Steps for Execution:

1. Verify Docker Images

- Run the command:

```
sh
CopyEdit
docker images
```

- Confirm the presence of required images:
  - mahashwin/java-maven
  - mahashwin/ecommerce
  - nginx
  - hello-world
  - gcr.io/k8s-minikube/kicbase
  - node:alpine3.17
- If any required image is missing, pull it using:

```
sh
CopyEdit
docker pull <image_name>
```

2. Run Java-Maven Container

- Start the container with:

```
sh
CopyEdit
docker run -d -p 8090:8080 mahashwin/java-maven
```

- Check running containers:

```
sh
CopyEdit
docker ps
```

- Ensure the application is accessible at `http://localhost:8090`.

3. Build Docker Image for Java-Maven

- Navigate to the project directory and build the image:

```
sh
CopyEdit
docker build -t mahashwin/java-maven .
```

- Verify the image is built successfully:

```
sh
```

```
CopyEdit
docker images | grep java-maven
```

4. Deploy with Jenkins

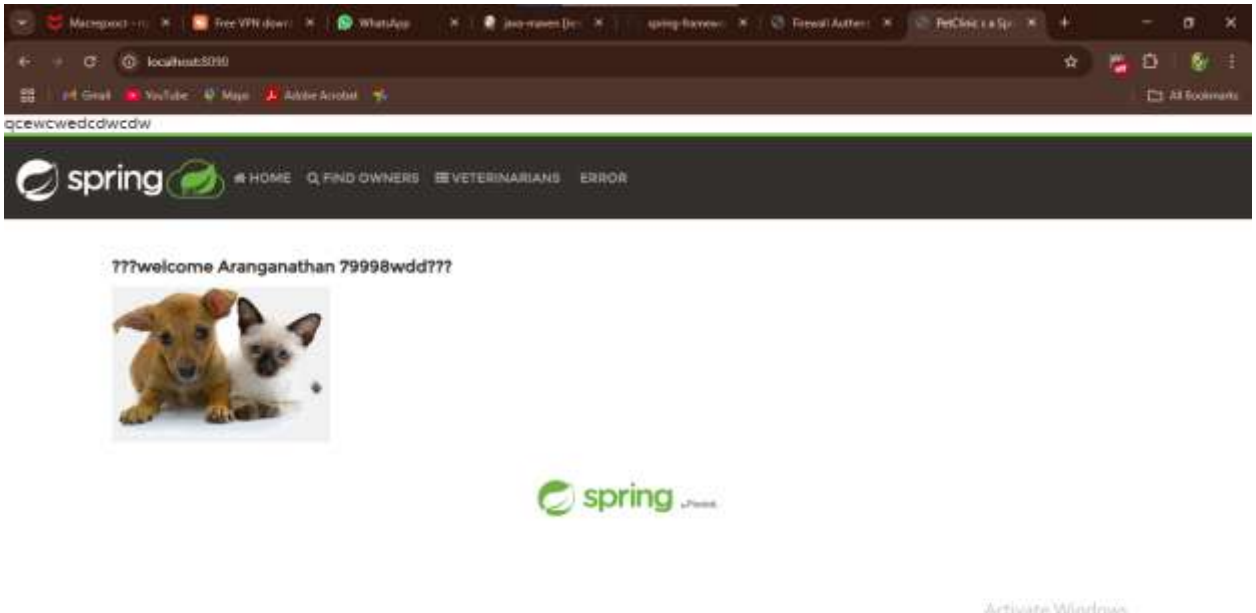
- Ensure Jenkins is running.
- Check the pipeline execution:
  - The build logs show:
    - `Step 4/4 : CMD ["catalina.sh", "run"]`
    - Successfully tagged `mahashwin/java-maven:latest`
  - This confirms that the image is built and tagged correctly.

