DISTRIBUTED SYSTEMS

PROJECT PHASE – I

REPORT

Frameworks & Programming Language:

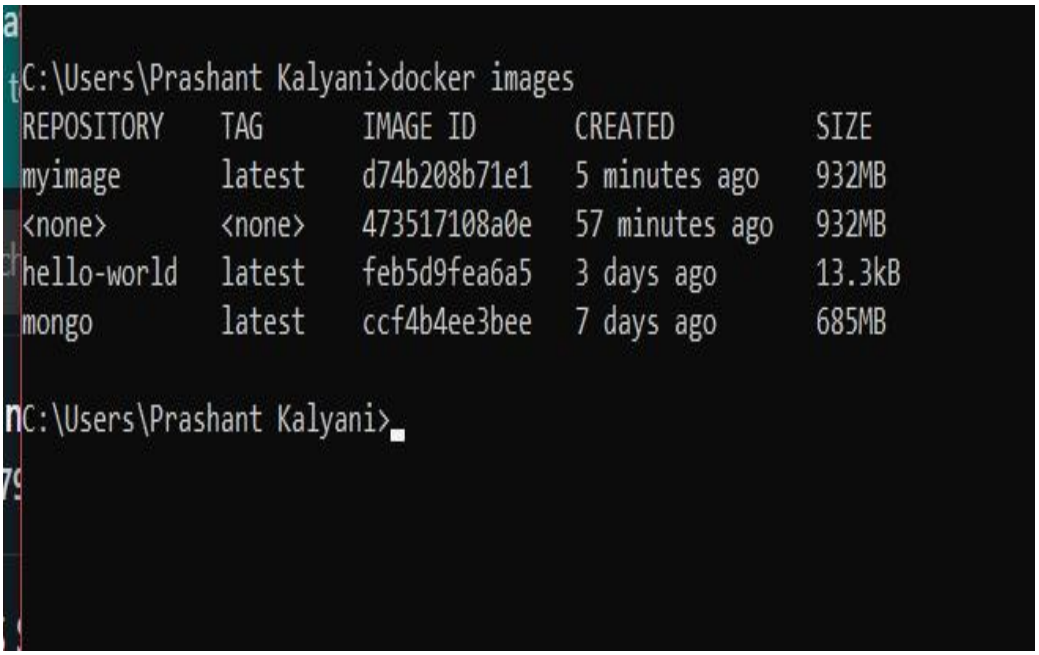
1. Flask
2. Python
3. MongoDB

Project Description:

Our application consists of 2 separate containers, out of which one has server on it, while on the other hand, database is stored in the other one. Both the containers are connected using network bridge for intercommunication. We have also mapped the database volume to our local machine volume so that data remains persistent, even after shutting down the containers.

