

BÀI TẬP TRÊN LỚP

MÔN HỌC: HỆ PHÂN TÁN

CHƯƠNG 1: TỔNG QUAN VÀ KIẾN TRÚC HỆ PHÂN TÁN

HỌ TÊN SV: Nguyễn Đức Thắng

MSSV: 20225089

MÃ LỚP: 157542

MÃ HỌC PHẦN: IT4611

1. Web server apache2

Tạo file docker-compose.yaml

services:

apache-server:

image: ubuntu/apache2

container_name: apache-server

ports:

- "80:80"

volumes:

- ./example.com/public_html:/var/www/example.com/public_html

- ./test.com/public_html:/var/www/test.com/public_html

- ./example.com/example.com.conf:/etc/apache2/sites-available/example.com.conf #

Mount file cấu hình example.com

- ./test.com/test.com.conf:/etc/apache2/sites-available/test.com.conf

- ./logs:/var/log/apache2

command: >

/bin/bash -c "a2ensite example.com.conf &&

a2ensite test.com.conf &&

service apache2 restart &&

`tail -f /var/log/apache2/error.log"`

Run docker-compose up -d

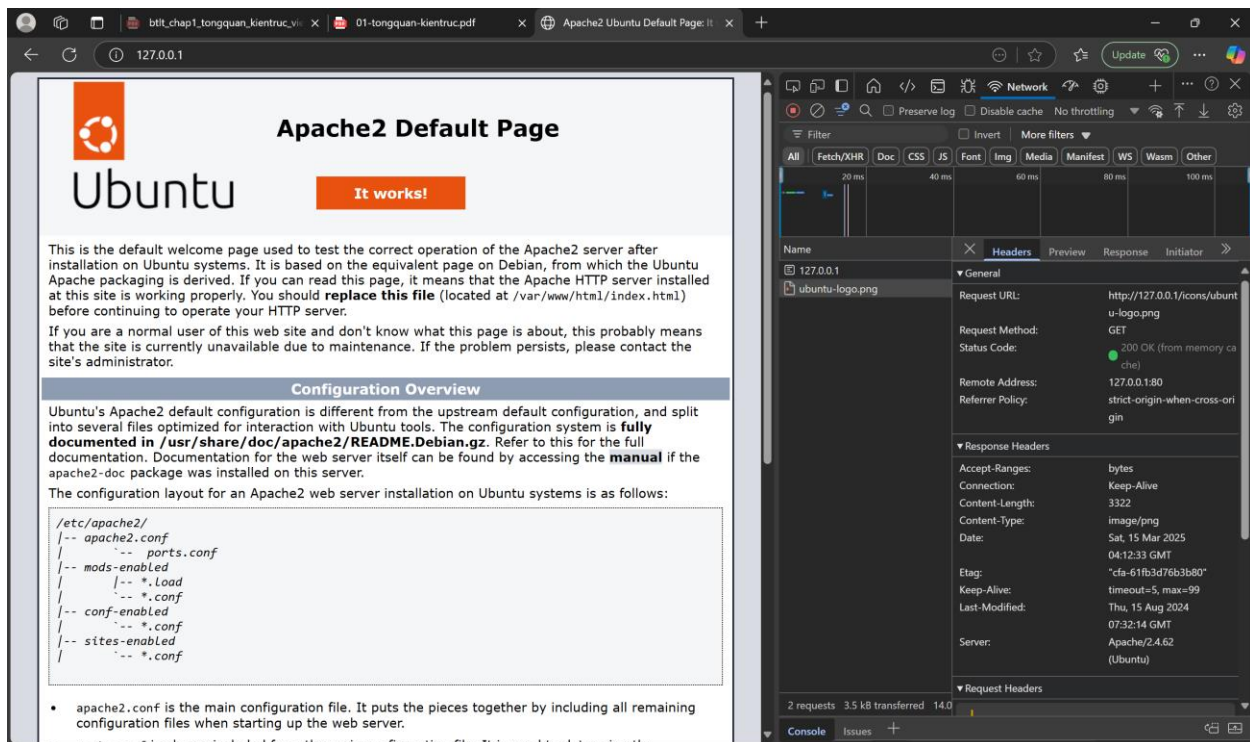
- Câu hỏi 1: Đường dẫn đến file html chứa nội dung mặc định của trang web các bạn vừa xem là gì?

`/var/www/html/index.html`



- Câu hỏi 2: Cổng mặc định của dịch vụ www là gì?

