

## INSTRUCTIONS FOR TASK A1.1

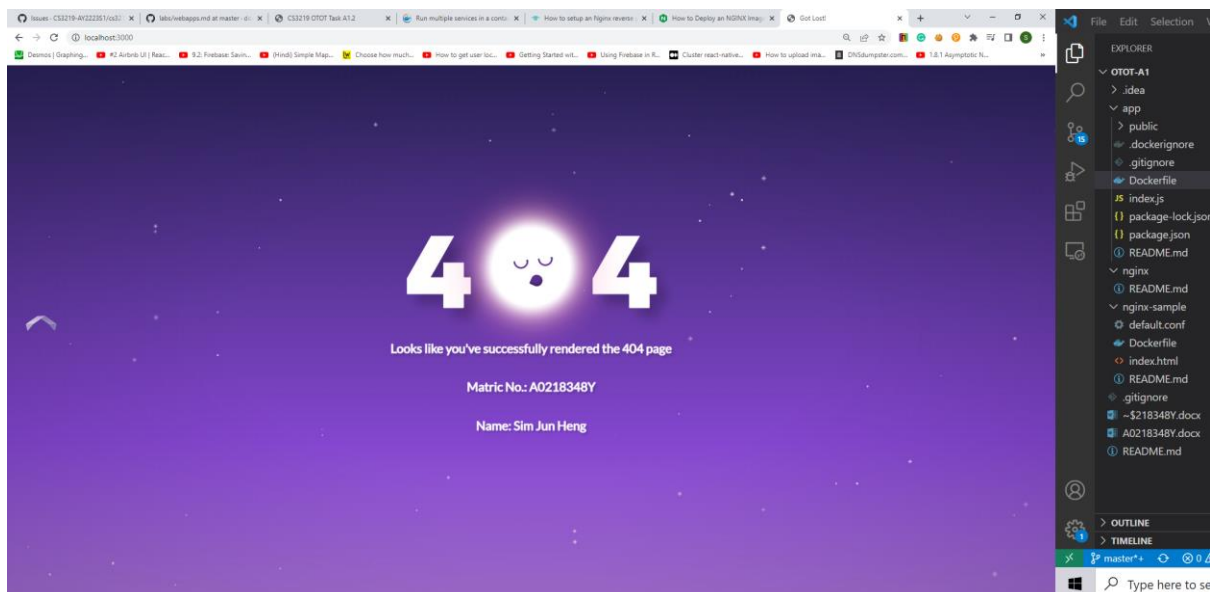
Step 1: Create a Dockerfile with the following contents in the app folder.

```
1 # Base image with node.js and NPM installed
2 FROM node:16
3
4 # Bundle app source
5 # Ensure that package.json and package-lock.json is copied
6 COPY . .
7
8 # Install all the modules and their dependencies listed in package.json
9 RUN npm install
10
11 # Binds my application to port 8080
12 EXPOSE 8080
13
14 # Start my application
15 CMD ["node", "index.js"]
```

Step 2: Run the command **docker build -t simjunheng/myfirstapp .**

Step 3: Run the command **docker run -p 3000:8080 --name myfirstapp simjunheng/myfirstapp**

Step 4: Go to your browser, type localhost:3000 in your url and you should see the following webpage.



## INSTRUCTIONS FOR TASK A1.2

Step 1: Create default.conf file with the following contents in the nginx-sample folder

```
server {
    listen      80;
    listen     [::]:80;
    server_name localhost;

    #access_log /var/log/nginx/host.access.log  main;

    location / {
        root    /usr/share/nginx/html;
        index   index.html index.htm;
    }

    #error_page  404              /404.html;

    # redirect server error pages to the static page /50x.html
    #
    error_page   500 502 503 504  /50x.html;
    location = /50x.html {
        root    /usr/share/nginx/html;
    }
}
```

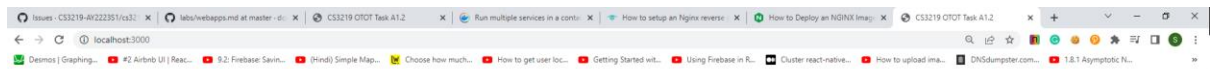
Step 2: Create a Dockerfile with the following contents in the nginx-sample folder

```
nginx-sample > Dockerfile
1  # Base NGINX image
2  FROM nginx:alpine
3
4  COPY index.html /usr/share/nginx/html/index.html
5
6  COPY default.conf /etc/nginx/conf.d/default.conf
7
8  EXPOSE 80
9
```

Step 3: Run the command **docker build -t simjunheng/nginx-sample .**

Step 4: Run the command **docker run -p 3000:80 --name nginx-sample simjunheng/nginx-sample**

Step 5: Go to your browser, type localhost:3000 in your url and you should see the following webpage.



