DEVOPS- ASSIGNMENT- 2

Name: MAYANK

ROLL NO: 205224010

BRANCH: M.TECH DATA ANALYTICS

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

Objective:

To create a Docker container with PostgreSQL database installed, perform basic database operations, and document the process.

Environment Details:

- Docker Version: 3.8
- Operating System: Windows 10
- PostgreSQL Version: 13

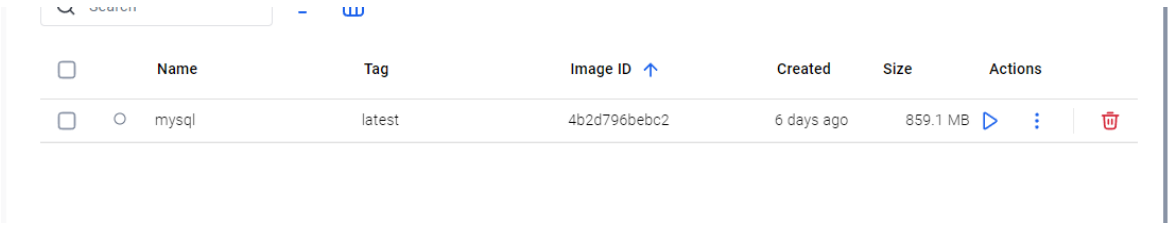
Docker Compose Configuration:

The following **docker-compose.yml** file was created to set up the MySQL container:

Commands Executed:

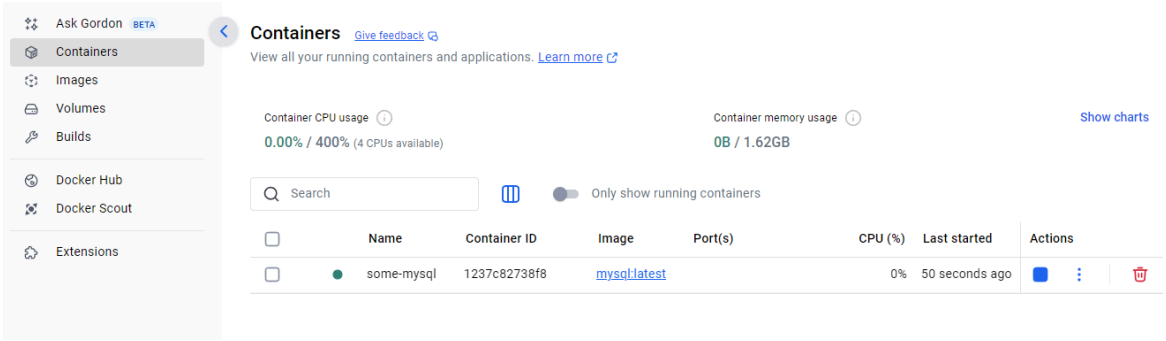
- Starting the container: **docker-compose up -d**

```
C:\Users\mayan>docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
cea172a6e83b: Pull complete
fa811e9a869e: Pull complete
47a2982daa21: Pull complete
634d7076afe3: Pull complete
aa8a3958f09f: Pull complete
84e4e5ea3754: Pull complete
2275c0ff11a0: Pull complete
2792ea2d4e0e: Pull complete
f488b2cd8494: Pull complete
9451290759df: Pull complete
Digest: sha256:7839322bd6c3174a699586c3ea36314c59b61b4ce01b4146951818b94aef5fd7
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
```



- To check running containers: **docker ps**

```
C:\Users\mayan>docker run --name some-mysql -e MYSQL_ROOT_PASSWORD=my-secret-pw -d mysql:latest
1237c82738f875b97cc1e641407b1804cbdff08fa6c58af30128804fc16af4a6
```



MySQL CLI:

- Table creation

```
C:\Users\mayan>docker exec -it some-mysql mysql -u root -p

Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 9.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE name;
Query OK, 1 row affected (0.097 sec)

mysql> USE name;
Database changed
mysql> CREATE TABLE test_table (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.965 sec)

mysql> INSERT INTO test_table (name) VALUES ('MAYANK_205224010');
Query OK, 1 row affected (0.992 sec)

mysql> SELECT * FROM test_table;
+----+------------------+
| id | name             |
+----+------------------+
|  1 | MAYANK_205224010 |
+----+------------------+
1 row in set (0.064 sec)
```

- Stopping the container: **docker-compose down**

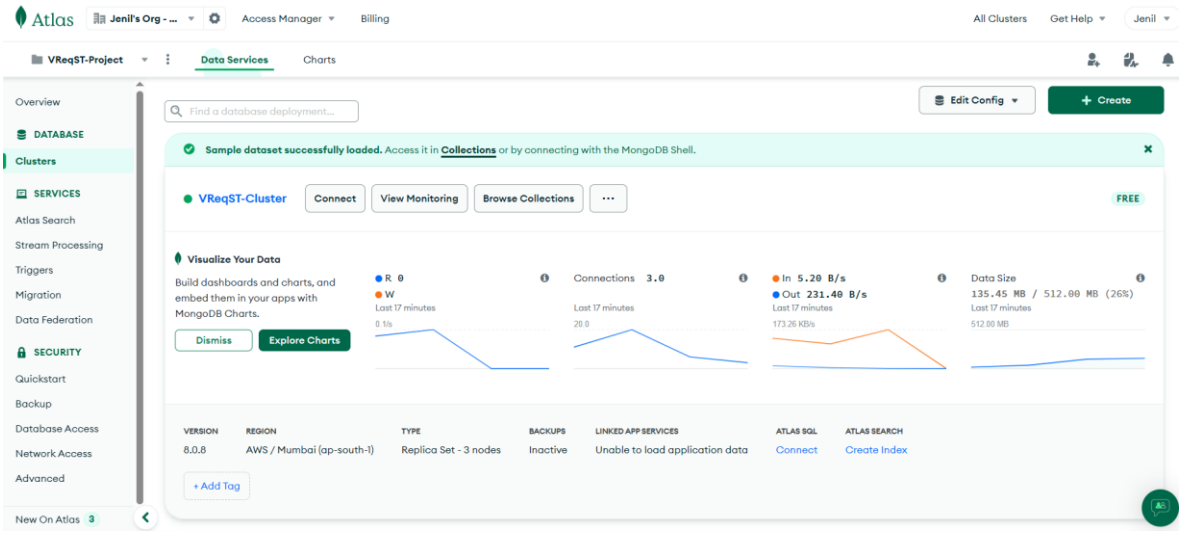
Q1 (B): Deploy VReqST – A requirement specification tool in a container.

Objective:

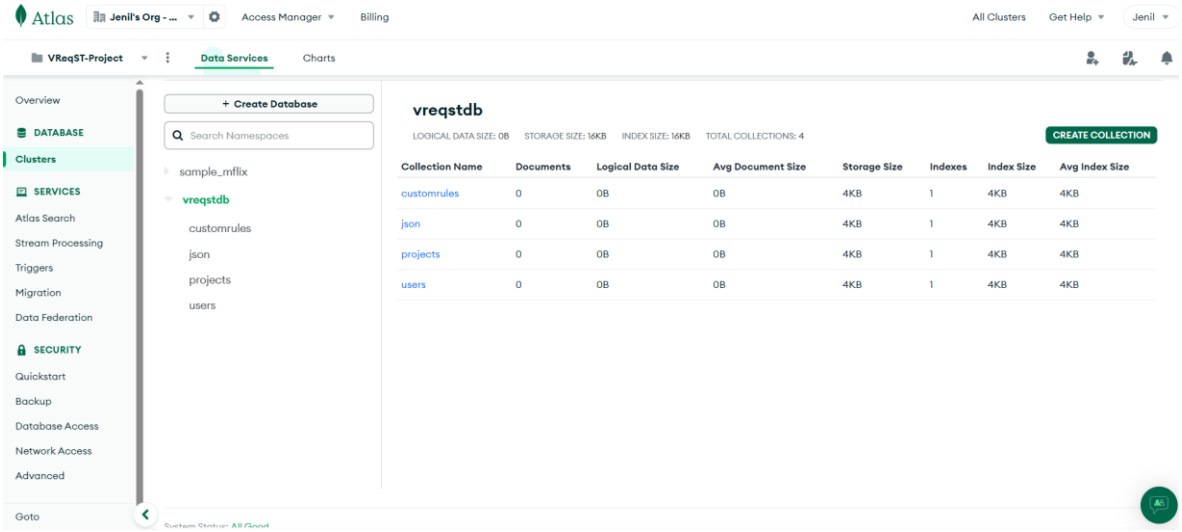
To deploy the VReqST (Virtual Reality Requirement Specification Tool) application using Docker, containerizing both the application and the associated MongoDB database service, ensuring proper database configuration and connectivity.

