MINGW64 ~/Downloads/FSWD PROJECT 44_17_51/copy/Your Wardrobe
$ docker exec -it e2fc4ffe502b sh
/ # ls
bin                 docker-entrypoint.sh  lib          opt          run          sys          var
dev                 etc                   media        proc         sbin         tmp
docker-entrypoint.d home                  mnt          root         srv          usr
/ # exit
```

```
91875@DRAGON MINGW64 ~/Downloads/FSWD PROJECT 44_17_51/copy/Your Wardrobe
$ docker exec -it f1e76c7a1d0b sh
/ # ls
bin                 docker-entrypoint.sh  lib          opt          run          sys          var
dev                 etc                   media        proc         sbin         tmp
docker-entrypoint.d home                  mnt          root         srv          usr
/ # exit
```

Kubernetes:

Starting minikube:

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>minikube start
* minikube v1.25.1 on Microsoft Windows 11 Home Single Language 10.0.22000 Build 22000
* Using the docker driver based on existing profile
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
* Updating the running docker "minikube" container ...
* Preparing Kubernetes v1.23.1 on Docker 20.10.12 ...
  - kubelet.housekeeping-interval=5m
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```
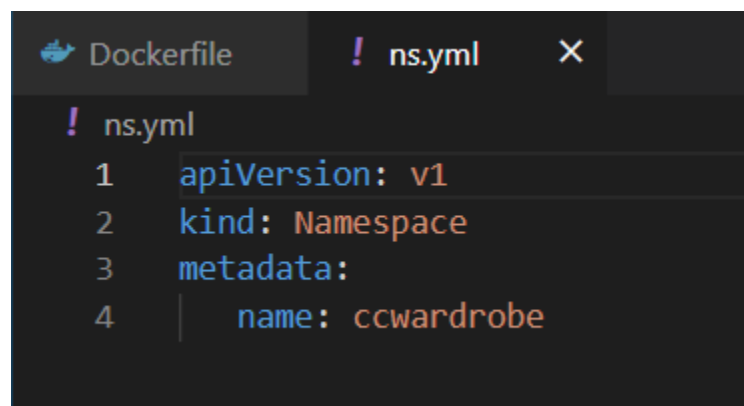
```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>minikube image load webserver-image:v1
```

Creating Namespace:

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get namespaces
NAME              STATUS    AGE
default           Active    17m
kube-node-lease   Active    17m
kube-public       Active    17m
kube-system       Active    17m

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl apply -f ns.yml
namespace/ccwardrobe created

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get namespaces
NAME              STATUS    AGE
