DEVOPS ASSIGNMENT 2

Name: PEMMAREDDY LIKHITH

ROLL NO: 205224015

BRANCH: M.TECH DATA ANALYTICS

Q1. A) Create a Container with PostgresDB or mySQL database installed

Step 1: We need to pull the docker image using the below command.

```
PS C:\Users\pemma> docker pull postgres
Using default tag: latest
latest: Pulling from library/postgres
6948dc7760c1: Pull complete
0c942aac37b1: Pull complete
6cea4d95608f: Pull complete
97f28320a07a: Pull complete
c1b7de8085d1: Pull complete
97cdd47d9131: Pull complete
8c63b71925de: Pull complete
07db60713289: Pull complete
8a628cdd7ccc: Pull complete
f15c43cffa70: Pull complete
e4847368ad17: Pull complete
3a6f8814136c: Pull complete
2a08aad74366: Pull complete
2817206b0512: Pull complete
Digest: sha256:fe3f571d128e8efadcd8b2fde0e2b73ebab6dbec33f6bfe69d98c682c7d8f7bd
Status: Downloaded newer image for postgres:latest
docker.io/library/postgres:latest
```

Step 2: Run the docker image using the Postgres credentials

```
PS C:\Users\pemma> docker run --name pg-container -e POSTGRES_USER=admin -e POSTGRES_PASSWORD=admin123 -e POSTGRES_DB=te
stdb -p 5432:5432 -d postgres
95b1f9a61dc6fa4f156d418421ca8a2978042b3a1a376385f21cb7dfad02ed7a
```

Step 3: Check whether the docker image is running or not

```
PS C:\Users\pemma> docker run --name pg-container -e POSTGRES_USER=admin -e POSTGRES_PASSWORD=admin123 -e POSTGRES_DB=te
stdb -p 5432:5432 -d postgres
95b1f9a61dc6fa4f156d418421ca8a2978042b3a1a376385f21cb7dfad02ed7a
PS C:\Users\pemma> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED          STATUS          PORTS                    NAMES
95b1f9a61dc6   postgres   "docker-entrypoint.s…"   35 seconds ago   Up 33 seconds   0.0.0.0:5432->5432/tcp   pg-containe
r
```

Step 4: Connect to the database and test by creating a TABLE

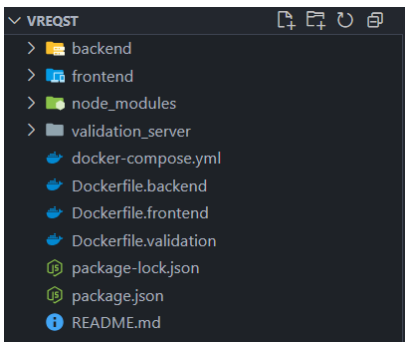
```
PS C:\Users\pemma> docker exec -it pg-container psql -U admin -d testdb
psql (17.4 (Debian 17.4-1.pgdg120+2))
Type "help" for help.

testdb=# CREATE TABLE users (
    id SERIAL PRIMARY KEY,
    username VARCHAR(50) UNIQUE NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    created_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP
);

INSERT INTO users (username, email) VALUES
('john_doe', 'john.doe@example.com'),
('jane_smith', 'jane.smith@example.org');

SELECT * FROM users;
CREATE TABLE
INSERT 0 2
 id |  username  |         email          |          created_at
----+------------+------------------------+-------------------------------
  1 | john_doe   | john.doe@example.com   | 2025-04-20 13:55:47.275177+00
  2 | jane_smith | jane.smith@example.org | 2025-04-20 13:55:47.275177+00
(2 rows)
```

Q1 B) Deploy VReqST – A requirement specification tool in a container.

