Emmalu Joseph

s224791713

8.3HD

https://github.com/emmalujosephwork/SIT725.git

Containerization and Dockerization

Containerization has significantly transformed the way software applications are developed, tested, and deployed. Docker, as one of the most widely used containerization platforms, provides a solution to many challenges faced in software development, particularly in ensuring consistency across different environments.

One of the key advantages of Docker is **portability**. By packaging an application along with all its dependencies into a container, it ensures that the application runs uniformly across different systems, eliminating compatibility issues. This solves the common problem where software behaves differently on development, testing, and production environments.

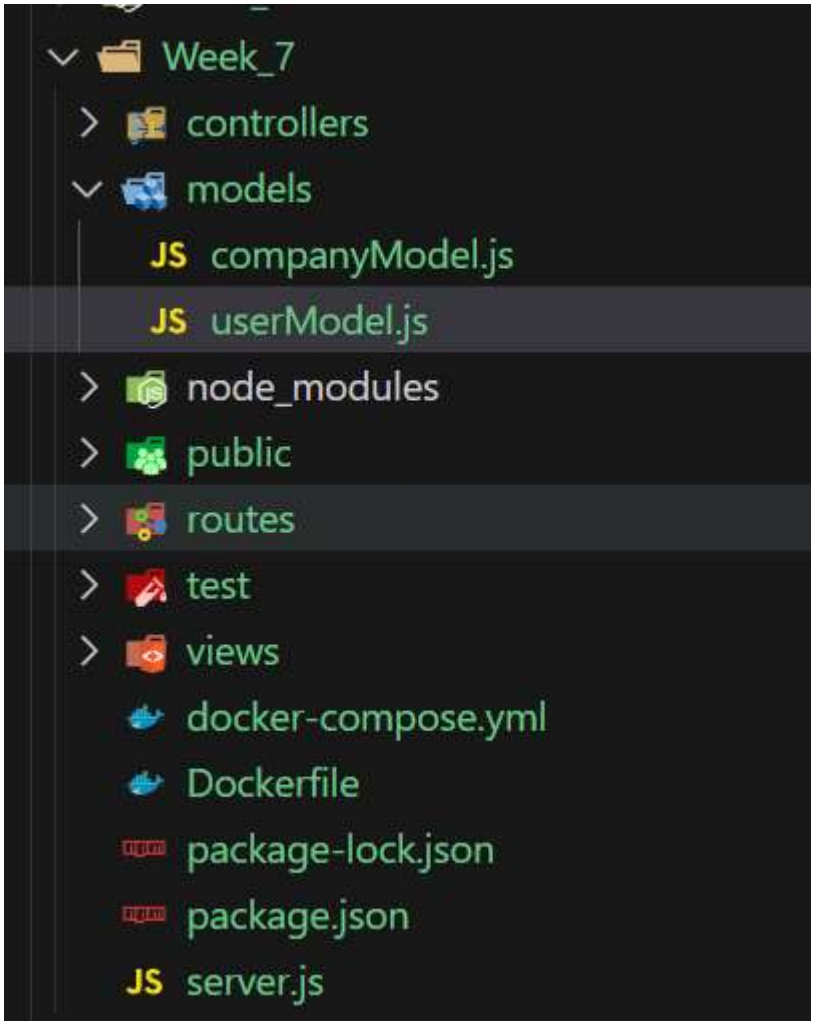
Another important aspect of Dockerization is **efficiency in resource utilization**. Unlike traditional virtual machines, containers share the same operating system kernel, making them lightweight and faster to deploy. This helps in optimizing system resources while improving application performance.

