

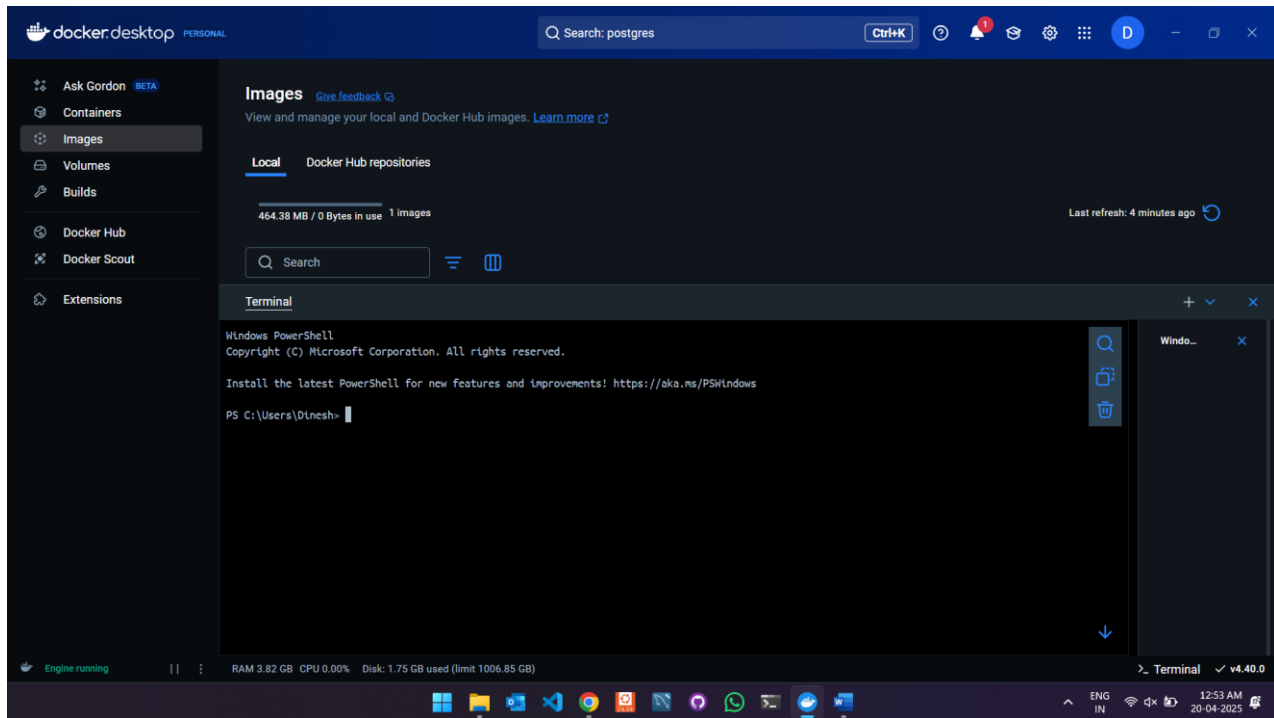
DEVOPS ASSIGNMENT 2

Name: Thumma Dinesh

ROLL NO: 205224025

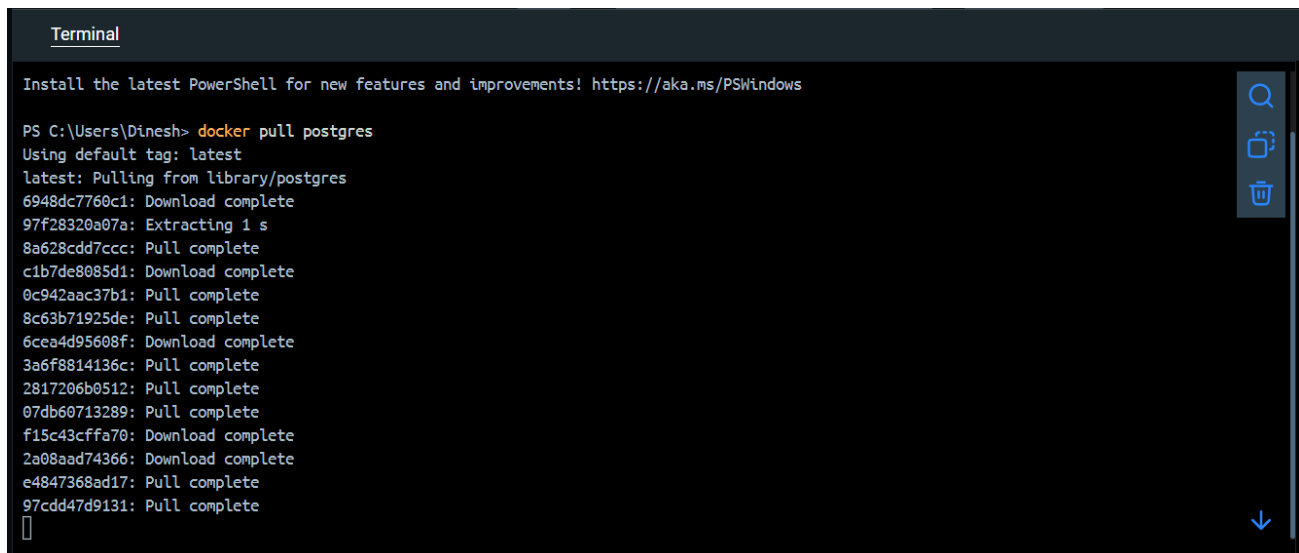
BRANCH: M.Tech Data Analytics

- First, I have installed Docker Desktop Version 4.40.0



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

- First, we need to pull the docker image.



- Now run the docker image using the Postgres credentials

```
PS C:\Users\Dinesh> docker run --name pg-container -e POSTGRES_USER=admin -e POSTGRES_PASSWORD=admin123 -e POSTGRES_DB=testdb -p 5432:5432 -d postgres
8a11cccf338afc009f0e667c152429fc636738722c05fd33d2623fe4643b432
PS C:\Users\Dinesh>
```

RAM 3.51 GB CPU 1.16% Disk: 1.71 GB used (limit 1006.85 GB)

- Now check once if the image is running properly or not

```
PS C:\Users\Dinesh> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