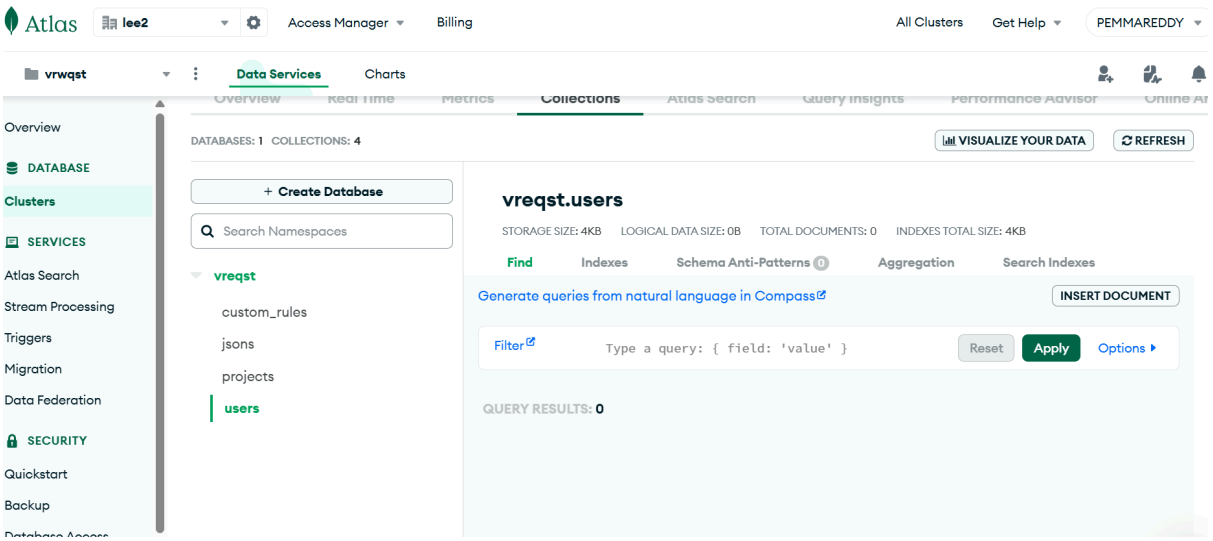
Step 1 : Clone the Repository using the commands

```
git clone https://github.com/<your-repository>/VReqST-2
cd VReqST-2/VReqST-main/VReqST
```

Step 2: Verify the folder structure



Step 3: Create Collections in MongoDB Atlas and use MONGO_URI to link to the application (the link is given in docker-composer file)

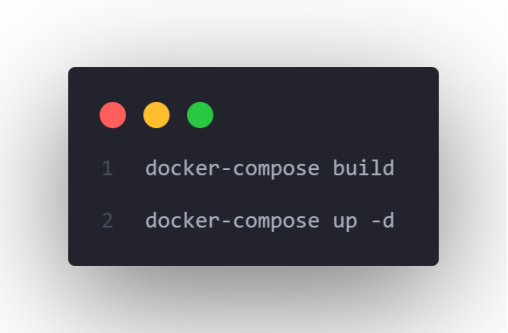


Step 4: Setup Docker

a. **Install Docker & Docker Compose -** Ensure both Docker and Docker Compose are installed and working.
b. **Create docker file `docker-compose.yml` -** Used to run the docker containers
c. **Create subfolders for different services** - Configure `Dockerfile.frontend`, `Dockerfile.backend`, `Dockerfile.validation`

```yaml
version: '3.8'

services:
  backend:
    build:
      context: .
      dockerfile: Dockerfile.backend
    ports:
      - "5002:5002"
    environment:
      - <MONGODB_URI>

  validation:
    build:
      context: .
      dockerfile: Dockerfile.validation
    ports:
      - "5001:5001"
    environment:
      - <MONGODB_URI>

  frontend:
    build:
      context: .
      dockerfile: Dockerfile.frontend
    ports:
      - "3000:3000"
```

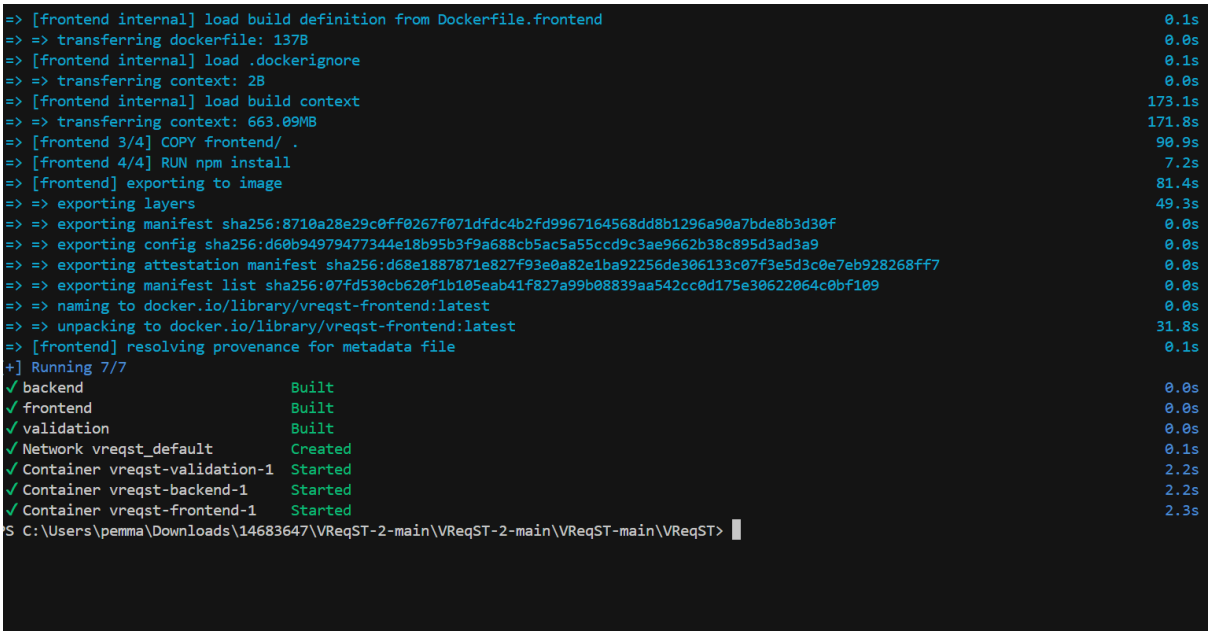
d. **Build and Start Container**



Outputs:

Running the Application

```
PS C:\Users\pemma\Downloads\14683647\VReqST-2-main\VReqST-2-main\VReqST-main\VReqST> docker-compose up --build -d
time="2025-04-20T20:42:41+05:30" level=warning msg="C:\\Users\\pemma\\Downloads\\14683647\\VReqST-2-main\\VReqST-2-main\\VReqST-main\\VReqST
\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
Compose can now delegate builds to bake for better performance.
 To do so, set COMPOSE_BAKE=true.
[+] Building 582.1s (26/26) FINISHED                                                                                    docker:desktop-linux
 => [validation internal] load build definition from Dockerfile.validation                                                          0.2s
 => => transferring dockerfile: 154B                                                                                                0.0s
 => [backend internal] load build definition from Dockerfile.backend                                                                0.2s
 => => transferring dockerfile: 280B                                                                                                0.0s
 => [frontend internal] load metadata for docker.io/library/node:14                                                                16.0s
 => [backend auth] library/node:pull token for registry-1.docker.io                                                                 0.0s
 => [backend internal] load .dockerignore                                                                                           0.1s
 => => transferring context: 2B                                                                                                     0.0s
 => [validation internal] load .dockerignore                                                                                        0.1s
 => => transferring context: 2B                                                                                                     0.0s
 => [frontend 1/4] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa          82.6s
 => => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa                    0.1s
 => => sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbc9b83d8 4.19kB / 4.19kB                                      0.4s
 => => sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 450B / 450B                                          0.6s
 => => sha256:0c8cc2f24a4dcb64e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 2.29MB / 2.29MB                                      1.7s
 => => sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 35.24MB / 35.24MB                                   15.4s
 => => sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fdfba8e23d1b1569 191.85MB / 191.85MB                                 42.4s
 => => sha256:3d2201bd995cccf12851a50820de03d34a17011dcbb9ac9fdf3a50c952cbb131 10.00MB / 10.00MB                                    5.7s
 => => sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2 51.88MB / 51.88MB                                   19.1s
 => => sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5 7.86MB / 7.86MB                                      4.0s
 => => sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408 50.45MB / 50.45MB                                   15.8s
 => => extracting sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408                                           17.5s
 => => extracting sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5                                           0.8s
 => => extracting sha256:3d2201bd995cccf12851a50820de03d34a17011dcbb9ac9fdf3a50c952cbb131                                           0.4s
 => => extracting sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2                                           5.4s
```



Step 5: Access the Application

Open your browser and go to: **http://localhost:3000**