ccwardrobe        Active    3s
default           Active    17m
kube-node-lease   Active    17m
kube-public       Active    17m
kube-system       Active    17m
```
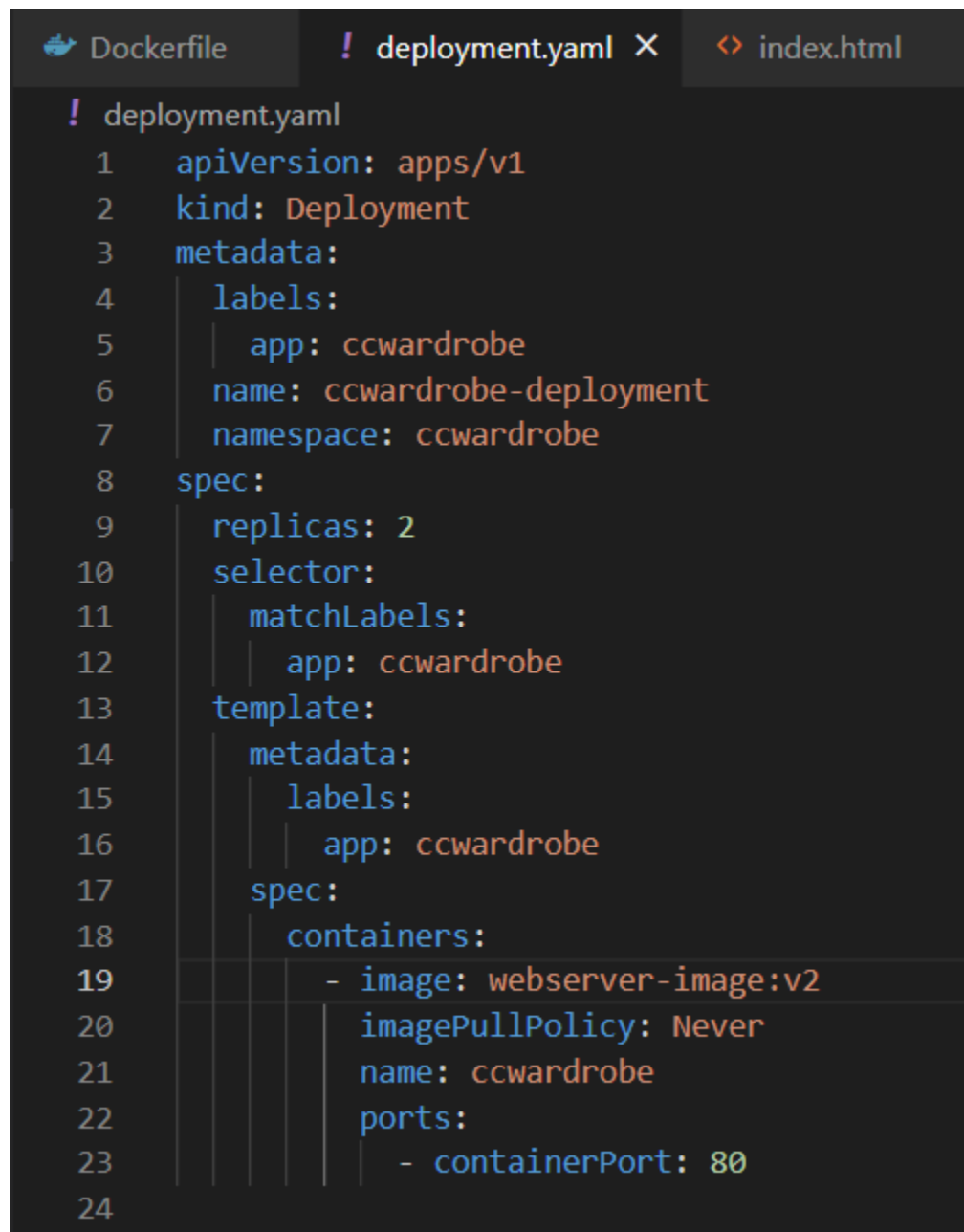
```
Dockerfile    ! ns.yml    X

! ns.yml
1   apiVersion: v1
2   kind: Namespace
3   metadata:
4       name: ccwardrobe
```

Deploying The Pod In The Namespace:

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl apply -f deployment.yaml
deployment.apps/ccwardrobe-deployment created
```

! deployment.yaml

```yaml
1    apiVersion: apps/v1
2    kind: Deployment
3    metadata:
4      labels:
5        app: ccwardrobe
6      name: ccwardrobe-deployment
7      namespace: ccwardrobe
8    spec:
9      replicas: 2
10     selector:
11       matchLabels:
12         app: ccwardrobe
13     template:
14       metadata:
15         labels:
16           app: ccwardrobe
17       spec:
18         containers:
19         - image: webserver-image:v2
20           imagePullPolicy: Never
21           name: ccwardrobe
22           ports:
23           - containerPort: 80
24
```

After we configured our deployment pod, we have created a new configured deployment with 2 replicas:

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get deployments -n=ccwardrobe
NAME                     READY    UP-TO-DATE    AVAILABLE    AGE
ccwardrobe-deployment    2/2      2             2            73m

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get pods -n=ccwardrobe
NAME                                    READY    STATUS     RESTARTS    AGE
ccwardrobe-deployment-d64cb9c88-fjhq6   1/1      Running    0           2m47s
ccwardrobe-deployment-d64cb9c88-vpjzz   1/1      Running    0           2m44s
```

Service Exposure(v1):

```
Dockerfile        ! deployment.yaml        ! service.yaml  ×

! service.yaml
1    apiVersion: v1
2    kind: Service
3    metadata:
4      name: ccwardrobe-service
5      namespace: ccwardrobe
6    spec:
7      selector:
8        app: ccwardrobe
9      type: LoadBalancer
10     ports:
11       - protocol: TCP
12         port: 8080
13         targetPort: 80
14         nodePort: 30000
15
```

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl apply -f service.yaml
```

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get service
NAME           TYPE        CLUSTER-IP     EXTERNAL-IP    PORT(S)      AGE
kubernetes     ClusterIP   10.96.0.1      <none>         443/TCP      49m

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get service -n=ccwardrobe
NAME                TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
ccwardrobe-service  LoadBalancer   10.103.67.51   <pending>      8080:30000/TCP   34m

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>minikube service ccwardrobe-service -n=ccwardrobe
|------------|--------------------|-------------|----------------------------|
| NAMESPACE  |        NAME        | TARGET PORT |            URL             |
|------------|--------------------|-------------|----------------------------|
| ccwardrobe | ccwardrobe-service |        8080 | http://192.168.49.2:30000  |
|------------|--------------------|-------------|----------------------------|
* Starting tunnel for service ccwardrobe-service.
|------------|--------------------|-------------|-----------------------|
| NAMESPACE  |        NAME        | TARGET PORT |          URL          |
|------------|--------------------|-------------|-----------------------|
| ccwardrobe | ccwardrobe-service |             | http://127.0.0.1:61178 |
|------------|--------------------|-------------|-----------------------|
* Opening service ccwardrobe/ccwardrobe-service in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
* Stopping tunnel for service ccwardrobe-service.