8a11cccf338    postgres "docker-entrypoint.s..." 40 seconds ago Up 40 seconds 0.0.0.0:5432->5432/tcp             pg-container
PS C:\Users\Dinesh>
```

- Now, connect to the database and test by creating a simple TABLE

```
PS C:\Users\Dinesh> 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 students (
        id SERIAL PRIMARY KEY,
        name VARCHAR(100),
        age INT,
        email VARCHAR(100)
    );
```

Terminal

Column	Type	Table "public.students"	Collation	Nullable	Default
id	integer			not null	nextval('students_id_seq'::regclass)
name	character varying(100)				
age	integer				
email	character varying(100)				

Indexes:

"students_pkey" PRIMARY KEY, btree (id)

Q1.

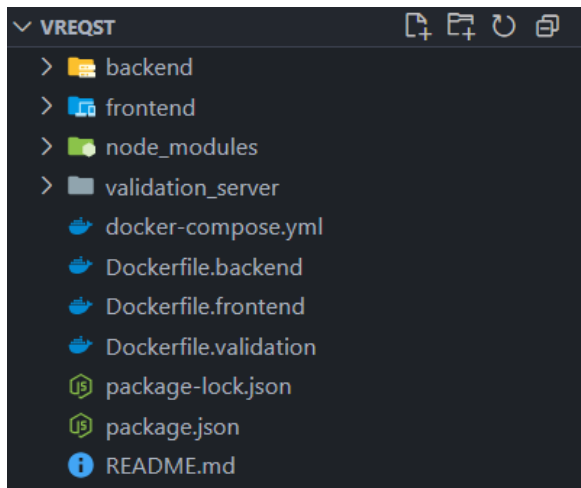
- B) Deploy VReqST – A requirement specification tool in a container.