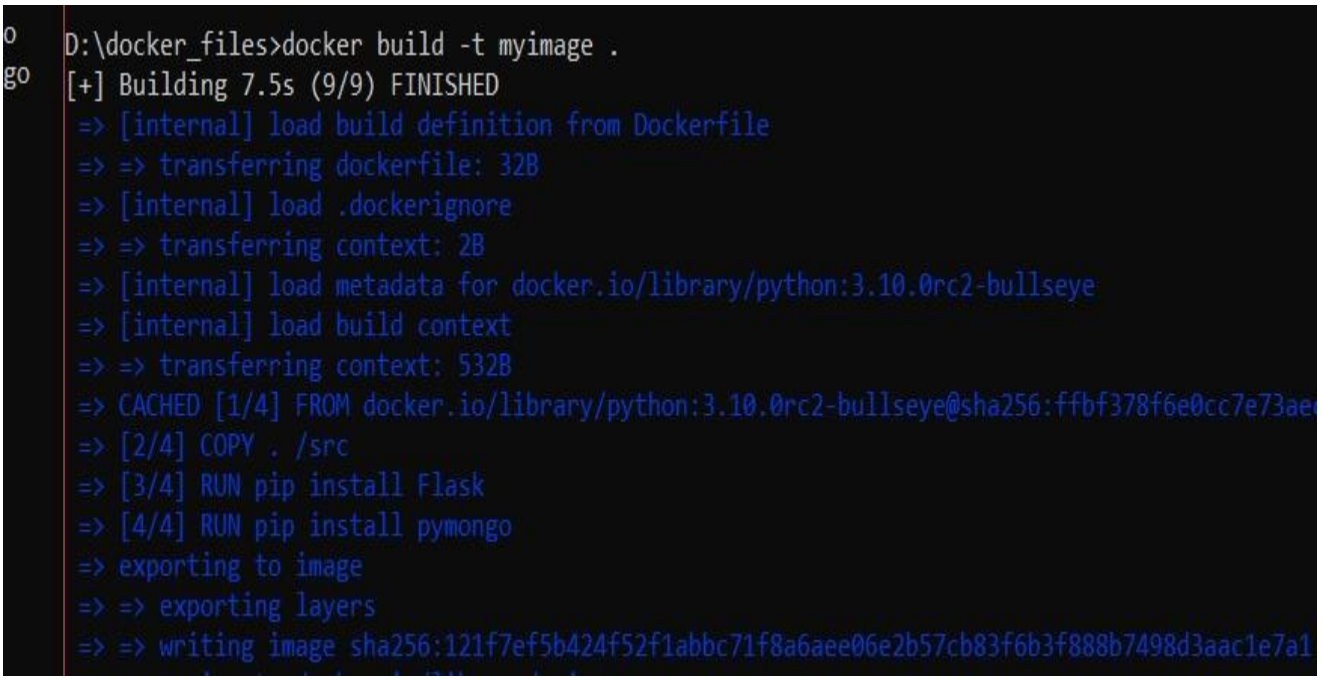
Instructions Executed:

- **docker images**



Above instruction displays the list of all the images in our local machine after instruction execution

- **docker build -t myimage .**



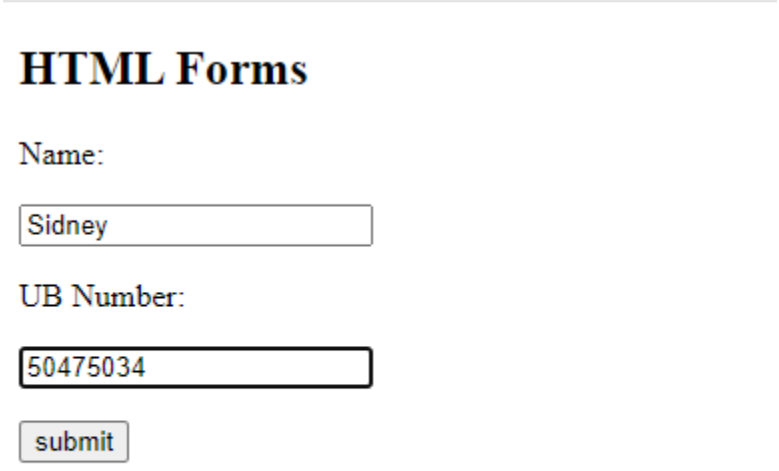
We created a new image with the above instructions.
The following docker file contained installation of Flask and pymongo libraries command and configuration for the image.

- **docker run -it -p 80:5000 --network mode-webapp-network --name my_app --rm --env-file .env myimage**

```
D:\docker_files>docker run -it -p 80:5000 --network mode-webapp-network --name my_app --rm --env-file .env myimage
MongoClient(host=['mongodb1:27017'], document_class=dict, tz_aware=False, connect=True)
 * Serving Flask app 'first_flask_app' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on all addresses.
   WARNING: This is a development server. Do not use it in a production deployment.
 * Running on http://172.18.0.3:5000/ (Press CTRL+C to quit)
```

When we executed the above the instruction, the container was mapped to the localhost, port no 80 and was connected to the network bridge “mode-network-webapp-network”