**Nginx reverse proxy server listening on port 80 for incoming HTTP request.**

Done by: Sim Tan Heng

## INSTRUCTIONS FOR TASK A1.3

Step 1: Create the docker-compose.yml file with the following contents under the nginx folder

```
nginx > docker-compose.yml
1  version: "3.9"
2  services:
3    web:
4      image: node:16
5      ports:
6        - 4000:8080
7      volumes:
8        - C:\Users\sjh_9\Desktop\NUS_Y3S1\CS3219_Software_Engineering_Principles\Assignments\OTOT-A1\app:/app
9      command: bash -c "cd app && npm install && node index.js"
10
11   nginx:
12     image: nginx:alpine
13     ports:
14       - 3000:80
15     volumes:
16       - C:\Users\sjh_9\Desktop\NUS_Y3S1\CS3219_Software_Engineering_Principles\Assignments\OTOT-A1\nginx\default.conf:/etc/nginx/conf.d/default.conf
```

Step 2: Create the default.conf file with the following contents under the nginx folder

```
server {
    listen 80;
    listen [::]:80;
    server_name localhost;

    #access_log /var/log/nginx/host.access.log main;

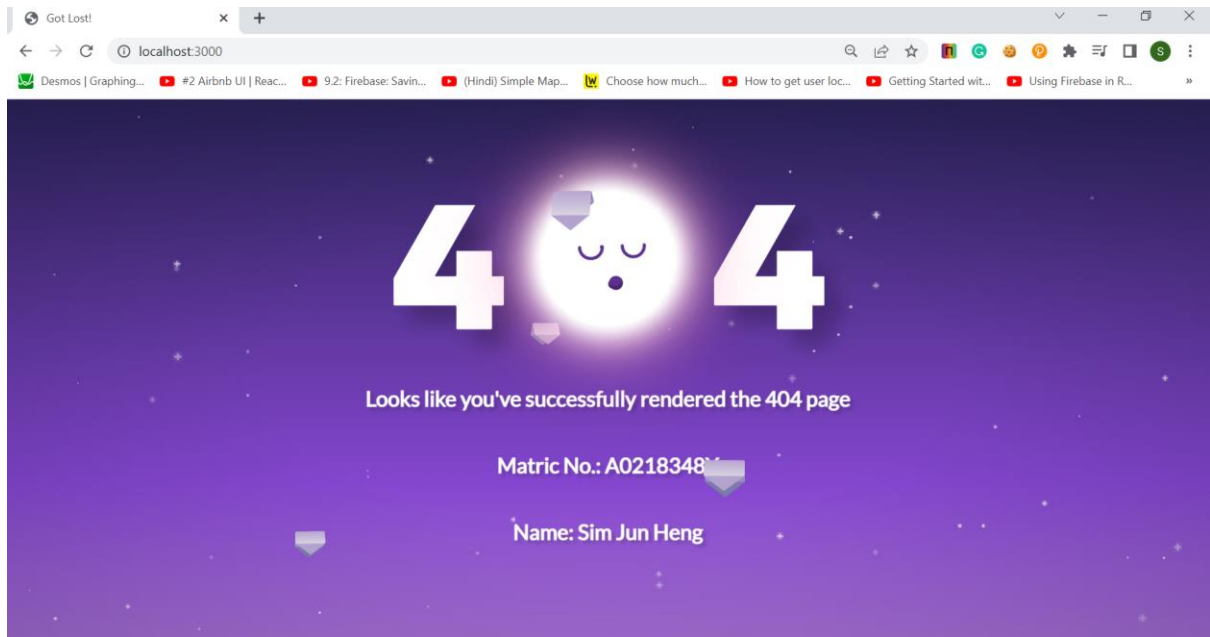
    location / {
        root /usr/share/nginx/html;
        proxy_pass http://web:8080/;
    }

    #error_page 404 /404.html;

    # redirect server error pages to the static page /50x.html
    #
    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }
}
```

Step 3: Run the command **docker compose up**

Step 4: Go to your browser, type localhost:3000 in your url and you should see the following webpage.



More explanation of how proxy works