- Clone the Repository using the commands

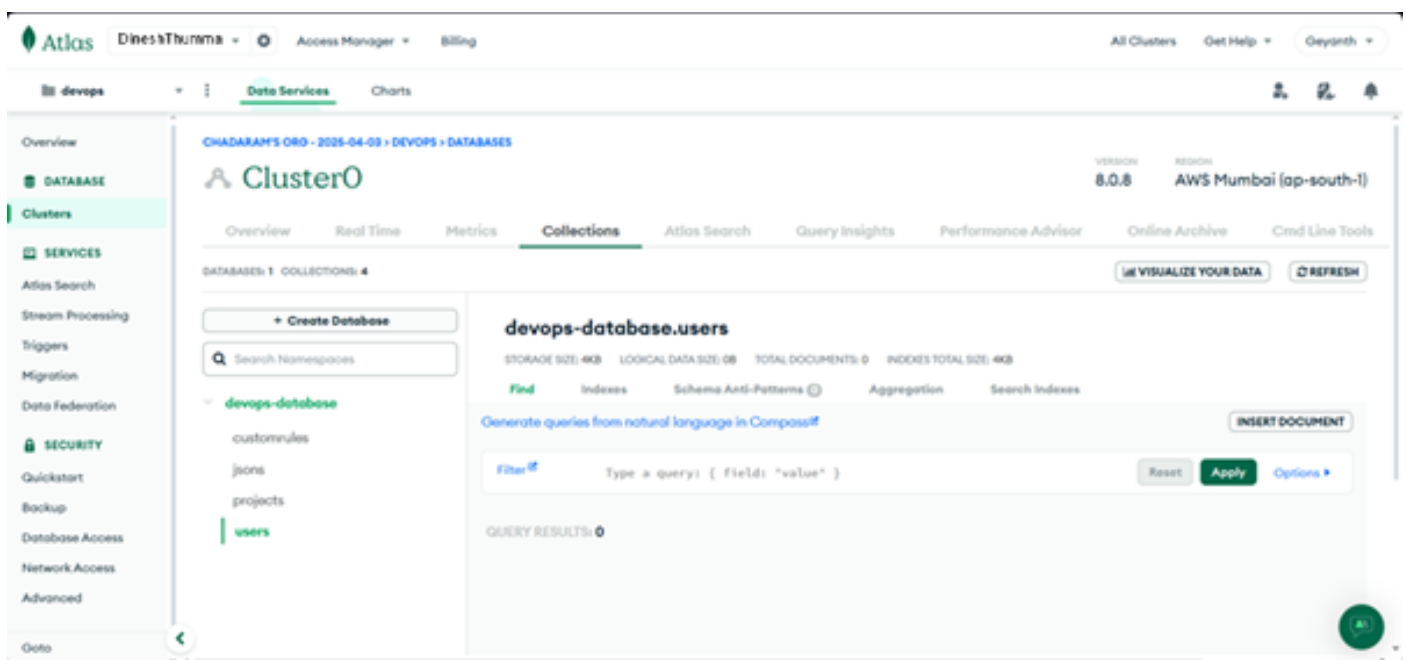
```
PS C:\Users\Dinesh> git clone https://github.com/<repository>/VReqST-2
Cloning into 'VReqST-2'...
```

```
PS C:\Users\Dinesh> cd VReqST-2/VReqST-main/VReqST
```

- Verify the folder structure



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



- Setup Docker

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

```
1 version: '3.8'
2
3 services:
4   backend:
5     build:
6       context: .
7       dockerfile: Dockerfile.backend
8     ports:
9       - "5002:5002"
10    environment:
11      - <MONGODB_URI>
12
13  validation:
14    build:
15      context: .
16      dockerfile: Dockerfile.validation
17    ports:
18      - "5001:5001"
19    environment:
20      - <MONGODB_URI>
21
22  frontend:
23    build:
24      context: .
25      dockerfile: Dockerfile.frontend
26    ports:
27      - "3000:3000"
28    depends_on:
29      - backend
30      - validation
```

d. Build and Start Container

```
1 docker-compose build
2 docker-compose up -d
```