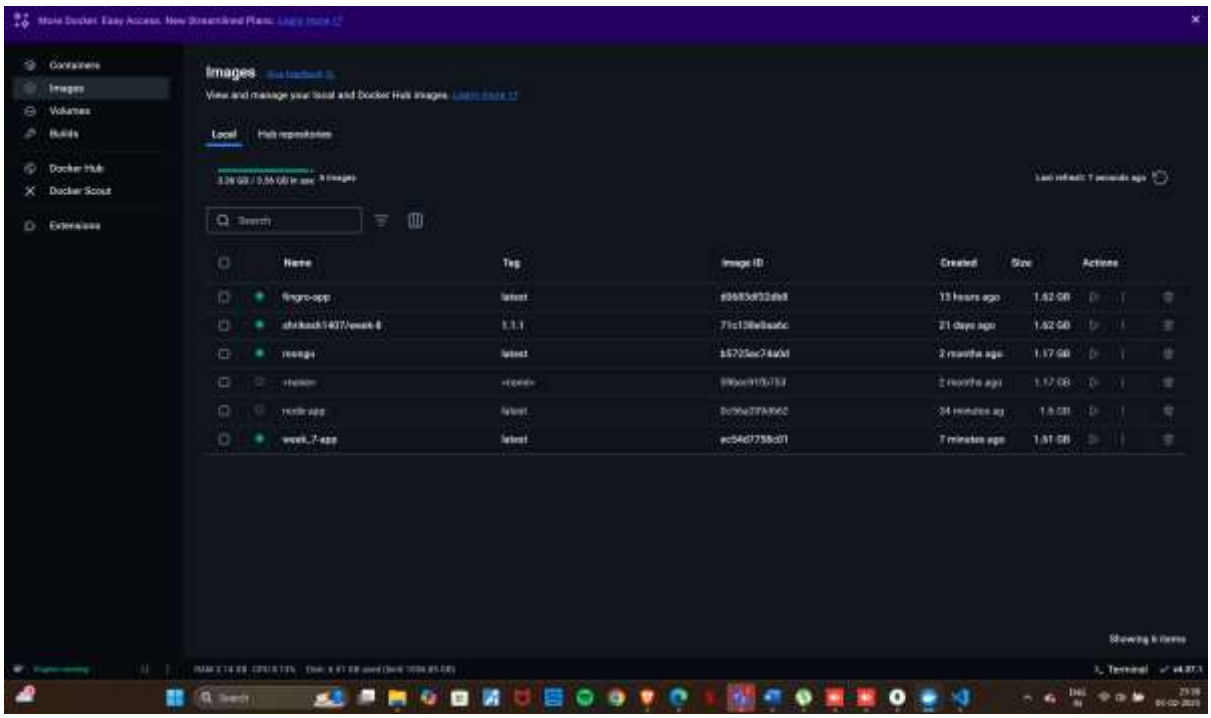
Additionally, Docker enables **scalability and flexibility**, making it easier to manage applications as microservices. With tools like **Docker Compose** and **Kubernetes**, developers can efficiently deploy, scale, and manage multiple containers, which is crucial for handling large-scale applications.

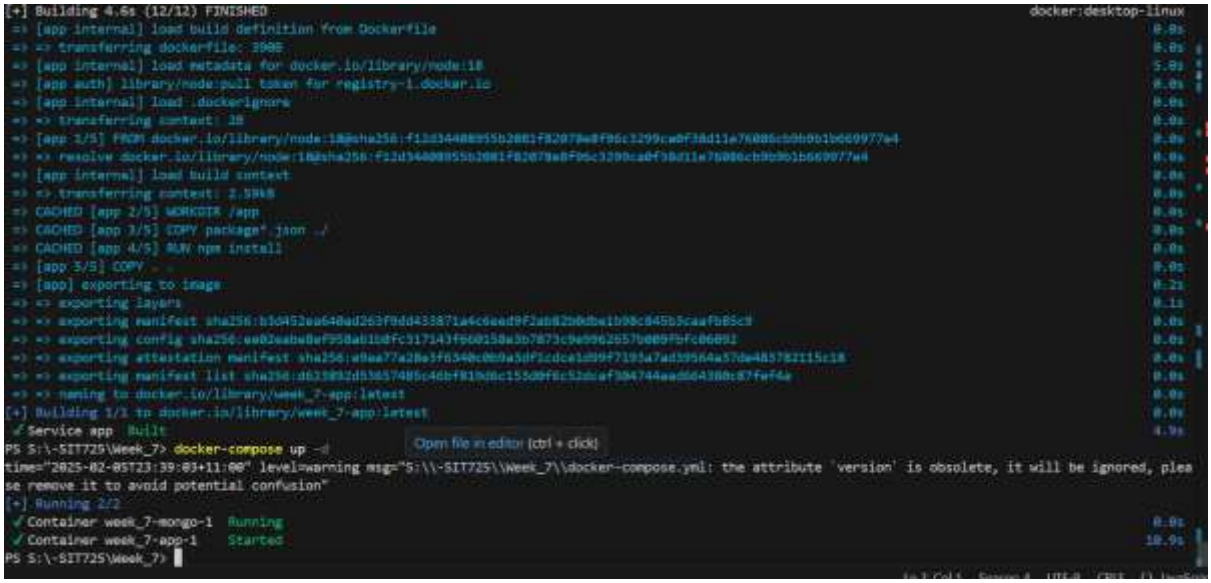
However, despite its many benefits, containerization also presents challenges. Security risks, such as vulnerabilities in container images, require careful management. Additionally, learning how to properly configure and orchestrate containers can be complex, especially for beginners.