```

Here is our hosted webpage:
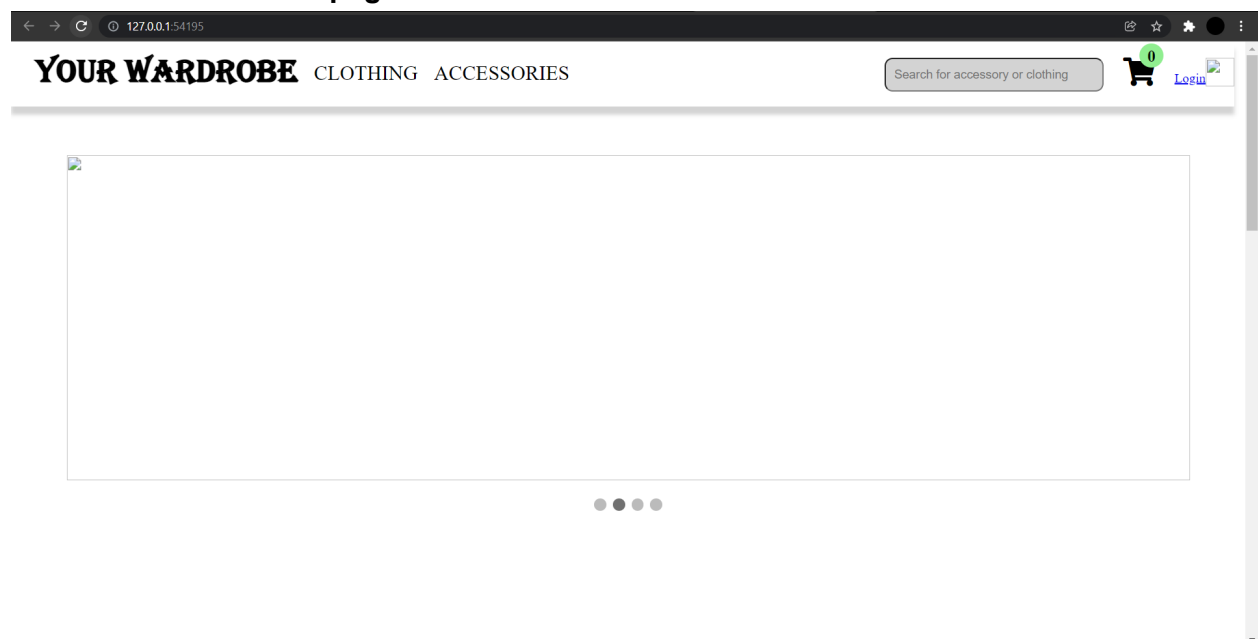
Resource Limitations:

```yaml
! deployment.yaml
1    apiVersion: apps/v1
2    kind: Deployment
3    metadata:
4      labels:
5        app: ccwardrobe
6      name: ccwardrobe-deployment
7      namespace: ccwardrobe
8    spec:
9      replicas: 2
10     selector:
11       matchLabels:
12         app: ccwardrobe
13     template:
14       metadata:
15         labels:
16           app: ccwardrobe
17       spec:
18         containers:
19           - image: webserver-image:v2
20             imagePullPolicy: Never
21             name: ccwardrobe
22             ports:
23               - containerPort: 80
24             resources:
25               requests:
26                 cpu: 0.1
27                 memory: 1Mi
28               limits:
29                 cpu: 0.1
30                 memory: 1Mi
```

```
  Limits:
    cpu:        100m
    memory:     1Mi
  Requests:
    cpu:        100m
    memory:     1Mi
  Environment:  <none>
  Mounts:
    /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-nrbtk (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             False
  ContainersReady   False
  PodScheduled      True
Volumes:
  kube-api-access-nrbtk:
    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  102s                default-scheduler  Successfully assigned ccwardrobe/ccwardrobe-deployment-7b5cfcdb7-lws44 to minikube
  Normal   Pulled     10s (x9 over 101s)  kubelet            Container image "webserver-image:v2" already present on machine
  Warning  Failed     10s (x9 over 101s)  kubelet            Error: Error response from daemon: Minimum memory limit allowed is 6MB
```

Roll-Back Strategy:

Version 1:
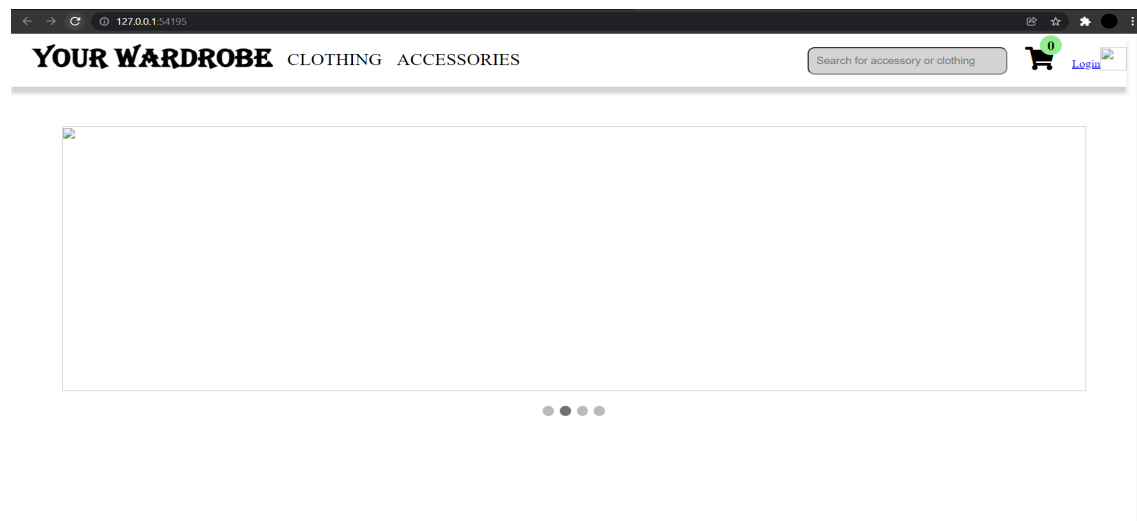
```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>docker build -t webserver-image:v2 .
[+] Building 20.5s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 31B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load build context
=> => transferring context: 2.85MB
=> [1/2] FROM docker.io/library/nginx:alpine@sha256:da9c94bec1da829ebd52431a84502ec471c8e548ffb2cedbf36260fd9bd1d4d3
=> => resolve docker.io/library/nginx:alpine@sha256:da9c94bec1da829ebd52431a84502ec471c8e548ffb2cedbf36260fd9bd1d4d3
=> => sha256:050385609d832fae11b007fbbfba77d0bba12bf72bc0dca0ac03e09b1998580f 1.57kB / 1.57kB
=> => sha256:bef258acf10dc257d641c47c3a600c92f87be4b4ce4a5e4752b3eade7533dcd9 8.89kB / 8.89kB
=> => sha256:5288d7ad7a7f84bdd19c1e8f0abb8684b5338f3da86fe9ae1d7f0e9bc2de6595 601B / 601B
=> => sha256:39e51c61c033442d00c40a30b2a9ed01f40205875fbd8664c50b4dc3e99ad5cf 894B / 894B
=> => sha256:da9c94bec1da829ebd52431a84502ec471c8e548ffb2cedbf36260fd9bd1d4d3 1.65kB / 1.65kB
=> => sha256:8d6ba530f6489d12676d7f61628427d067243ba4a3a512c3e28813b977cb3b0e 7.34MB / 7.34MB
=> => sha256:ee6f71c6f4a82b2afd01f92bdf6be0079364d03020e8a2c569062e1c06d3822b 665B / 665B
=> => sha256:f2303c6c88653b9a6739d50f611c170b9d97d161c6432409c680f6b46a5f112f 1.39kB / 1.39kB
=> => extracting sha256:8d6ba530f6489d12676d7f61628427d067243ba4a3a512c3e28813b977cb3b0e
=> => extracting sha256:5288d7ad7a7f84bdd19c1e8f0abb8684b5338f3da86fe9ae1d7f0e9bc2de6595
=> => extracting sha256:39e51c61c033442d00c40a30b2a9ed01f40205875fbd8664c50b4dc3e99ad5cf
=> => extracting sha256:ee6f71c6f4a82b2afd01f92bdf6be0079364d03020e8a2c569062e1c06d3822b
=> => extracting sha256:f2303c6c88653b9a6739d50f611c170b9d97d161c6432409c680f6b46a5f112f
=> [2/2] COPY . /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> => writing image sha256:1600250b0af601223b26a1804e59245f13f9866fe50db1b3c86339a92781dc33
=> => naming to docker.io/library/webserver-image:v2

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

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl apply -f deployment.yaml
deployment.apps/ccwardrobe-deployment configured

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl get deployment -n=ccwardrobe
NAME                     READY    UP-TO-DATE    AVAILABLE    AGE
ccwardrobe-deployment    2/2      1             2            59m

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>minikube service ccwardrobe-service -n=ccwardrobe
! Executing "docker container inspect minikube --format={{.State.Status}}" took an unusually long time: 2.065985s
* Restarting the docker service may improve performance.
|------------|----------------------|--------------|----------------------------|
| NAMESPACE  |         NAME         | TARGET PORT  |            URL             |
|------------|----------------------|--------------|----------------------------|
| ccwardrobe | ccwardrobe-service   |        8080  | http://192.168.49.2:30000  |
```

```
|------------|----------------------|--------------|----------------------------|
* Starting tunnel for service ccwardrobe-service.
|------------|----------------------|--------------|----------------------------|
| NAMESPACE  |         NAME         | TARGET PORT  |            URL             |
|------------|----------------------|--------------|----------------------------|
| ccwardrobe | ccwardrobe-service   |              | http://127.0.0.1:54142     |
|------------|----------------------|--------------|----------------------------|
* Opening service ccwardrobe/ccwardrobe-service in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
* Stopping tunnel for service ccwardrobe-service.