Now the final Outputs are :

```
D:\M.Tech\2nd Sem\Devops\Assignments\devops_tool\VReqST-2\VReqST-main\VReqST>docker-compose up -d
time="2025-04-19T21:23:15+05:30" level=warning msg="D:\M.Tech\2nd Sem\Devops\Assignments\devops_tool\VReqST-2\VReqST-main\VReqST\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 4/4
  ✓ Network vreqst_default          Created
  ✓ Container vreqst-backend-1      Started
  ✓ Container vreqst-validation-1    Started
  ✓ Container vreqst-frontend-1     Started
```

➤ The Terminal Output while Running the Application is:

```
D:\M.Tech\2nd Sem\Devops\Assignments\devops_tool\VReqST-2\VReqST-main\VReqST>docker-compose up --build -d
time="2025-04-19T21:30:42+05:30" level=warning msg="D:\M.Tech\2nd Sem\Devops\Assignments\devops_tool\VReqST-2\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 226.9s (26/26) FINISHED                                docker:desktop-linux
=> [validation internal] load build definition from Dockerfile.validation 0.0s
=> => transferring dockerfile: 154B 0.0s
=> [backend internal] load build definition from Dockerfile.backend 0.1s
=> => transferring dockerfile: 280B 0.0s
=> [frontend internal] load metadata for docker.io/library/node:14 2.2s
=> [backend internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [validation internal] load .dockerignore 0.1s
=> => transferring context: 2B 0.0s
=> [backend internal] load build context 17.3s
=> => transferring context: 32.59MB 17.1s
=> [frontend 1/5] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d 0.2s
=> => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461 0.0s
=> [validation internal] load build context 1.1s
=> => transferring context: 67.22kB 0.9s
=> CACHED [frontend 2/5] WORKDIR /app 0.0s
=> CACHED [validation 3/4] COPY validation_server/ . 0.0s
=> CACHED [validation 4/4] RUN npm install 0.0s
=> [validation] exporting to image 0.6s
=> => exporting layers 0.0s
=> => exporting manifest sha256:054f0155f3a3378988bbb8a442d1b686e32e8f2052deebecee896cec402ce587 0.0s
```

```
=> [backend] resolving provenance for metadata file 0.1s
=> [frontend internal] load build definition from Dockerfile.frontend 0.0s
=> => transferring dockerfile: 177B 0.0s
=> [frontend internal] load .dockerignore 0.1s
=> => transferring context: 2B 0.0s
=> [frontend internal] load build context 201.8s
=> => transferring context: 664.95MB 200.6s
=> CACHED [frontend 3/5] COPY frontend/ . 0.0s
=> CACHED [frontend 4/5] RUN npm install 0.0s
=> CACHED [frontend 5/5] RUN ls -la /app/client # Add this line 0.0s
=> [frontend] exporting to image 0.4s
=> => exporting layers 0.0s
=> => exporting manifest sha256:25d3061c140fb874e71d76009238eeec972e7437de7cce4db936ad66f562193 0.0s
=> => exporting config sha256:00413a90cc6393cfac6ab26a6665f5d79e1368a46754e1a20d739e9019c950a3 0.0s
=> => exporting attestation manifest sha256:ba87c8f2edb3299ed013c51304019001bc46aa6302e5e2d3f20e1acc866eefc5 0.1s
=> => exporting manifest list sha256:3100cf308e74f287d2b670743cfa1993cd21145778fe7368091658cda829af30 0.1s
=> => naming to docker.io/library/vreqst-frontend:latest 0.0s
=> => unpacking to docker.io/library/vreqst-frontend:latest 0.0s
=> [frontend] resolving provenance for metadata file 0.0s
[+] Running 7/7
  ✓ backend          Built
  ✓ frontend         Built
  ✓ validation       Built
  ✓ Network vreqst_default Created
  ✓ Container vreqst-backend-1 Started
  ✓ Container vreqst-validation-1 Started
  ✓ Container vreqst-frontend-1 Started
```

➤ Finally, we can access the application using a browser and the link:

<http://localhost:3000>