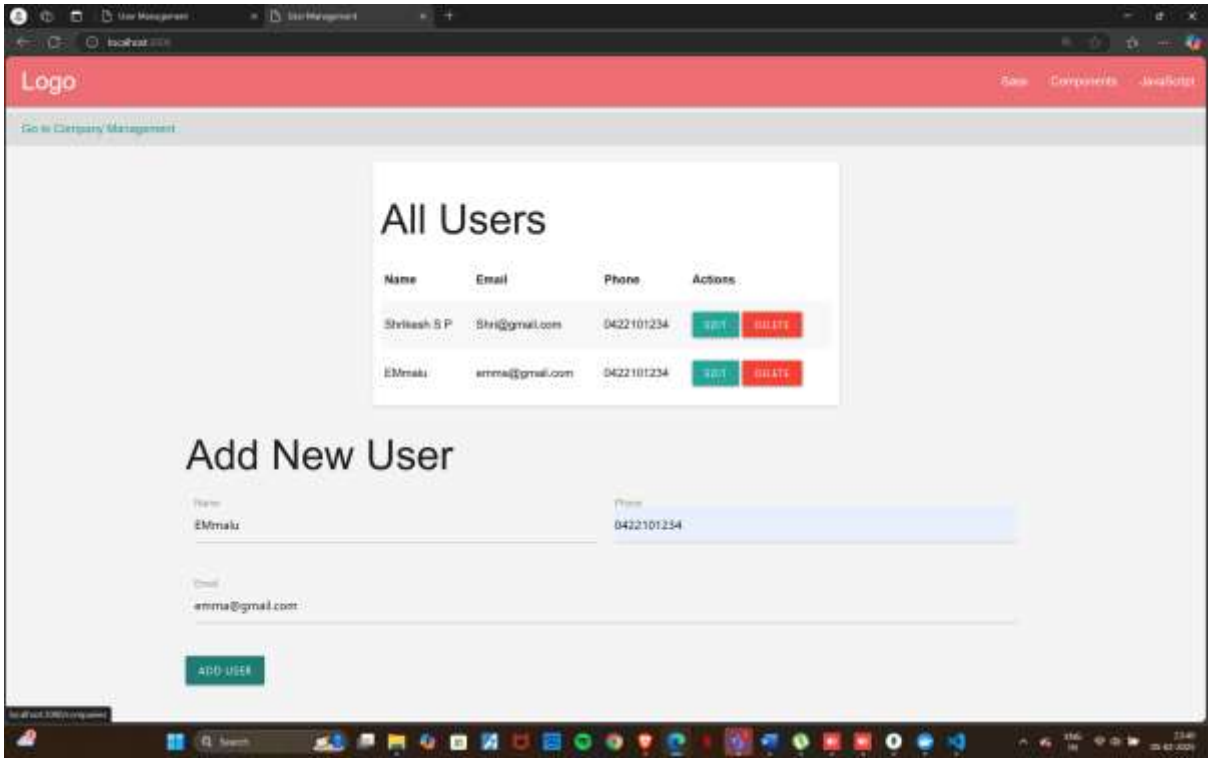
In conclusion, Dockerization is a powerful approach to modern software development, offering portability, efficiency, and scalability. While it comes with a learning curve, its advantages outweigh the challenges, making it an essential tool for developers and organizations aiming for smooth and consistent application deployment.