```
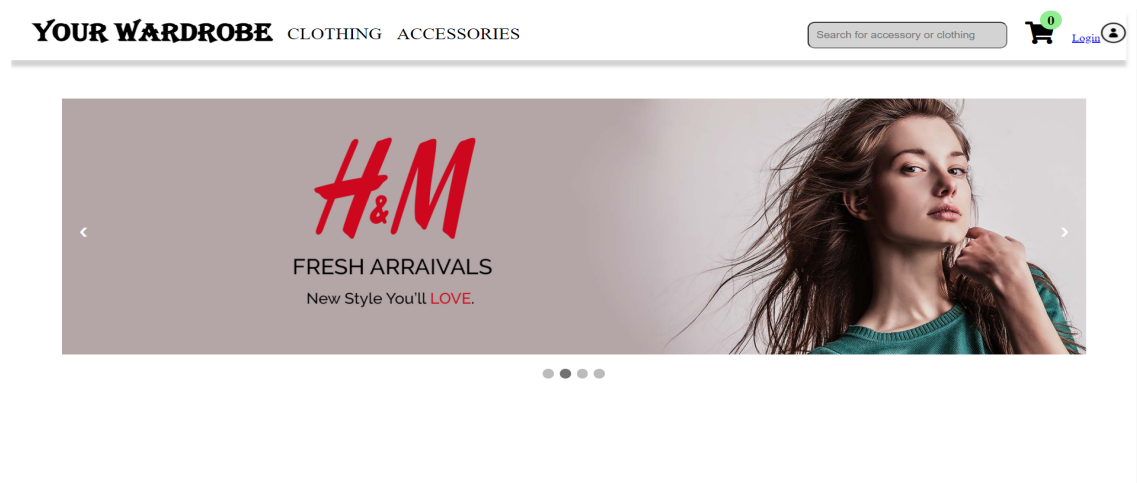
Version 2:

Rolling Back To Version 1:

```
C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl rollout history deploy ccwardrobe-deployment -n ccwardrobe
deployment.apps/ccwardrobe-deployment
REVISION   CHANGE-CAUSE
1          <none>
2          <none>
3          <none>


C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>kubectl rollout undo deploy ccwardrobe-deployment --to-revision=2 -n ccwardrobe
deployment.apps/ccwardrobe-deployment rolled back

C:\Users\91875\Downloads\FSWD PROJECT 44_17_51\copy\Your Wardrobe>minikube service ccwardrobe-service -n=ccwardrobe
! Executing "docker container inspect minikube --format={{.State.Status}}" took an unusually long time: 2.1128996s
* Restarting the docker service may improve performance.
|------------|--------------------|--------------|----------------------------|
| NAMESPACE  |        NAME        | TARGET PORT  |            URL             |
|------------|--------------------|--------------|----------------------------|
| ccwardrobe | ccwardrobe-service |        8080  | http://192.168.49.2:30000  |
|------------|--------------------|--------------|----------------------------|
* Starting tunnel for service ccwardrobe-service.
|------------|--------------------|--------------|------------------------|
| NAMESPACE  |        NAME        | TARGET PORT  |          URL           |
|------------|--------------------|--------------|------------------------|
| ccwardrobe | ccwardrobe-service |              | http://127.0.0.1:57900 |
|------------|--------------------|--------------|------------------------|
* Opening service ccwardrobe/ccwardrobe-service in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
* Stopping tunnel for service ccwardrobe-service.
```