Setting up MongoDB Database:

- Create a new Project within MongoDB Atlas.
- Inside the project, create a Database Cluster (free-tier is sufficient for this assignment).



- Once the cluster is created, define four collections within a new database using the following names:
  - customrules
  - jsons
  - projects
  - users



- Update the application's server code to replace any local MongoDB connection string (e.g. mongodb://localhost:27017/vreqst) with the above cloud-hosted MongoDB Atlas connection string.

Typically, this connection string is found in either:

- backend/server.js
- backend/app.js
- or a configuration file such as backend/config.js or backend/config/db.js

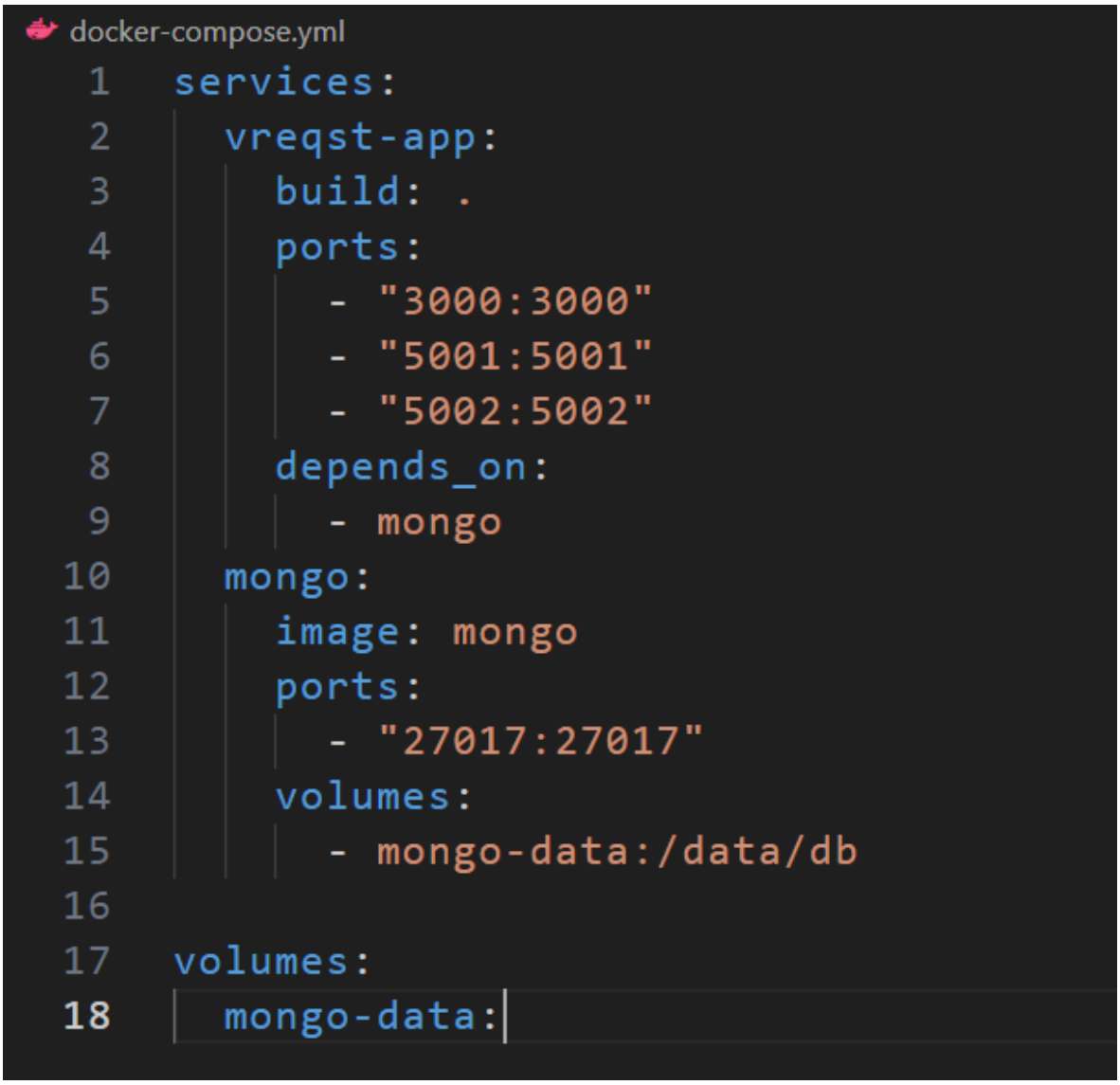
Writing the DockerFile and docker-compose.yml File:

Create a **Dockerfile** file in the project root directory with the following code:

Dockerfile

```
FROM node:14

WORKDIR /app

COPY . .

WORKDIR /app/backend
RUN npm install

WORKDIR /app/validation_server
RUN npm install

WORKDIR /app/frontend
RUN npm install
RUN npm run client-install

EXPOSE 3000 5001 5002

CMD ["bash", "-c", "cd /app/backend && npx nodemon index.js & cd /app/validation_server && npx nodemon index.js & cd /app/frontend && npm run dev"]
```

Create a **docker-compose.yml** file in the project root directory with the following configuration:



Explanation:

- The vreqst-app service builds the VReqST application from the Dockerfile in the current directory and maps necessary ports.
- The mongo service pulls the official MongoDB image and binds port 27017.
- mongo-data volume ensures data persistence for MongoDB.

Building and Running the Docker Containers:

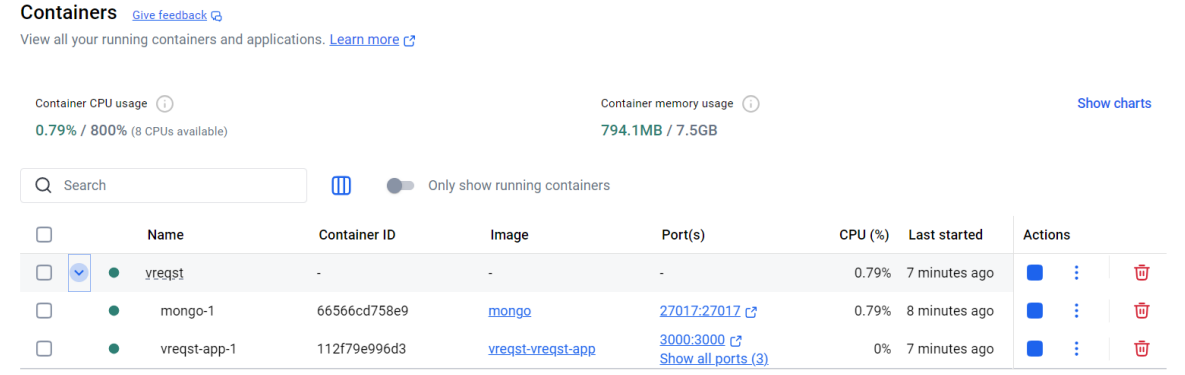
Building and running: **docker-compose up –build**
This command will:

- Build the Docker images.
- Start both vreqst-app and mongo services.
- Map ports as defined in docker-compose.yml

```
D:\Project\DevOps\Final Assignment\VReqST>docker-compose up --build
time="2025-04-20T18:35:41+05:30" level=warning msg="D:\\Project\\DevOps\\Final Assignment\\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 360.4s (28/28) FINISHED                                                                                                                                                                docker:desktop-linux
 => [backend internal] load build definition from Dockerfile                                                                                                                                                        0.1s
 => => transferring dockerfile: 161B                                                                                                                                                                                0.0s
 => [validation_server internal] load build definition from Dockerfile                                                                                                                                              0.1s
 => => transferring dockerfile: 167B                                                                                                                                                                                0.0s
 => [frontend internal] load build definition from Dockerfile                                                                                                                                                       0.1s
 => => transferring dockerfile: 163B                                                                                                                                                                                0.0s
 => [backend internal] load metadata for docker.io/library/node:14                                                                                                                                                  4.3s
 => [validation_server auth] library/node:pull token for registry-1.docker.io                                                                                                                                       0.0s
 => [validation_server internal] load .dockerignore                                                                                                                                                                 0.2s
 => => transferring context: 2B                                                                                                                                                                                     0.0s
 => [frontend internal] load .dockerignore                                                                                                                                                                          0.3s
 => => transferring context: 2B                                                                                                                                                                                     0.0s
 => [backend internal] load .dockerignore                                                                                                                                                                           0.2s
 => => transferring context: 2B                                                                                                                                                                                     0.0s
 => [validation_server 1/5] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa                                                                                 50.9s
 => => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa                                                                                                    0.2s
 => => sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 35.24MB / 35.24MB                                                                                                                    9.4s
 => => sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 450B / 450B                                                                                                                          0.9s
 => => sha256:0c8cc2f24a4dcb64e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 2.29MB / 2.29MB                                                                                                                      1.5s
 => => sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbc9b83d8 4.19kB / 4.19kB                                                                                                                      1.0s
 => => sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fdfba8e23d1b1569 191.85MB / 191.85MB                                                                                                                 28.3s
 => => sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2 51.38MB / 51.88MB                                                                                                                   353.8s
 => => sha256:3d2201bd995cccf12851a50820de03d34a17011dcbb9ac9fdf3a50c952cbb131 10.00MB / 10.00MB                                                                                                                    3.5s
 => => sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5 7.86MB / 7.86MB                                                                                                                      2.5s
 => => sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408 50.45MB / 50.45MB                                                                                                                    10.3s
 => => extracting sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408                                                                                                                            6.4s
 => => extracting sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5                                                                                                                            1.1s
 => => extracting sha256:3d2201bd995cccf12851a50820de03d34a17011dcbb9ac9fdf3a50c952cbb131                                                                                                                            0.7s
 => => extracting sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2                                                                                                                            5.4s
 => => extracting sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fdfba8e23d1b1569                                                                                                                           12.7s
 => => extracting sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbc9b83d8                                                                                                                            0.2s
 => => extracting sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590                                                                                                                            5.3s
 => => extracting sha256:0c8cc2f24a4dcb64e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3                                                                                                                            0.2s
 => => extracting sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb                                                                                                                            0.1s
 => [validation_server internal] load build context                                                                                                                                                                 4.1s
```

```
 => => transferring context: 31.69MB                                                                                                                                                                               21.4s
 => [frontend internal] load build context                                                                                                                                                                        152.9s
 => => transferring context: 662.32MB                                                                                                                                                                             152.5s
 => [frontend 2/5] WORKDIR /app                                                                                                                                                                                     1.1s
 => [validation_server 3/5] COPY package*.json ./                                                                                                                                                                   0.2s
 => [backend 3/5] COPY package*.json ./                                                                                                                                                                             0.5s