Cổng 80

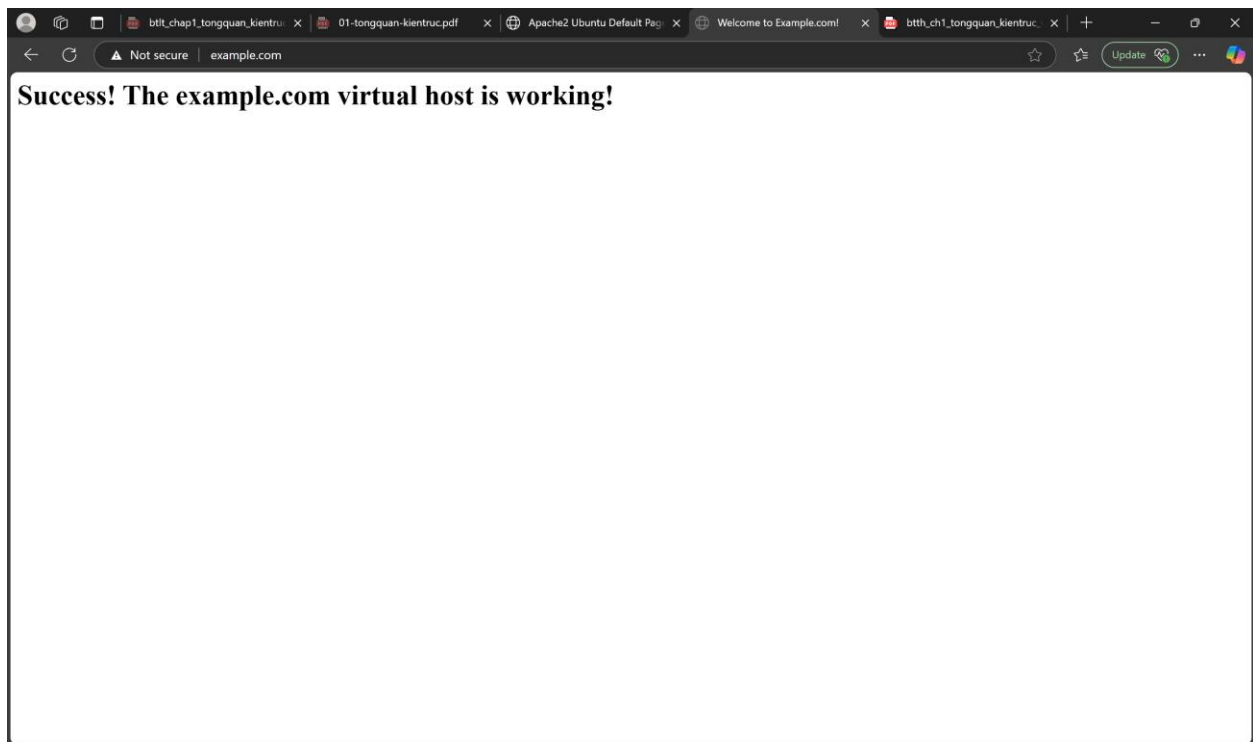


- Câu hỏi 3: Hãy giải thích quyền mang số 755 là gì?

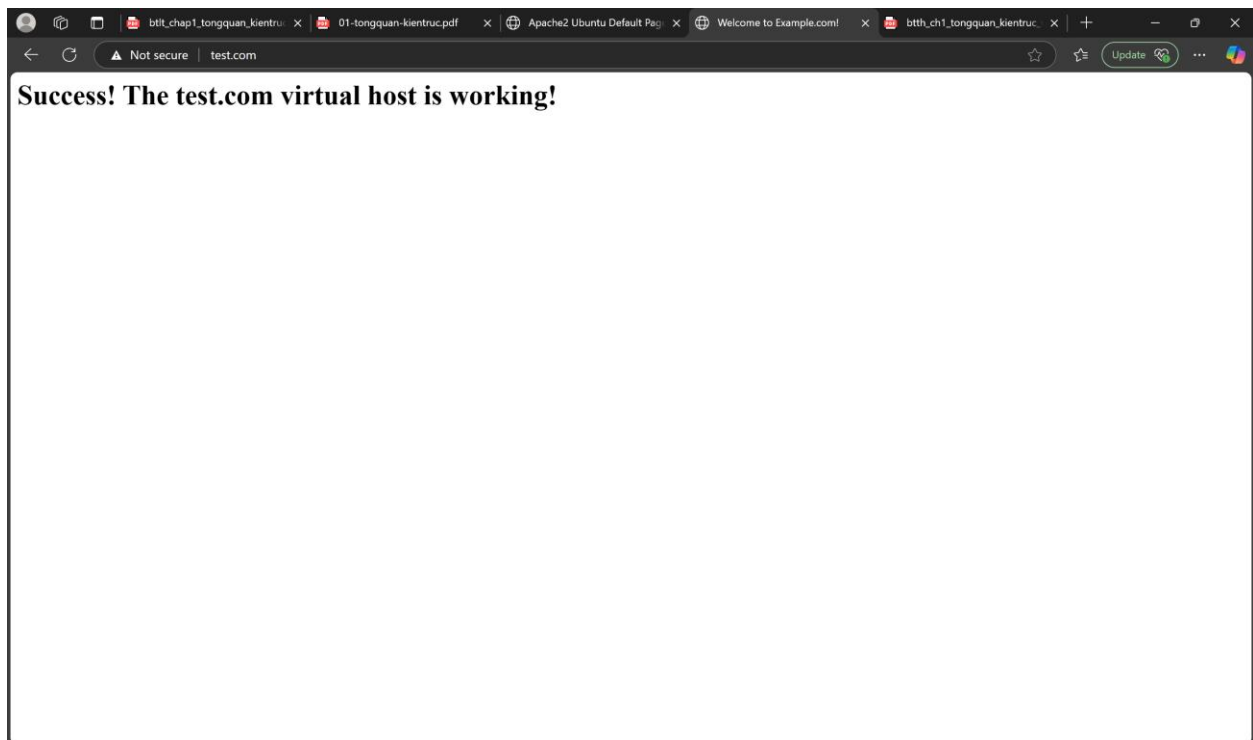
Cho phép chủ sở hữu truy cập, sửa file theo cách họ muốn. Mọi người khác chỉ có thể truy cập mà không có quyền sửa nó. Tất cả mọi người đều có quyền thực thi.

Câu hỏi 4: Bạn quan sát thấy nội dung gì sau khi gõ 2 địa chỉ trên? Giải thích.

- “Welcome to Example.com!”, “Success! The example.com virtual host is working!” trong phần nội dung.



- “Welcome to Test.com!”, “Success! The test.com virtual host is working!” trong phần nội dung.

