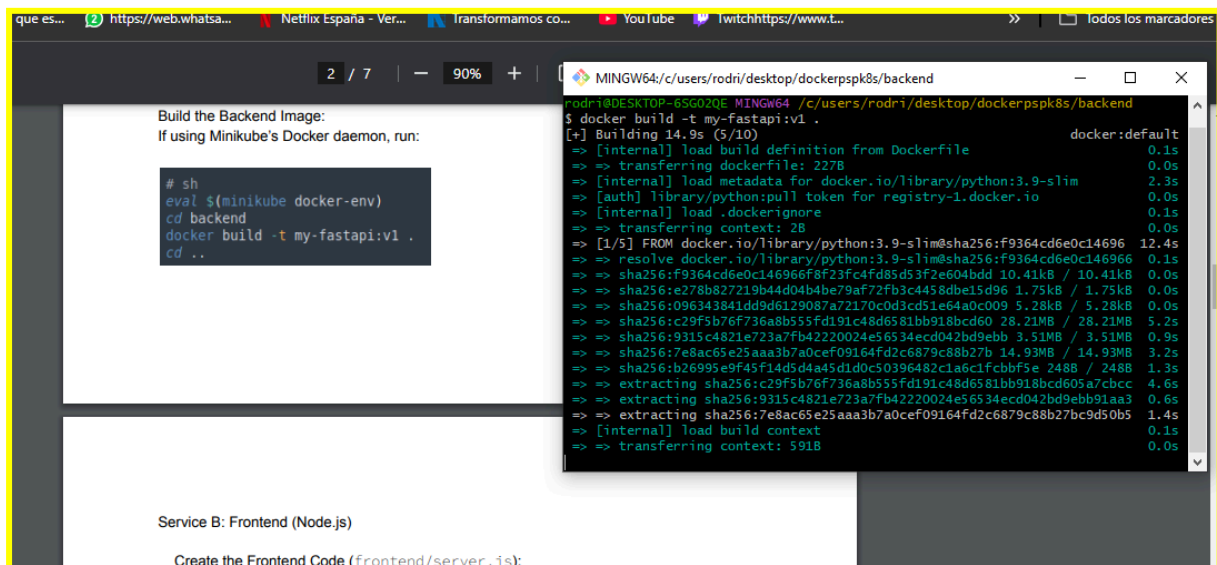


Part 1: Set Up Your Kubernetes Environment

Done ☒

Part 2: Create and Containerize Your Services

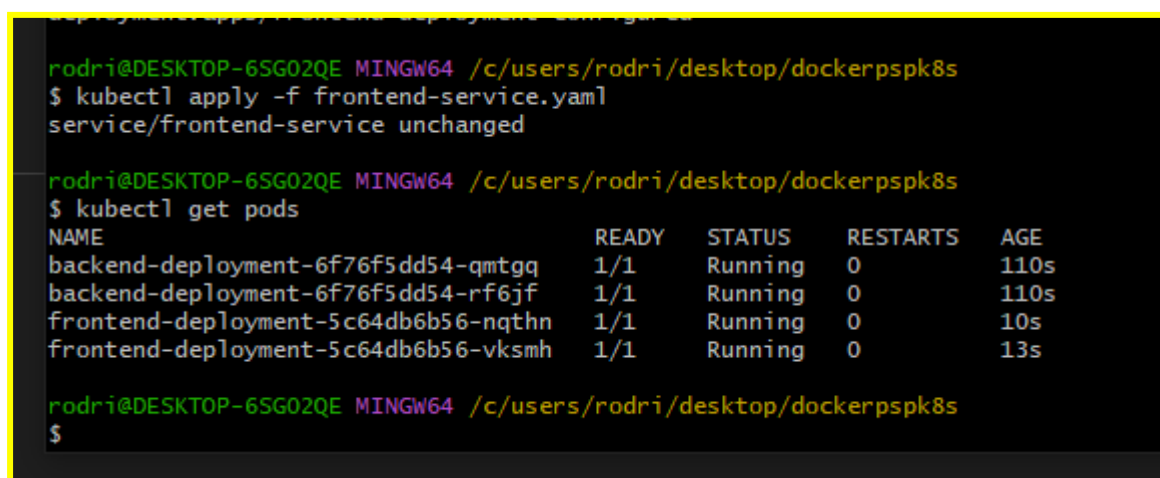


The screenshot shows a Windows terminal window with a yellow border. On the left, there's a dark-themed window titled 'Build the Backend Image:' with instructions to use Minikube's Docker daemon. It contains a code block with the following commands:

```
# sh
eval $(minikube docker-env)
cd backend
docker build -t my-fastapi:v1 .
cd ..
```