 => [validation_server 4/5] RUN npm install                                                                                                                                                                        50.7s
 => [backend 4/5] RUN npm install                                                                                                                                                                                  84.9s
 => [validation_server 5/5] COPY . .                                                                                                                                                                                0.5s
 => [validation_server] exporting to image                                                                                                                                                                          2.4s
 => => exporting layers                                                                                                                                                                                             0.8s
 => => exporting manifest sha256:37577a70433431d7aed10a83f24d6d4744860d985823deea4771631318a1128b                                                                                                                   0.0s
 => => exporting config sha256:bb97a2a88771fac7f68c4dcccd47a76c10221775960bd644b51ee2273967be56                                                                                                                     0.0s
 => => exporting attestation manifest sha256:0e0cb3b773a2d8323920b6eba79ebad94e34d0d9fb1ecef4982095b049127f71                                                                                                       0.1s
 => => exporting manifest list sha256:142a50201da08a76b6eaad9e765f1ac9e8a6f780aa30280daf7660fd2c6d1a5d                                                                                                              0.0s
 => => naming to docker.io/library/vreqst-validation_server:latest                                                                                                                                                  0.0s
 => => unpacking to docker.io/library/vreqst-validation_server:latest                                                                                                                                               1.2s
 => [validation_server] resolving provenance for metadata file                                                                                                                                                      0.1s
 => [backend 5/5] COPY . .                                                                                                                                                                                          8.0s
 => [backend] exporting to image                                                                                                                                                                                   17.3s
 => => exporting layers                                                                                                                                                                                             7.0s
 => => exporting manifest sha256:7533ba38e8b9ed396f982b10180223592b93e9991316b0d35840f00a65766da2                                                                                                                   0.0s
 => => exporting config sha256:1be18f4003472becd5fb50e0913cc0d82e2703ea93a8555af1f9f698e528a779                                                                                                                     0.0s
 => => exporting attestation manifest sha256:0b380f15fa958b28ec7c6a02262caff55f1c1cff832e5353d690a8e5ccecf482                                                                                                       0.1s
 => => exporting manifest list sha256:adf4c2f1a0afe404bcf683980a7420a52d1e922de9585b50da8712c612e1fc2b5                                                                                                              0.0s
 => => naming to docker.io/library/vreqst-backend:latest                                                                                                                                                            0.0s
 => => unpacking to docker.io/library/vreqst-backend:latest                                                                                                                                                        10.1s
 => [frontend 3/5] COPY package*.json ./                                                                                                                                                                            1.0s
 => [frontend 4/5] RUN npm install                                                                                                                                                                                 55.5s
 => [backend] resolving provenance for metadata file                                                                                                                                                                0.0s
 => [frontend 5/5] COPY . .                                                                                                                                                                                        20.0s
 => [frontend] exporting to image                                                                                                                                                                                 125.5s
 => => exporting layers                                                                                                                                                                                            64.7s
 => => exporting manifest sha256:8ce45161458f5b800ef11f351369b295bef7a1ab3024b439b445753f685ab974                                                                                                                    0.0s
 => => exporting config sha256:9f6197f47626fc2d3a9314dcb3c443660a50cd4732e86a2f14cd4ac004ff035c                                                                                                                      0.0s
 => => exporting attestation manifest sha256:aa1d92b9dcb31c516332c0c556eeaa6c276dd5459ab9181ca187b2e430297154                                                                                                       0.1s
 => => exporting manifest list sha256:3fa04b60c7342710398abe42d1f80f4d1caaa6a825f6f366e59683f3b8adbfee                                                                                                               0.0s
 => => naming to docker.io/library/vreqst-frontend:latest                                                                                                                                                           0.0s
 => => unpacking to docker.io/library/vreqst-frontend:latest                                                                                                                                                       60.5s
 => [frontend] resolving provenance for metadata file                                                                                                                                                               0.0s
[+] Running 3/3
 ⠿ backend          Built                                                                                                                                                                                           0.0s
[+] Running 6/6     Built                                                                                                                                                                                           0.0s
 ⠿ backend                          Built                                                                                                                                                                      0.0s
 ⠿ frontend                         Built                                                                                                                                                                      0.0s
 ⠿ validation_server                Built                                                                                                                                                                      0.0s
 ⠿ Container vreqst-backend-1       Created                                                                                                                                                                    1.9s
 ⠿ Container vreqst-validation_server-1  Created                                                                                                                                                               1.8s
 ⠿ Container vreqst-frontend-1      Created                                                                                                                                                                    1.9s
Attaching to backend-1, frontend-1, validation_server-1
```

