

EXERCISE 8 - SWED

1.

Virtual Machine (VM): Emulates an entire operating system with its own kernel. Docker

Container: Runs applications in isolated processes using the host OS kernel.

VMs are heavy, slower to start, and include a full OS. Containers are lightweight, start fast, and share the host OS.

2.

c)

CI ensures that developers merge their changes frequently, and every commit is automatically tested, enabling fast feedback and reducing integration issues.

4.b) Jenkins

Jenkins is widely used for building, testing, and deploying applications in a CI/CD pipeline. (Kubernetes is for container orchestration, not CI/CD specifically.)

5.

Unit Tests: Test individual functions or components in isolation to ensure functional correctness.

Integration Tests: Verify that multiple components work together as expected.

API/Service Tests: Validate (RESTful) service interfaces, often using mock-ups or staging environments.

End-to-End or UI Tests: Simulate real user interactions across the entire system, often using tools such as Selenium.

Performance/Load Tests: Measure system behaviour under stress to detect bottlenecks.

Security Tests: Scan for vulnerabilities or misconfigurations

Smoke Tests: Quick, high-level tests to ensure the system isn't fundamentally broken after a change.

Regression Tests: Re-run existing test suites to ensure new changes don't break existing features

2.

docker file execution:

```
cd "$HOME\OneDrive\Bureau\WebMonitor"
```

```
docker build -t webmonitor .
```

```
docker run --rm webmonitor https://netflix.com
```

output expected:

```
Windows PowerShell
PS C:\Users\mehti\OneDrive\Bureau\WebMonitor> docker build -t webmonitor .
[+] Building 45.7s (9/9) FINISHED   docker:desktop-linux
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 177B 0.0s
=> [internal] load metadata for docker.io/library/e 2.1s
=> [internal] load .dockerignore 0.1s
=> => transferring context: 2B 0.0s
=> [1/4] FROM docker.io/library/eclipse-temurin:17 36.8s
=> => resolve docker.io/library/eclipse-temurin:17 0.1s
=> => sha256:d9815e77b54f33dbcd213d 2.28kB / 2.28kB 0.1s
=> => sha256:b61b7669ae30b45ca 144.65MB / 144.65MB 32.4s
=> => sha256:c70eff9bffbfe89f4317fcb6 160B / 160B 0.5s
=> => sha256:7bbafedfaf521e89144 22.95MB / 22.95MB 12.7s
=> => sha256:d9d352c11bbd3880007 29.72MB / 29.72MB 12.0s
=> => extracting sha256:d9d352c11bbd3880007953ed6ee 1.9s
=> => extracting sha256:7bbafedfaf521e89144400b08c3 1.4s
=> => extracting sha256:b61b7669ae30b45ca88696ba9b5 3.4s
=> => extracting sha256:c70eff9bffbfe89f4317fcb62 0.1s
=> => extracting sha256:d9815e77b54f33dbcd213d596ee 0.0s
=> [internal] load build context 0.2s
=> => transferring context: 574B 0.0s
=> [2/4] WORKDIR /app 2.2s
=> [3/4] COPY src/ ./src 0.3s
=> [4/4] RUN javac src/*.java -d out 2.7s
=> exporting to image 0.9s
=> => exporting layers 0.6s
=> => exporting manifest sha256:699baadb9ef568366e 0.0s
=> => exporting config sha256:ea38221995e25b9eade64 0.0s
=> => exporting attestation manifest sha256:190a0b0 0.1s
=> => exporting manifest list sha256:f07b96e49b9991 0.0s
=> => naming to docker.io/library/webmonitor:latest 0.0s
=> => unpacking to docker.io/library/webmonitor:lat 0.1s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/y9uzh2p628csjr5af9aatsc06
PS C:\Users\mehti\OneDrive\Bureau\WebMonitor> docker run --rm webmonitor https://netflix.com
PS C:\Users\mehti\OneDrive\Bureau\WebMonitor> java -cp out Main https://netflix.com
Error: Could not find or load main class Main
Caused by: java.lang.ClassNotFoundException: Main
PS C:\Users\mehti\OneDrive\Bureau\WebMonitor> docker run --rm webmonitor https://netflix.com
User Mehdi notified about update on: https://netflix.com
Notifying Mehdi@example.com via EMAIL: Website updated: https://netflix.com
```