On the right, a standard Windows command prompt window shows the execution of the Docker build command. The output indicates a successful build of the 'my-fastapi:v1' image.

```
rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s/backend
$ docker build -t my-fastapi:v1 .
[+] Building 14.9s (5/10) docker:default
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 227B 0.0s
=> [internal] load metadata for docker.io/library/python:3.9-slim 2.3s
=> [auth] library/python:pull token for registry-1.docker.io 0.0s
=> [internal] load .dockerignore 0.1s
=> => transferring context: 2B 0.0s
=> [1/5] FROM docker.io/library/python:3.9-slim@sha256:f9364cd6e0c14696 12.4s
=> => resolve docker.io/library/python:3.9-slim@sha256:f9364cd6e0c14696 0.1s
=> sha256:f9364cd6e0c14696f8f23fc4fd85d53f2e604bdd 10.41kB / 10.41kB 0.0s
=> sha256:e278b827219b44d04b4be79af72fb3c4458dbe15d96 1.75kB / 1.75kB 0.0s
=> sha256:096343841dd9d6129087a72170c0d3cd5e64a0c009 5.28kB / 5.28kB 0.0s
=> sha256:c29f5b76f736a8b555fd191c48d6581bb918bcd60 28.21MB / 28.21MB 5.2s
=> sha256:9315c4821e723a7fb42220024e56534ecd042bd9ebb 3.51MB / 3.51MB 0.9s
=> sha256:7e8ac65e25aaa3b7a0cef09164fd2c6879c88b27b 14.93MB / 14.93MB 3.2s
=> sha256:b26995e9f45f14d5d4a45d1d0c50396482c1a6c1fcbbf5e 248B / 248B 1.3s
=> extracting sha256:c29f5b76f736a8b555fd191c48d6581bb918bcd605a7cbcc 4.6s
=> extracting sha256:9315c4821e723a7fb42220024e56534ecd042bd9ebb91aa3 0.6s
=> extracting sha256:7e8ac65e25aaa3b7a0cef09164fd2c6879c88b27bc9d50b5 1.4s
=> [internal] load build context 0.1s
=> => transferring context: 591B 0.0s
```



The screenshot shows a Windows terminal window with a yellow border. It displays the execution of two Kubernetes commands. The first command, 'kubectl apply -f frontend-service.yaml', shows that the service is unchanged. The second command, 'kubectl get pods', displays a table of running pods.

```
rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s
$ kubectl apply -f frontend-service.yaml
service/frontend-service unchanged

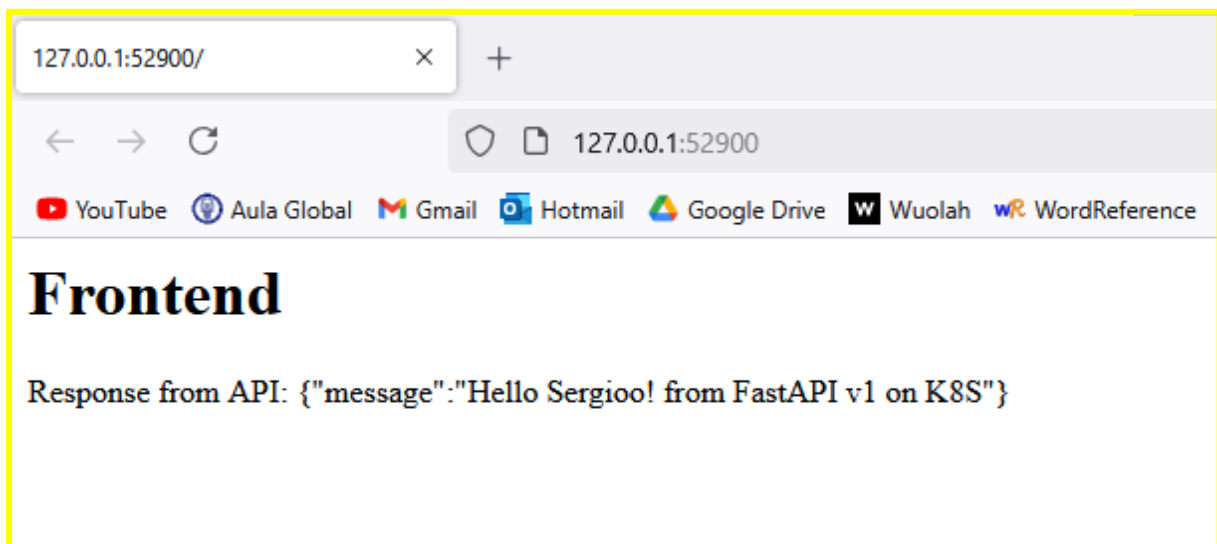
rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s
$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
backend-deployment-6f76f5dd54-qmtgq 1/1     Running   0           110s
backend-deployment-6f76f5dd54-rf6jf 1/1     Running   0           110s
frontend-deployment-5c64db6b56-nqthn 1/1     Running   0           10s
frontend-deployment-5c64db6b56-vksmh 1/1     Running   0           13s

rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s
$
```

Part 3: Deploy Services to Kubernetes

```
rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s
$ minikube service frontend-service --url
http://127.0.0.1:59951
! Porque estás usando controlador Docker en windows, la terminal debe
ra ejecutarlo.
```

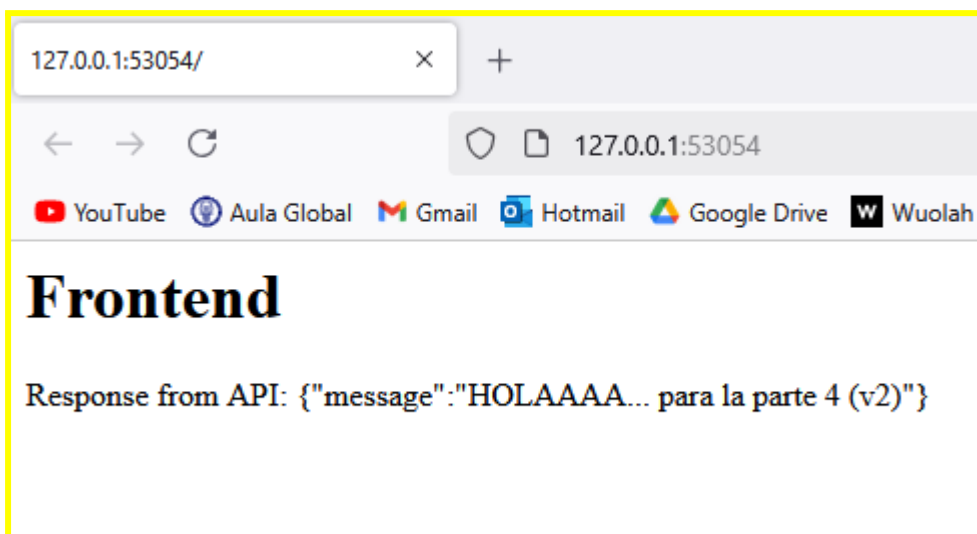
Open the URL in your browser



Part 4:

1. Modify the Backend API Code for an Upgrade Edit backend/app.py so that the API returns a different message (for example, indicating version 2):

```
rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/  
$ minikube service frontend-service --url  
http://127.0.0.1:53054  
! Porque estás usando controlador Docker en windows,  
necesitas usar minikube para ejecutarlo.
```



```
d rodri@DESKTOP-6SG02QE MINGW64 /c/users/rodri/desktop/dockerpspk8s  
$ kubectl rollout status deployment/backend-deployment  
deployment "backend-deployment" successfully rolled out
```