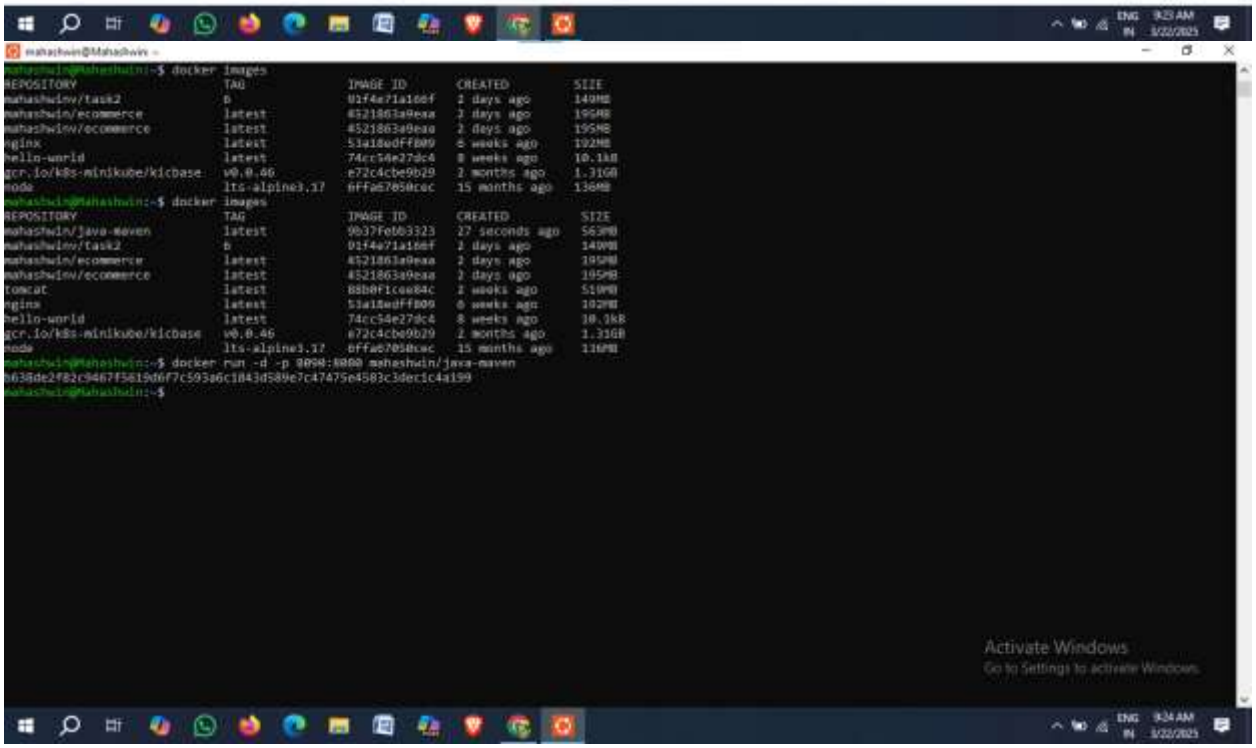
5. Verify Jenkins Build Status

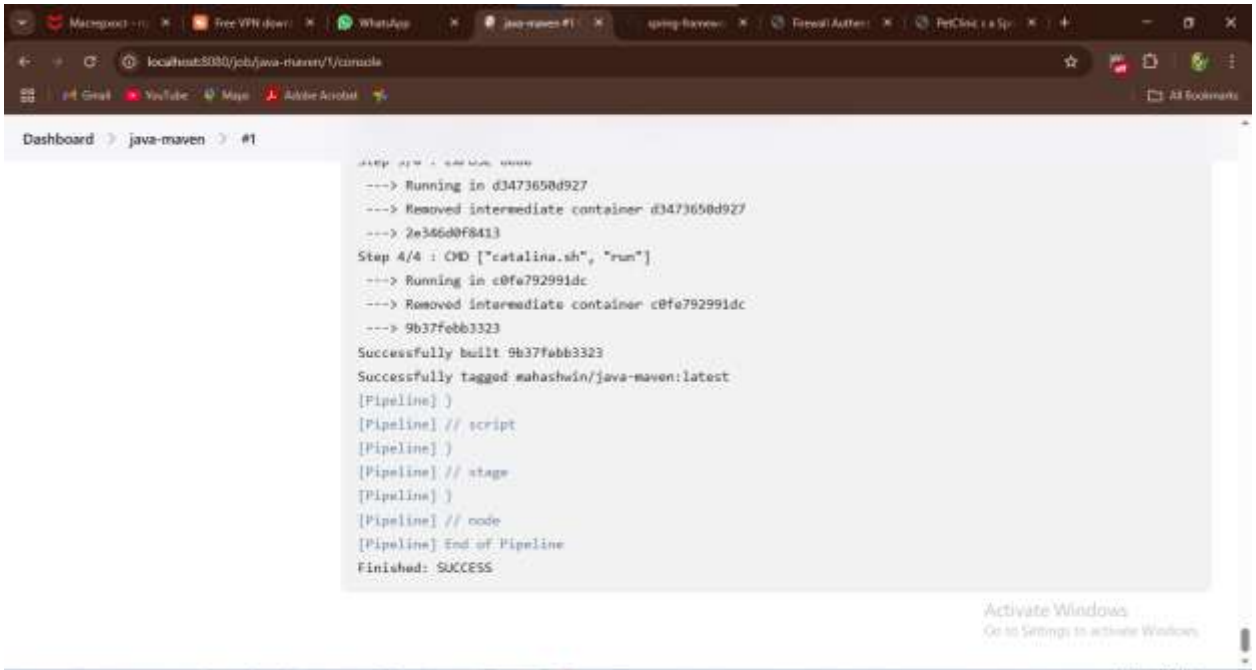
- Open Jenkins Dashboard (`http://localhost:8080`).
- Check the project status:
  - `java-maven` pipeline execution was **successful**.
- Ensure no pending builds in the queue.

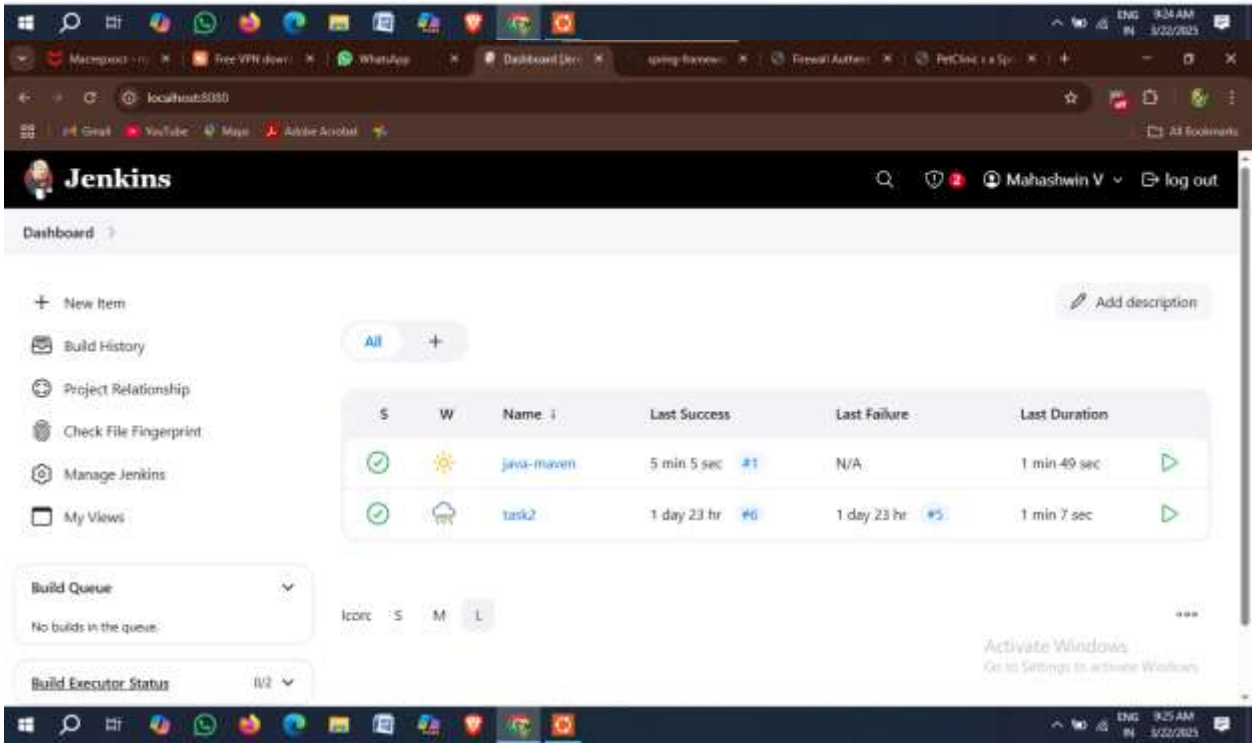
6. Verify Application Deployment

- Open `http://localhost:8090` in a browser.
- Ensure the application loads without errors.


localhost:8090
qcewcwedcdwcdw
spring
HOME
FIND OWNERS
VETERINARIANS
ERROR
???welcome Aranganathan 79998wdd???
spring
Activate Windows
Go to Settings to activate Windows.


maheshwin@Maheshwin: ~
maheshwin@Maheshwin:~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
maheshwinv/task2 0 01f4e71a160f 2 days ago 140MB
maheshwin/ecommerce latest 4521863a0eaa 2 days ago 195MB
maheshwinv/ecommerce latest 4521863a0eaa 2 days ago 195MB
nginx latest 53a18edff809 6 weeks ago 192MB
hello-world latest 74cc54e27dc4 8 weeks ago 10.1kB
gcr.io/k8s-minikube/kicbase v0.0.46 e72c4cbe9b29 2 months ago 1.31GB
node lts-alpine3.17 6ffa67050cec 15 months ago 136MB
maheshwin@Maheshwin:~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
maheshwin/java-maven latest 9b37feb03323 27 seconds ago 563MB
maheshwinv/task2 0 01f4e71a160f 2 days ago 140MB
maheshwin/ecommerce latest 4521863a0eaa 2 days ago 195MB
maheshwinv/ecommerce latest 4521863a0eaa 2 days ago 195MB
tomcat latest 88b0f1cea84c 2 weeks ago 519MB