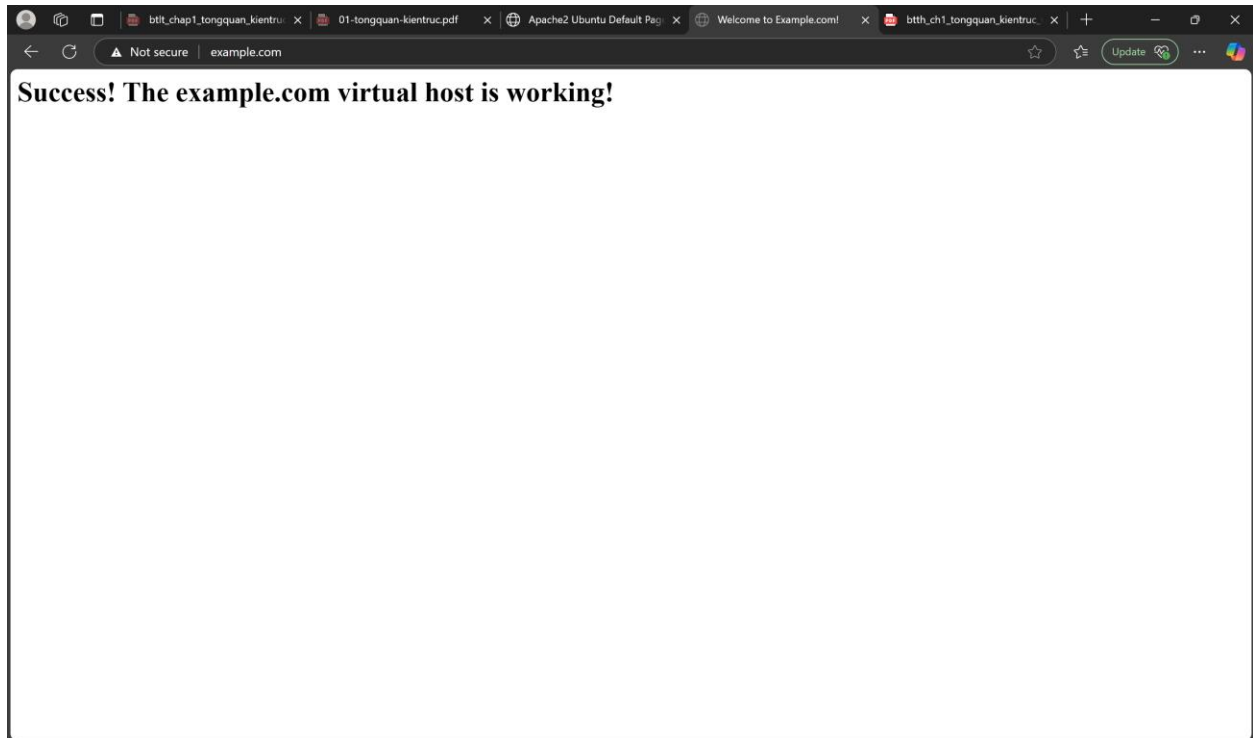


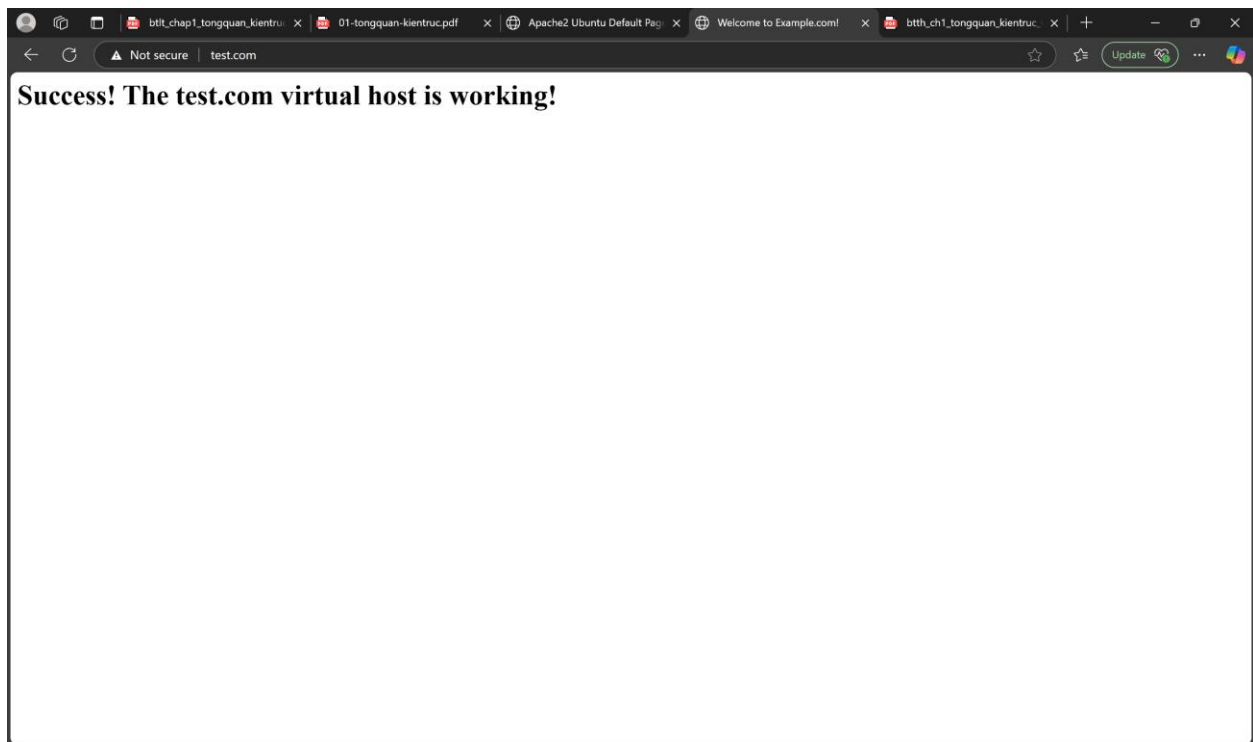
- Giải thích: Trang web đã được thực thi và chủ sở hữu có thể nhìn thấy giao diện web.

Câu hỏi 5: Thử truy cập từ các máy tính khác trong cùng mạng LAN vào 2 trang web đó.

- Phải cấu hình file hosts và proxy mới có thể truy cập được.

- Kết quả tại máy chính (Windows) bật kết nối Host-only với máy ảo (Ubuntu) (cùng 1 LAN).





2. Interface trong Java

- Câu hỏi 6: Hãy tự viết một đoạn code để thực hiện 1 vòng lặp `while` sao cho nó sẽ nhận các số mà người dùng gõ và gửi về server, cho đến khi nào người dùng gõ ký tự rỗng rồi ấn enter.

Gợi ý: hãy dùng lệnh sau để nhận xâu ký tự người dùng gõ vào: `String message = scanner.nextLine();`

```
// Khởi tạo Scanner để nhập dữ liệu từ người dùng
Scanner scanner = new Scanner(System.in);
while (true) {
    System.out.print("Enter numbers separated by spaces (or press Enter to exit): ");
    String message = scanner.nextLine();
    if (message.isEmpty()) {
        break;
    }
    out.println(message);
    String response = in.readLine();
    System.out.println("Sorted numbers: " + response);
}
// Đóng kết nối
scanner.close();
```

- Câu hỏi 7: Vai trò của phương thức `run` là gì? Khi nào thì nó được gọi?