- **localhost:80**

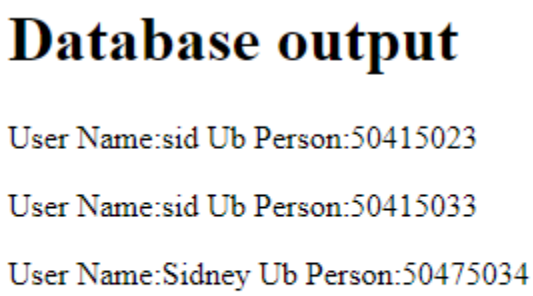


Above image is the flask app running on the localhost at port 80

- **After submitting data**

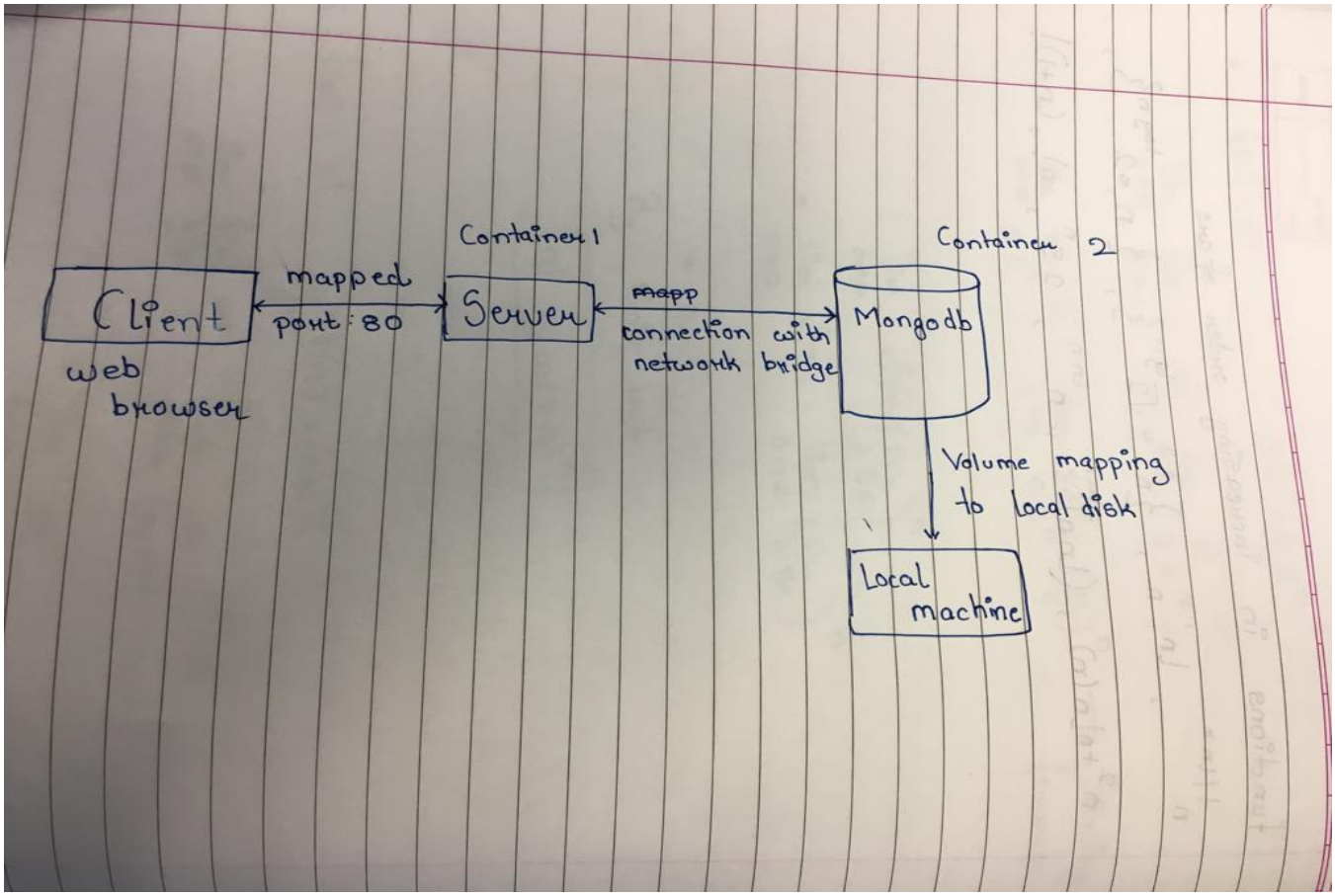
After submitting the data, successful image is popped up, indicating that the data has been stored in the database

Database Output:



In the above image, the last entry was the most recently inserted through our flask application.

Block Diagram:



Desktop Proof: