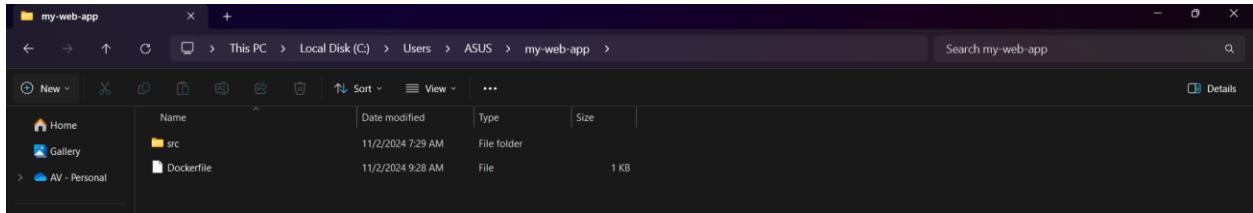


# Report

## Part I

### Folder structure



### b) html code

```

<html>
  <head>
    <title>Cloud Application Development</title>
    <style>
      body {
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
        margin: 0;
        font-family: Arial, sans-serif;
        background-color: #eef7f9; /* Light blue background */
      }
      div {
        text-align: center;
        font-size: 2.5em; /* Bigger font size */
        color: #00ad40; /* Dark teal text color */
      }
    </style>
  </head>
  <body>
    <div>
      <p>vishwa darshana-22ug1-0460</p>
      <p>CCS3312 - Cloud Application Development</p>
      <p>Third Year Second Semester</p>
      <p>BSc Hons in Data Science</p>
      <p>Sri Lanka Technological Campus</p>
    </div>
  </body>
</html>

```



## Part II

### a) Docker setup

```

File Edit Selection View Go Run Terminal Help ⌘ my-web-app
Dockerfile indexhtml
Dockerfile > ...
1 # Use an official Python image as the base image
2 FROM python:3.8-slim
3
4 # Set the working directory in the container
5 WORKDIR /app
6
7 # Copy the content of the folder to the working directory
8 COPY src /app/
9
10 # Install Flask to serve the application
11 RUN pip install flask
12
13 # Define the environment variable to specify the Flask app
14 ENV FLASK_APP=index.html
15
16 # Expose port 5000 to be accessible outside the container
17 EXPOSE 5000
18
19 # Command to run the Flask development server
20 CMD ["python", "-m", "http.server", "5000"]
21

```

OUTLINE TIMELINE

In 14, Col 25 Spaces: 4 UTT-8 CRLF Dockerfile Go Live

### b) docker build -t my-web-app .( create the Docker image)

```

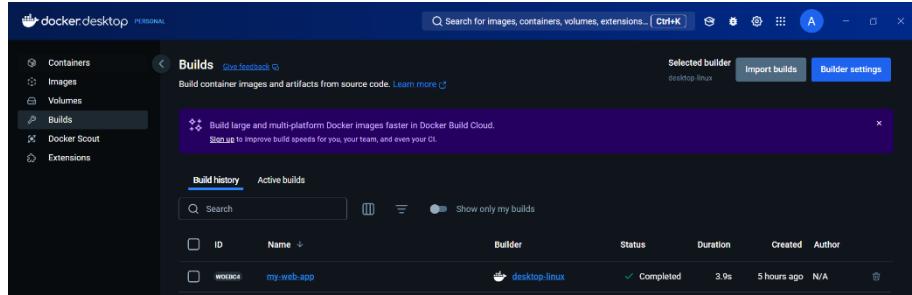
Microsoft Windows [Version 10.0.22631.4391]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\my-web-app>docker build -t my-web-app .
[+] Building 38.3s (9/10)
=> [internal] load build definition from Dockerfile
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load .dockerignore
=> transferring context: 2B
=> [1/4] FROM docker.io/library/python:3.8-slim@sha256:1d52838af602b4b5a831beb13a0e4d073280665ea7be7f69ce2382f29 0.1s
=> => resolve docker.io/library/python:3.8-slim@sha256:1d52838af602b4b5a831beb13a0e4d073280665ea7be7f69ce2382f29 0.0s
=> [internal] load build context
=> => transferring context: 1.02kB
[+] Building 38.4s (9/10)
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 574B
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/4] FROM docker.io/library/python:3.8-slim@sha256:1d52838af602b4b5a831beb13a0e4d073280665ea7be7f69ce2382f29 0.1s
=> => resolve docker.io/library/python:3.8-slim@sha256:1d52838af602b4b5a831beb13a0e4d073280665ea7be7f69ce2382f29 0.0s
=> [internal] load build context
=> => transferring context: 1.02kB
[+] Building 38.6s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 574B
=> [internal] load metadata for docker.io/library/python:3.8-slim 18.6s
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/4] FROM docker.io/library/python:3.8-slim@sha256:1d52838af602b 0.1s
=> => resolve docker.io/library/python:3.8-slim@sha256:1d52838af602b 0.0s
=> [internal] load build context
=> => transferring context: 1.02kB
=> [2/4] WORKDIR /app
=> [3/4] COPY src /app/
=> [4/4] RUN pip install Flask
=> exporting image
=> => exporting layers
=> => exporting manifest sha256:935595979a5da4026257727b5403a2aefad5 0.0s
=> => exporting config sha256:c5a256d5e6eca072fa99d1129739bf56a85fb 0.4s
=> => exporting attestation manifest sha256:b6d66d090fa138ea29255f42 0.0s
=> => exporting manifest list sha256:43c7262205b4b17de8c720f49018d18 0.0s
=> => naming to docker.io/library/my-web-app:latest 0.0s
=> => unpacking to docker.io/library/my-web-app:latest 0.3s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/10bbqntw5gvaokhqvqv5ua7w
What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview

```

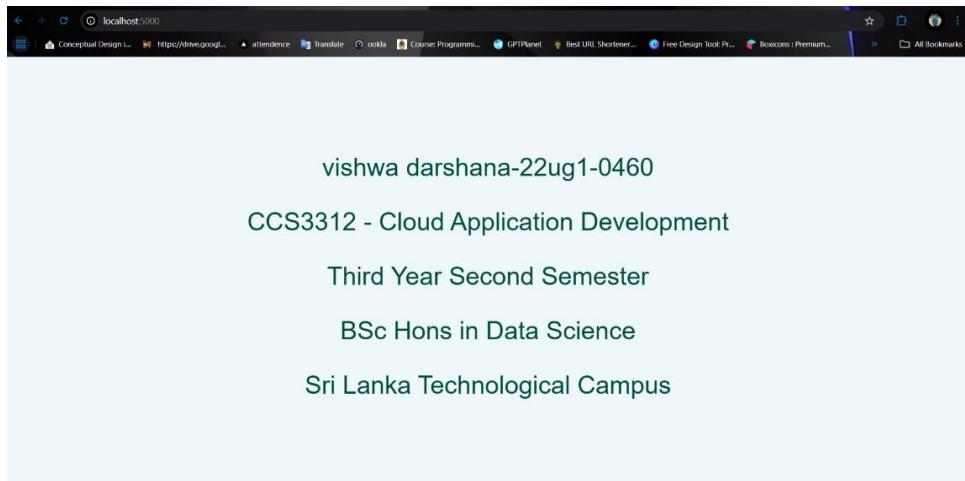
## docker application



c) **docker run -d -p 5000:5000 my-web-app**(Once you build the image, run)

```
C:\Users\ASUS\my-web-app>docker run -d -p 5000:5000 my-web-app
16abca433e525c2834a49495e2f87d083c55387469b6f069f11ff7f9f6d4a0af
```

d) "http://localhost:5000" to view your web page.



e) create a Docker account.

The screenshot shows the Docker Hub interface. At the top, there are navigation links: Explore, Repositories (which is underlined), Organizations, and Usage. A search bar says 'Search Docker Hub'. On the right, there's a user profile section with a large blue circular icon containing a white letter 'A'. Below the icon, the username 'avdavd' is displayed. To the right of the profile are links for 'What's new', 'My profile', 'Account settings', and 'Billing'. At the bottom right of the profile area is a 'Sign out' link.

f) tag your Docker image.

**docker tag my-web-app avdavd/my-web-app**

```
C:\Users\ASUS\my-web-app>docker tag my-web-app avdavd/my-web-app
```

g) push your docker image to docker hub.

**docker push avdavd/my-web-app**

my-web-app repository under your account.

The screenshot shows the Docker Hub interface again. The top navigation bar includes 'Explore', 'Repositories' (underlined), 'Organizations', and 'Usage'. A search bar says 'Search Docker Hub'. On the left, there's a dropdown menu showing 'avdavd' and a search bar for 'Search by repository name'. On the right, there's a 'Create repository' button. Below the search bar, the repository 'avdavd / my-web-app' is listed. It shows 'Contains: Image' and 'Last pushed: about 3 hours ago'. To the right of the repository listing are icons for '0 stars', '5 forks', 'Public', and 'Scout inactive'.

h) verify your Docker image on Docker Hub.

use friend's device for this part

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4391]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User>docker pull avdavd/my-web-app
Using default tag: latest
latest: Pulling from avdavd/my-web-app
5e227ea6cc2f: Download complete
8b4500955bcf: Download complete
83e108643cc9: Download complete
Digest: sha256:43c7262205b4b17de8c720f49018d184cfe8265df953a898874a9aa1637f76a8
Status: Downloaded newer image for avdavd/my-web-app:latest
docker.io/avdavd/my-web-app:latest

What's next:
  View a summary of image vulnerabilities and recommendations + docker scout quickview avdavd/my-web-app

C:\Users\User>docker run -d -p 5000:5000 avdavd/my-web-app
f53de40ab2623c815bdac1586183293356f6cie867526f40e8e95fc71ba0f5b

C:\Users\User>
```

your friend to open their web browser and check the output.