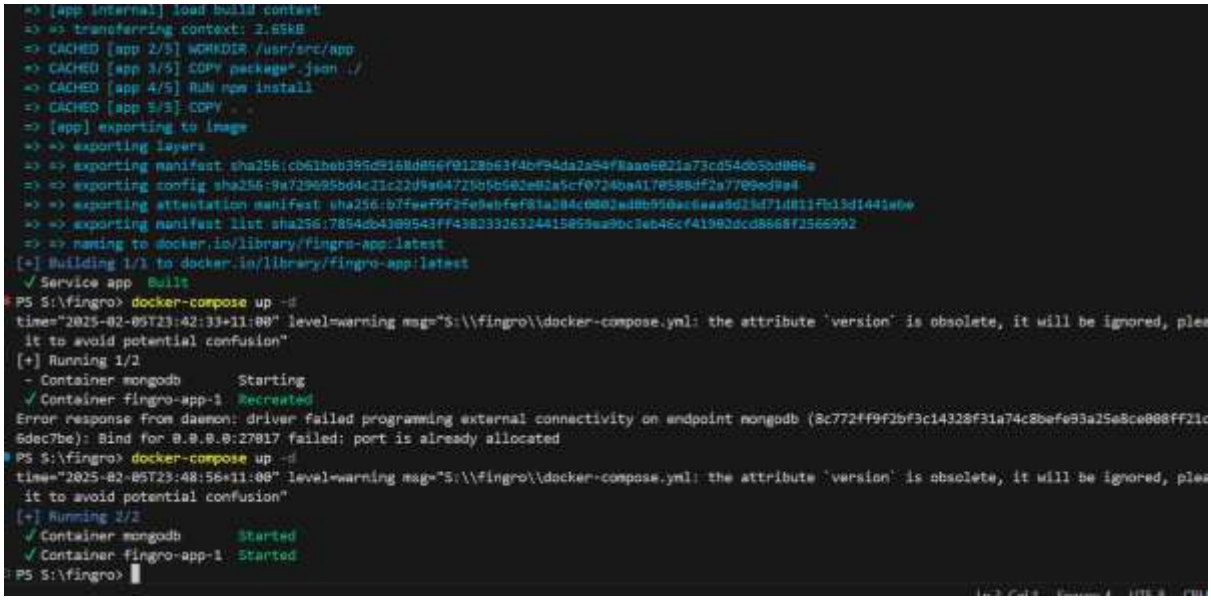
Below I am attaching Dockerization for weekly task


Week_7
controllers
models
companyModel.js
userModel.js
node_modules
public
routes
test
views
docker-compose.yml
Dockerfile
package-lock.json
package.json
server.js


Images
Containers
Volumes
Builds
Docker Hub
Docker Scout
Extensions
Local
Name
Tag
Image ID
Created
Size
Actions


[+] Building 4.6s (12/12) FINISHED
docker:desktop-linux
=> [app internal] load build definition from Dockerfile
=> => transferring dockerfile: 390B
=> [app internal] load metadata for docker.io/library/node:18
=> [app auth] library/node:pull token for registry-1.docker.io
=> [app internal] load .dockerignore
=> => transferring context: 2B
=> [app 1/5] FROM docker.io/library/node:18@sha256:f12d34408955b2081f82078e8f96c3299ca0f38d11e76086cb9b9b1b669977e4
=> => resolve docker.io/library/node:18@sha256:f12d34408955b2081f82078e8f96c3299ca0f38d11e76086cb9b9b1b669977e4
=> [app internal] load build context
=> => transferring context: 2.59kB
=> CACHED [app 2/5] WORKDIR /app
=> CACHED [app 3/5] COPY package* json ./
=> CACHED [app 4/5] RUN npm install
=> [app 5/5] COPY . .
=> [app] exporting to image
=> => exporting layers
=> => exporting manifest sha256:b3d452ea640ad263f9dd433871a4c6eed9f2ab82b0dbe1b98c845b3caafb85c9
=> => exporting config sha256:ee02eabe8ef958ab1b0fc317143f660158a3b7873c9e9962557b009fbfc06091
=> => exporting attestation manifest sha256:e9ea77a28e3f6340c0b9a5df1cdce1d99f7193a7ad39564a37de483782115c18
=> => exporting manifest list sha256:d6339932d5365748bc46bf819d6c153d9f6c52dcaf304744aad664388c87fef4a
=> => naming to docker.io/library/week_7-app:latest
[+] Building 1/1 to docker.io/library/week_7-app:latest
✓ Service app Built
PS S:\-SIT725\Week_7> docker-compose up -d
Open file in editor (ctrl + click)
time="2025-02-05T23:39:03+11:00" level=warning msg="S:\\-SIT725\\Week_7\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
✓ Container week_7-mongo-1 Running
✓ Container week_7-app-1 Started
PS S:\-SIT725\Week_7>


Logo
Save
Components
JavaScript
Go to Company Management
All Users
Name
Email
Phone
Actions
Shrikesh S P
Shri@gmail.com
0422101234
EMmalu
emma@gmail.com
0422101234
Add New User
Name
EMmalu
Phone
0422101234
Email
emma@gmail.com
ADD USER

PS S:\fingro> docker ps
>>
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
f0667b630f98 week_7-app "docker-entrypoint.s…" 2 minutes ago Up 2 minutes 0.0.0.0:3000->3000/tcp week_7-app-1
445f953bdced mongo "docker-entrypoint.s…" 12 minutes ago Up 12 minutes 0.0.0.0:27017->27017/tcp week_7-mongo-1
PS S:\fingro>

Below I am attaching Dockerization for Project