Phương thức “run” thực thi luồng tương ứng nếu phương thức “start” được gọi. Khi có một kết nối từ Client đến Server, một thread mới tại server được tạo ra, phương thức “start” được gọi để bắt đầu thực thi phương thức “run”.

3. Kiến trúc Microservices

Build/rebuild 3 dịch vụ:

```
[INFO] Unpacked only-script type wrapper distribution org.apache.maven.wrapper:maven-wrapper-distribution:zip:only-script:3.3.2
[INFO] Configuring .mvn/wrapper/maven-wrapper.properties to use Maven 3.9.9 and download from https://repo.maven.apache.org/maven2
[INFO] -----
[INFO] Reactor Summary for microservice-kubernetes-demo 0.0.1-SNAPSHOT:
[INFO]
[INFO] microservice-kubernetes-demo ..... SUCCESS [ 2.157 s]
[INFO] microservice-kubernetes-demo-customer ..... SKIPPED
[INFO] microservice-kubernetes-demo-catalog ..... SKIPPED
[INFO] microservice-kubernetes-demo-order ..... SKIPPED
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 9.480 s
[INFO] Finished at: 2025-03-15T21:35:31+07:00
[INFO] -----
PS E:\exercises\IT4611\btcl\microservice-demo>
```

Câu hỏi 1: Hãy thực hiện gõ những lệnh tương tự như trên với 3 dịch vụ còn lại.

docker build --tag=microservice-kubernetes-demo-apache apache

```
=> sha256:fb15d46c38dcd1ea8b1990806c3366ecd10c79d374f341687eb2cb23a2c8672e 170B / 170B 0.5s
=> sha256:da8ef40b9ecabc2679fe2419957220c0272a965c5cf7e0269falaeeb8c56f2e1 528B / 528B 0.8s
=> sha256:b51569e7c50720acf6860327847fe342a1afbe148d24c529fb81df105e3eed01 857B / 857B 1.1s
=> sha256:58690f9b18fca6469a14da4e212c96849469f9b1be6661d2342a4bf01774aa50 46.50MB / 46.50MB 58.6s
=> extracting sha256:58690f9b18fca6469a14da4e212c96849469f9b1be6661d2342a4bf01774aa50 0.9s
=> extracting sha256:b51569e7c50720acf6860327847fe342a1afbe148d24c529fb81df105e3eed01 0.0s
=> extracting sha256:da8ef40b9ecabc2679fe2419957220c0272a965c5cf7e0269falaeeb8c56f2e1 0.0s
=> extracting sha256:fb15d46c38dcd1ea8b1990806c3366ecd10c79d374f341687eb2cb23a2c8672e 0.0s
=> [internal] load build context
=> transferring context: 1.88kB 0.0s
=> [2/5] RUN apt-get update ; apt-get dist-upgrade -y -qq 13.8s
=> [3/5] RUN apt-get install -y -qq apache2 && a2enmod proxy proxy_http proxy_ajp rewrite deflate headers proxy_connect proxy_html lbmethod_byrequests && 43.9s
=> [4/5] COPY index.html /var/www/html/index.html 0.0s
=> [5/5] COPY 000-default.conf /etc/apache2/sites-enabled/000-default.conf 0.0s
=> exporting to image 4.1s
=> exporting layers 3.2s
=> exporting manifest sha256:1c26f28be4cf63beb5cd25ddd50fc46ff7a6bfd6d6df50fbd0772475529b955 0.0s
=> exporting config sha256:fd384cf32e12eef6074742d7b7cf255591c668ab956b99bf3a4853db49305c41 0.0s
=> exporting attestation manifest sha256:ebb8dea4222c3727120133a513a55233fc9fda5c45df95c83d86e363cd6a3925 0.0s
=> exporting manifest list sha256:202a17f4dcf1fd4ee1be1551a1c927e471c9305226bb4a768a1d27394ac9bbf 0.0s
=> naming to docker.io/library/microservice-kubernetes-demo-apache:latest 0.0s
=> unpacking to docker.io/library/microservice-kubernetes-demo-apache:latest 0.9s

view build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/zdwrld1u2jul9v1i6noh3tj1u

1 warning found (use docker: --debug to expand):
- JSONArgsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS signals (line 16)
PS E:\exercises\IT4611\btcl\microservice-demo>
```

docker tag microservice-kubernetes-demo- apache thangit67/microservice-kubernetes-demo- apache:latest

docker push thangit67/microservice-kubernetes-demo-apache

```
f5dc8524418f: Pushed
da8ef40b9eca: Pushed
b51569e7c507: Pushed
58690f9b18fc: Pushing [=====>] 19.92MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
7c36cc874227: Pushing [=====>] 8.389MB/33.86MB
f3b41828c776: Pushed
58690f9b18fc: Pushing [=====>] 20.97MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 20.97MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 20.97MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 20.97MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 20.97MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 22.02MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushing [=====>] 22.02MB/46.5MB
0069142d94b4: Pushing [=====>] 6.291MB/20.31MB
58690f9b18fc: Pushed
0069142d94b4: Pushed
7c36cc874227: Pushed
f3b41828c776: Pushed
latest: digest: sha256:202a17f4dcf1fd4ee1be1551a1c927e4171c9305226bb4a768a1d27394ac9bbf size: 856
PS E:\exercises\IT4611\btcl\microservice-demo>
```

docker build --tag=microservice-kubernetes-demo-order microservice-kubernetes-demo-order

```
PS E:\exercises\IT4611\btcl\microservice-demo> docker build --tag=microservice-kubernetes-demo-order microservice-kubernetes-demo-order
[+] Building 20.8s (8/8) FINISHED                                docker:desktop-linux
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 247B                               0.0s
=> [internal] load metadata for docker.io/library/openjdk:10.0.2-jre-slim 2.9s
=> [auth] library/openjdk:pull token for registry-1.docker.io    0.0s
=> [internal] load .dockerignore                                  0.0s
=> => transferring context: 2B                                     0.0s
=> [internal] load build context                                  0.8s
=> => transferring context: 41.38MB                                0.8s
=> [1/2] FROM docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da40b34d146ae5fee266908836e55328b3788e84af2253 14.0s
=> => resolve docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da40b34d146ae5fee266908836e55328b3788e84af2253 0.0s
=> => sha256:cf7ae0d40212ecf57b3d98406b3812c7171c16286bfb9450da71b14c228dff5 131B / 131B 0.8s
=> => sha256:0ab64f7fb1486a34cb500304417350ba3833b16369cf9357b484527d4e4eeaca 78.39MB / 78.39MB 11.7s
=> => sha256:d1b0083010f5a1580b0f0d04ae3fef44e71ec2e3f9aa6e8ff466b096857324d5 237B / 237B 1.2s
=> => sha256:096ebee0eb6534eadf6d1ae8173145e8f43aedf29f5ae179f8b2e899f84f344 26.57MB / 26.57MB 9.4s
=> => sha256:3ac6aae685cb5ee685511876e43026b65163450ee8c4a9f0ffabe065911b11d7 460.38kB / 460.38kB 0.7s
=> => extracting sha256:096ebee0eb6534eadf6d1ae8173145e8f43aedf29f5ae179f8b2e899f84f344 2.1s
=> => extracting sha256:3ac6aae685cb5ee685511876e43026b65163450ee8c4a9f0ffabe065911b11d7 0.2s
=> => extracting sha256:d1b0083010f5a1580b0f0d04ae3fef44e71ec2e3f9aa6e8ff466b096857324d5 0.0s
=> => extracting sha256:cf7ae0d40212ecf57b3d98406b3812c7171c16286bfb9450da71b14c228dff5 0.0s
=> => extracting sha256:0ab64f7fb1486a34cb500304417350ba3833b16369cf9357b484527d4e4eeaca 2.1s
=> [2/2] COPY target/microservice-kubernetes-demo-order-0.0.1-SNAPSHOT.jar . 0.7s
=> exporting to image 2.5s
=> => exporting layers 2.1s
```

docker tag microservice-kubernetes-demo-order thangit67/microservice-kubernetes-demo-order:latest

docker push thangit67/microservice-kubernetes-demo-order


```

PS E:\exercises\IT4611\btcl\microservice-demo> docker tag microservice-kubernetes-demo-order thangit67/microservice-kubernetes-demo-order:latest
PS E:\exercises\IT4611\btcl\microservice-demo> docker push thangit67/microservice-kubernetes-demo-order
Using default tag: latest
The push refers to repository [docker.io/thangit67/microservice-kubernetes-demo-order]
01e57c9ee221: Pushed
cd3579e05cb5: Pushed
3ac6aea685cb: Pushed
cf7ae0d40212: Pushed
0ab64f7fb148: Pushed
096ebeeadeb6: Pushed
d1b0083010f5: Pushed
latest: digest: sha256:c636327453e77ce9897a61c9a627e4434f48bc74fc7d4e730243af6a1a8ff9da size: 856
PS E:\exercises\IT4611\btcl\microservice-demo>

```

docker build --tag=microservice-kubernetes-demo-catalog microservice-kubernetes-demo-catalog

```

latest: digest: sha256:c636327453e77ce9897a61c9a627e4434f48bc74fc7d4e730243af6a1a8ff9da size: 856
PS E:\exercises\IT4611\btcl\microservice-demo> docker build --tag=microservice-kubernetes-demo-catalog microservice-kubernetes-demo-catalog
[*] Building 7.2s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 251B
=> [internal] load metadata for docker.io/library/openjdk:10.0.2-jre-slim
=> [auth] library/openjdk:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 41.37MB
=> CACHED [1/2] FROM docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da48b34d146ae5fee266908836e55328b3788e84af2253
=> => resolve docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da48b34d146ae5fee266908836e55328b3788e84af2253
=> [2/2] COPY target/microservice-kubernetes-demo-catalog-0.0.1-SNAPSHOT.jar .
=> exporting to image
=> => exporting layers
=> => exporting manifest sha256:e378a272cc6552c0c82caa0f43d0c90ad261b894862928fbf9f285a5a3157cf7
=> => exporting config sha256:17c28c055a8297833736a8111697c3326c0140c52ea1435c0490779ced5cc6a7
=> => exporting attestation manifest sha256:fd0bd32b1a51b59b7da5e644f67671fdfb4b67ccc7862c9647b434355407f2de
=> => exporting manifest list sha256:a8c96a83df06204e02ce2d236c0c55798b389ec1d8490e5fe05651f8e2f1d15e
=> => naming to docker.io/library/microservice-kubernetes-demo-catalog:latest
=> => unpacking to docker.io/library/microservice-kubernetes-demo-catalog:latest

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/4vo9x87pygzvxa84f88p41vs5

1 warning found (use docker --debug to expand):
- JSONArgsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS signals (line 3)
PS E:\exercises\IT4611\btcl\microservice-demo>

```

docker tag microservice-kubernetes-demo-catalog thangit67/microservice-kubernetes-demo-catalog:latest

docker push thangit67/microservice-kubernetes-demo-catalog

```

PS E:\exercises\IT4611\btcl\microservice-demo> docker tag microservice-kubernetes-demo-catalog thangit67/microservice-kubernetes-demo-catalog:latest
PS E:\exercises\IT4611\btcl\microservice-demo> docker push thangit67/microservice-kubernetes-demo-catalog
Using default tag: latest
The push refers to repository [docker.io/thangit67/microservice-kubernetes-demo-catalog]
774a0cad9282: Pushed
a0215617f272: Pushed
3ac6aea685cb: Mounted from thangit67/microservice-kubernetes-demo-order
cf7ae0d40212: Mounted from thangit67/microservice-kubernetes-demo-order
096ebeeadeb6: Mounted from thangit67/microservice-kubernetes-demo-order
d1b0083010f5: Mounted from thangit67/microservice-kubernetes-demo-order
0ab64f7fb148: Mounted from thangit67/microservice-kubernetes-demo-order
latest: digest: sha256:a8c96a83df06204e02ce2d236c0c55798b389ec1d8490e5fe05651f8e2f1d15e size: 856
PS E:\exercises\IT4611\btcl\microservice-demo>

```

docker build --tag=microservice-kubernetes-demo-customer microservice-kubernetes-demo-customer

```

PS E:\exercises\IT4611\btcl\microservice-demo> docker build --tag=microservice-kubernetes-demo-customer microservice-kubernetes-demo-customer
[+] Building 8.0s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 253B
=> [internal] load metadata for docker.io/library/openjdk:10.0.2-jre-slim
=> [auth] library/openjdk:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 41.37MB
=> CACHED [1/2] FROM docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da48b34d146ae5fee266908836e55328b3788e84af2253
=> => resolve docker.io/library/openjdk:10.0.2-jre-slim@sha256:256f5b4dd532b43336da48b34d146ae5fee266908836e55328b3788e84af2253
=> [2/2] COPY target/microservice-kubernetes-demo-customer-0.0.1-SNAPSHOT.jar .
=> exporting to image
=> => exporting layers
=> => exporting manifest sha256:ce0e2075e19823c95c17660a870a930980c2c8d70418925e962a20a5bac8d4c
=> => exporting config sha256:d395edaceebdf733dfbaf88574f19d41f8d0bfe3824687feb55c9c3bde6c493
=> => exporting attestation manifest sha256:7c5200546c5b8cb2d0cadaa170839411e378d2f1ca3185bb9b2dddb8b003746
=> => exporting manifest list sha256:5731118e53a7dc9116da19476def52736581dc49eda0bf75858f6fcc4bdd300b
=> => naming to docker.io/library/microservice-kubernetes-demo-customer:latest
=> => unpacking to docker.io/library/microservice-kubernetes-demo-customer:latest

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/xmkmsf6jgulgqkt7b195mq8t

1 warning found (use docker --debug to expand):
 - JSONArgsRecommended: JSON arguments recommended for CND to prevent unintended behavior related to OS signals (line 3)
PS E:\exercises\IT4611\btcl\microservice-demo>

```

docker tag microservice-kubernetes-demo-customer thangit67/microservice-kubernetes-demo-customer:latest

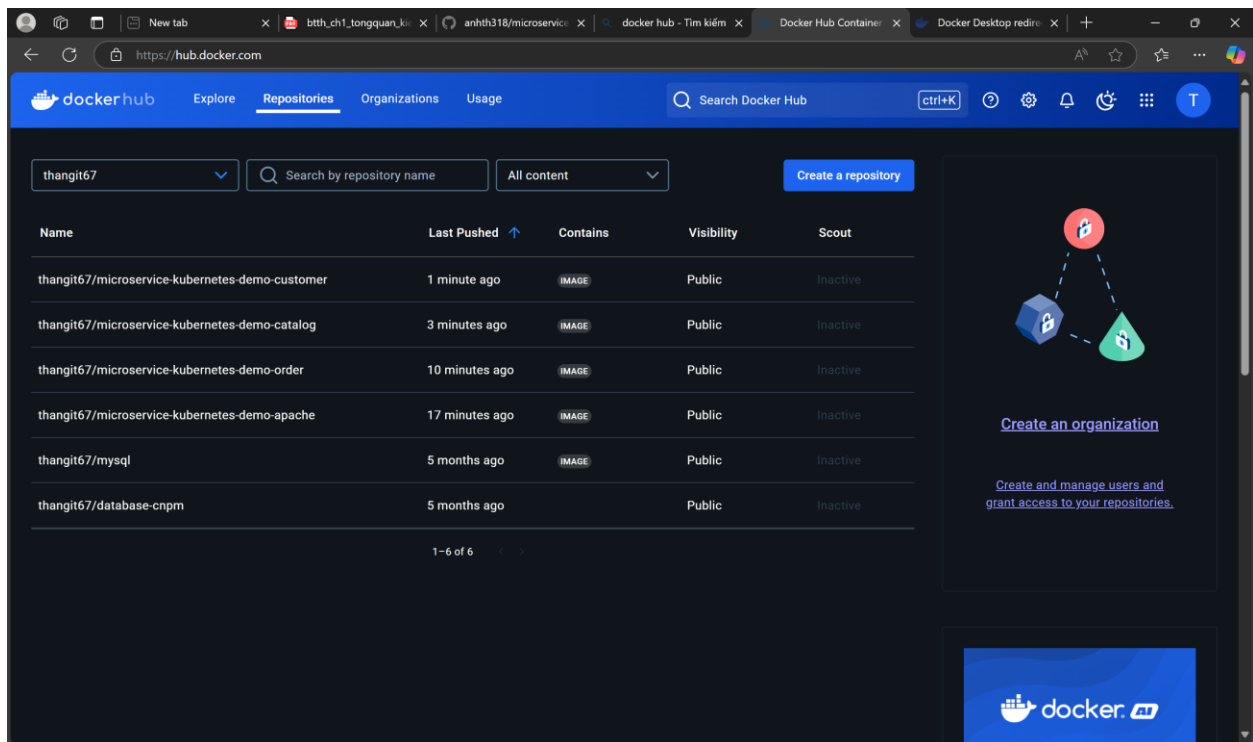
docker push thangit67/microservice-kubernetes-demo-customer

```

PS E:\exercises\IT4611\btcl\microservice-demo> docker tag microservice-kubernetes-demo-customer thangit67/microservice-kubernetes-demo-customer:latest
PS E:\exercises\IT4611\btcl\microservice-demo> docker push thangit67/microservice-kubernetes-demo-customer
Using default tag: latest
The push refers to repository [docker.io/thangit67/microservice-kubernetes-demo-customer]
68521cc06c04: Pushed
6dfc39b2a948: Pushed
3ac6aea685cb: Mounted from thangit67/microservice-kubernetes-demo-order
d1b0083010f5: Mounted from thangit67/microservice-kubernetes-demo-order
096ebeeadeb6: Mounted from thangit67/microservice-kubernetes-demo-order
cf7ae0d40212: Mounted from thangit67/microservice-kubernetes-demo-order
0ab64f7fb148: Mounted from thangit67/microservice-kubernetes-demo-order
latest: digest: sha256:5731118e53a7dc9116da19476def52736581dc49eda0bf75858f6fcc4bdd300b size: 856
PS E:\exercises\IT4611\btcl\microservice-demo>

```

Câu hỏi 2: Vào trang web DockerHub và đăng nhập vào tài khoản của bạn. Bạn thấy những gì mới xuất hiện trên docker hub repository của bạn?



Các image mới

thangit67/microservice-kubernetes-demo-customer

thangit67/microservice-kubernetes-demo-catalog

thangit67/microservice-kubernetes-demo-order

thangit67/microservice-kubernetes-demo-apache

kubecttl apply -f microservices.yaml

```
PS E:\exercises\IT4611\btc1\microservice-demo> kubecttl apply -f microservices.yaml
deployment.apps/apache created
deployment.apps/catalog created
deployment.apps/customer created
deployment.apps/order created
service/apache created
service/catalog created
service/customer created
service/order created
PS E:\exercises\IT4611\btc1\microservice-demo>
```

kubecttl get all

```
PS E:\exercices\IT4611\btcl\microservice-demo> kubectl get all

NAME                READY   STATUS    RESTARTS   AGE
pod/apache-84d69cb6c9-px64r   1/1     Running   0           28s
pod/catalog-56597ddb56-s9dvv   1/1     Running   0           28s
pod/customer-55645fb569-ffpgc   1/1     Running   0           28s
pod/order-5b9cb745df-gc8t5     1/1     Running   0           28s

NAME                TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
service/apache      LoadBalancer  10.103.110.151 localhost     80:30305/TCP     28s
service/catalog     LoadBalancer  10.104.215.33 localhost     8080:31129/TCP   28s
service/customer    LoadBalancer  10.103.83.67 <pending>     8080:32079/TCP   28s
service/kubernetes  ClusterIP     10.96.0.1    <none>        443/TCP          18d
service/order       LoadBalancer  10.103.87.222 <pending>     8080:32082/TCP   28s

NAME                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/apache  1/1     1             1           28s
deployment.apps/catalog  1/1     1             1           28s
deployment.apps/customer  1/1     1             1           28s
deployment.apps/order    1/1     1             1           28s

NAME                DESIRED   CURRENT   READY   AGE
replicaset.apps/apache-84d69cb6c9  1         1         1       28s
replicaset.apps/catalog-56597ddb56  1         1         1       28s
replicaset.apps/customer-55645fb569  1         1         1       28s
replicaset.apps/order-5b9cb745df     1         1         1       28s
PS E:\exercices\IT4611\btcl\microservice-demo>
```

Câu hỏi 3: Trạng thái (status) của các pods vừa mới tạo được là gì? Bây giờ, hãy chờ vài phút và gõ lại lệnh đó, trạng thái mới của các pods giờ đã chuyển thành gì?

Ban đầu các pods mới tạo ở trạng thái “ContainerCreating”, sau đó chuyển sang trạng thái “Running”.

4. Kiến trúc JMS và DDS

Tạo fill docker-compose/yaml

services:

payara:

image: payara/server-full

container_name: payara-server

ports:

- "8080:8080"

- "4848:4848"

environment:

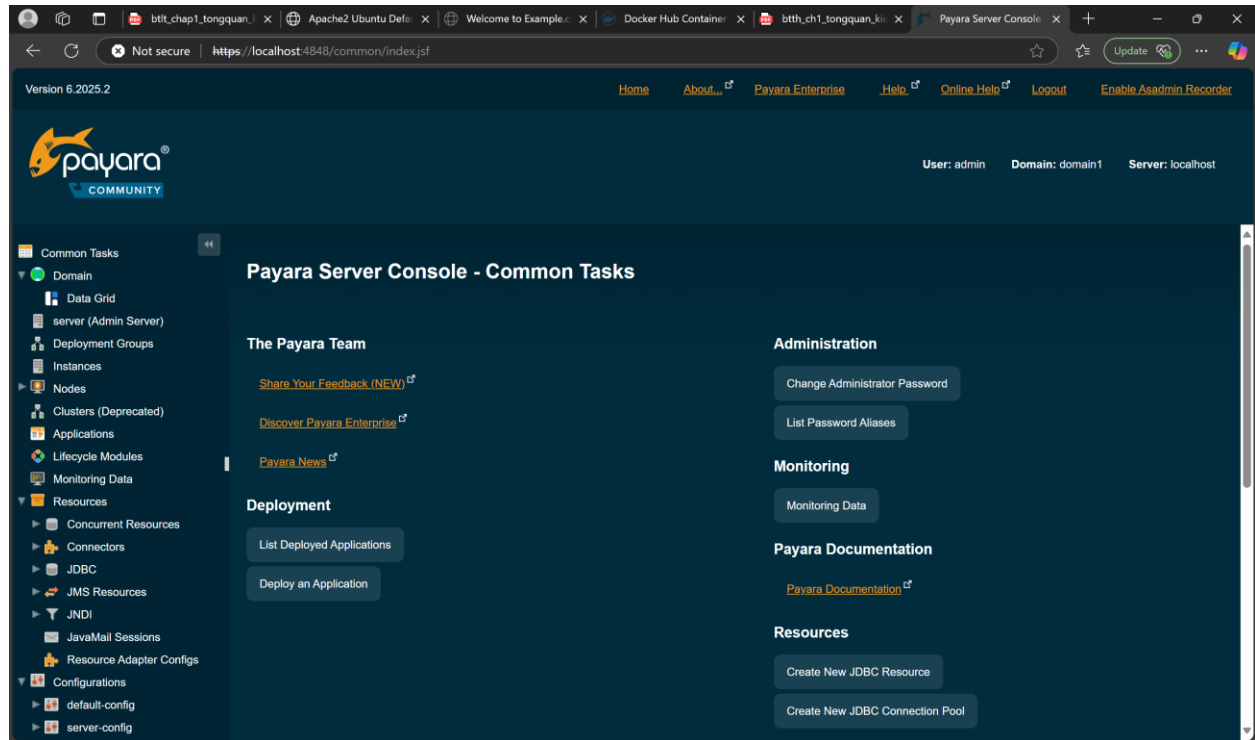
- PAYARA_DOMAIN=domain1

restart: always

run **docker-compose up -d**

Câu hỏi 1: Giải thích vai trò của application server glassfish.

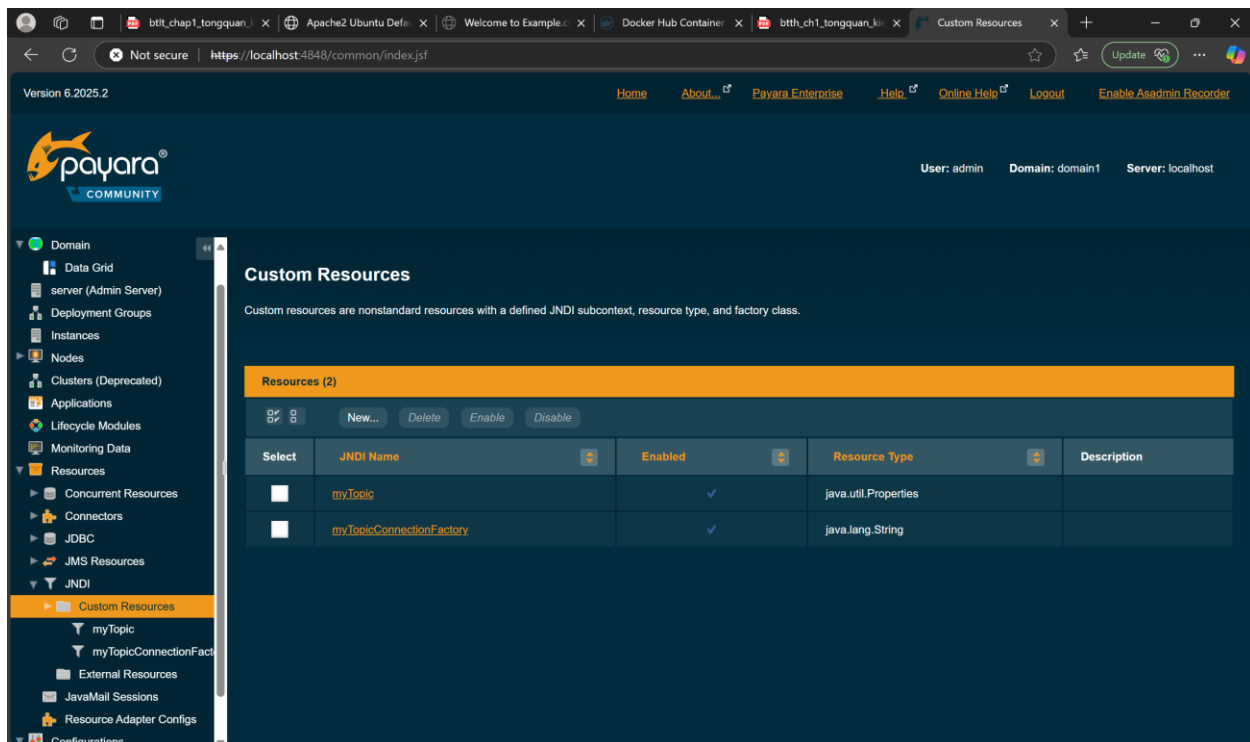
Do hiện tại image của glassfish đã hỏng sử dụng một server khác dựa trên glassfish để thay thế tên payara.



Payara Server là một máy chủ ứng dụng mã nguồn mở, dựa trên **GlassFish**, được thiết kế để chạy các ứng dụng **Java EE/Jakarta EE**. Vai trò chính của nó bao gồm:

1. **Triển khai ứng dụng doanh nghiệp:** Hỗ trợ các ứng dụng web, EJB, REST, SOAP.
2. **Hỗ trợ microservices và cloud:** Tích hợp với Docker, Kubernetes, và các nền tảng đám mây.
3. **Quản lý và giám sát:** Cung cấp giao diện quản trị (Admin Console) và công cụ dòng lệnh (asadmin).
4. **Bảo mật và hiệu suất:** Cung cấp tính năng bảo mật nâng cao và hỗ trợ clustering.

Câu hỏi 2: Tại sao lại phải tạo 2 JNDI như trên?



- Phải tạo 2 JNDI vì

myTopicConnectionFactory để payara tạo một connector connection pool và connector resource.

myTopic để payara tạo connector admin object resource.

Câu hỏi 3: Sau khi chạy thử chương trình Sender và Receiver, vận dụng lý thuyết kiến trúc hướng sự kiện đã học trên lớp để giải thích cơ chế chuyển và nhận thông điệp của Sender và Receiver.

Cơ chế chuyển và nhận thông điệp của Sender và Receiver:

- Cơ chế chuyển (Sender):

Khi có một nguyên nhân, kích thích làm thay đổi một sự kiện nào đó, bên Sender sẽ gửi một thông điệp (message) cho bên nhận. Thông điệp ấy được nằm trong bộ nhớ tạm, nếu có tín hiệu chuyển phát (commit) mới thì thông điệp mới được chuyển sang cho người nhận.

- Cơ chế nhận (Receiver):

Khi nhận được một thông điệp được gửi đến. Nếu chưa kịp xử lý thì bên nhận đưa tạm vào bộ nhớ đệm, nếu bộ nhớ đầy thì báo lỗi và sử dụng cơ chế rollback gửi lại cho người gửi. Trong quá trình xử lý gặp lỗi thì phía bên nhận cũng phải báo lại cho bên gửi

Câu hỏi 4: So sánh JMS và DDS.

Điểm giống:

Đặc điểm	JMS – Java Messaging	DDS – Data distributed service
Cấu trúc	Publish / Subscribe	Publish / Subscribe (đa hướng)
Nền tảng độc lập	Cùng một API được hiển thị cho tất cả HW, OS và các ngôn ngữ được hỗ trợ	Cùng một API được hiển thị cho tất cả HW, OS và các ngôn ngữ được hỗ trợ

Điểm khác:

Đặc điểm	JMS – Java Messaging	DDS – Data distributed service
Khám phá các điểm cuối	Máy chủ JNDI và JMS phải được chỉ định và cấu hình	Khám phá động, không cần chỉ định nơi các điểm cuối cư trú
Loại an toàn	Đối tượng Tổng quát (Generic) và XML không phải là loại an toàn	An toàn kiểu cách mạnh mẽ, các cuộc gọi ứng dụng write () và read () với một kiểu dữ liệu cụ thể
Điều chỉnh hành vi giao tiếp	Khả năng điều chỉnh thông tin liên lạc hạn chế	Các chính sách QoS cho phép dễ dàng điều chỉnh các hành vi ứng xử giao tiếp
Khả năng tương tác	Không có	Tiêu chuẩn mở với khả năng tương tác đã được chứng minh