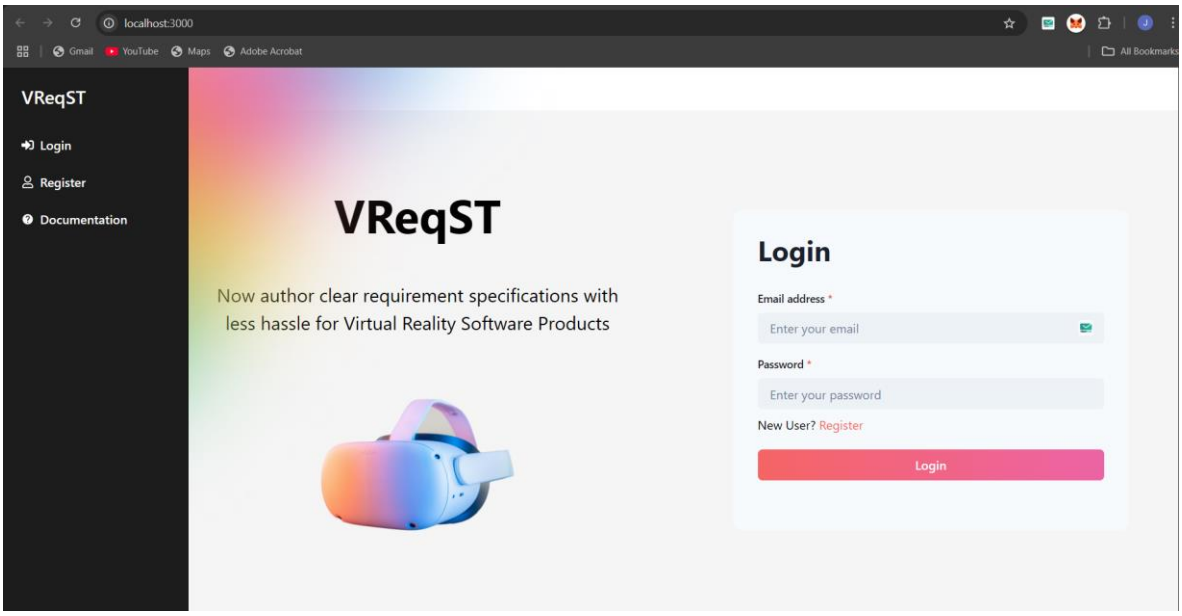
Running Container



Verifying Application and Database

Application Access:
Once the containers are running, access the application through:

- http://localhost:3000



MongoDB Access assumption :
If connecting to a locally running Mongo container, MongoDB would be accessible on mongodb://localhost:27017.
However, as per our configuration, we are using **MongoDB Atlas**, so no local connection is necessary after linking the Atlas connection string in the server code.

Outcome:

At the end of this task:

- The VReqST application runs inside a Docker container.
- It is connected to a cloud-hosted MongoDB Atlas database instance.
- Application services are accessible via defined ports.