nginx latest 53a18edff809 6 weeks ago 192MB
hello-world latest 74cc54e27dc4 8 weeks ago 10.1kB
gcr.io/k8s-minikube/kicbase v0.0.46 e72c4cbe9b29 2 months ago 1.31GB
node lts-alpine3.17 6ffa67050cec 15 months ago 136MB
maheshwin@Maheshwin:~$ docker run -d -p 8090:8080 maheshwin/java-maven
b638de2f82c9467f5819d6f7c593a6c1843d589e7c47475e4583c3dec1c4a199
maheshwin@Maheshwin:~$
Activate Windows
Go to Settings to activate Windows.


Dashboard > java-maven > #1
---> Running in d3473650d927
---> Removed intermediate container d3473650d927
---> 2e366d0f8413
Step 4/4 : CMD ["catalina.sh", "run"]
---> Running in c0fe792991dc
---> Removed intermediate container c0fe792991dc
---> 9b37febb3323
Successfully built 9b37febb3323
Successfully tagged mahashwin/java-maven:latest
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS


Jenkins
Mahashwin V
log out
Dashboard
New Item
Build History
Project Relationship
Check File Fingerprint
Manage Jenkins
My Views
Add description
All
S
W
Name
Last Success
Last Failure
Last Duration
java-maven
5 min 5 sec #1
N/A
1 min 49 sec
task2
1 day 23 hr #6
1 day 23 hr #5
1 min 7 sec
Build Queue
No builds in the queue.
Build Executor Status
Icon: S M L