CI-CD README

1. 简介

- 本项目是一个基于Spring Boot的仓库管理系统，实现了用户注册、登录、货物分类、货物管理、出入库单管理等功能。
- 本文档是CI/CD实验的README文档，主要介绍了CI/CD的实现过程。
- 本文档共分为以下这几个部分
  - 手动部署说明（同README）
  - CI/CD工作流
  - 执行结果

2. 本项目部署说明

- 手动部署教程（不使用自动化部署，同README对应部分）
- Docker version: `>= 25.0.2`
- Please configure your **Docker Source** before deployment!
  - you can configure it through `/etc/docker/daemon.json` in Linux.
- Deploy

```
git clone git@github.com:YXHXianYu/OP-Abstract-Warehouse-Management-System.git
cd OP-Abstract-Warehouse-Management-System
docker-compose up
```

- Rebuild without cache

```
docker-compose build --no-cache
docker-compose up
```

3. CI/CD工作流

- 本项目使用以下技术实现CI/CD
  - 后端构建：Maven
  - 前端构建：npm
  - 自动化部署：Github Actions
  - 化简构建流程：Docker
  - 自动化测试：JUnit
- 流程概述
  - 编写测试用例
  - 配置 Github Actions

    - 配置 Docker
- 流程
    - Test Case
        - 在后端代码中编写测试用例，使用JUnit框架进行测试
    - Github Actions
        - 当代码提交到Github仓库的main分支时，Github Actions会自动触发CI/CD流程
        - Github Actions配置了云服务器的SSH密钥，可以自动登录到云服务器
        - Github Actions登陆腾讯云服务器，执行Docker构建与部署
        - Github Actions deploy.yml脚本代码

```yaml
name: Deploy to Aliyun Docker

on:
  push:
    branches:
      - main

jobs:
  deploy:
    runs-on: ubuntu-22.04
    steps:
    - name: Setup SSH connection
      uses: appleboy/ssh-action@master
      with:
        host: ${{ secrets.ALIYUN_SERVER_IP }}
        username: ${{ secrets.ALIYUN_SERVER_USERNAME }}
        key: ${{ secrets.ALIYUN_SERVER_SSH_PRIVATE_KEY }}
        script: |
          cd ~/OP-Abstract-Warehouse-Management-System
          git pull origin main
          docker-compose down
          docker-compose build
          docker-compose up -d
```

    - Docker
        - Docker根据docker-compose.yml文件，分别构建前端与后端的Docker镜像
        - 前端和后端分别根据对应的Dockerfile文件构建Docker镜像
        - 待前后端Docker镜像构建完成后，Docker根据docker-compose.yml文件启动容器
        - docker-compose.yml文件

```yaml
version: '3.8'
services:
  backend:
    build: ./backend
    ports:
      - "8080:8080"
    environment:
      - SPRING_DATASOURCE_URL=jdbc:mysql://121.40.90.193:3306/prod_database
      - SPRING_DATASOURCE_USERNAME=root
      - SPRING_DATASOURCE_PASSWORD=MySQL_2024JACKY

  frontend:
    build: ./frontend
    ports:
      - "5173:5173"
```

- Backend Dockerfile

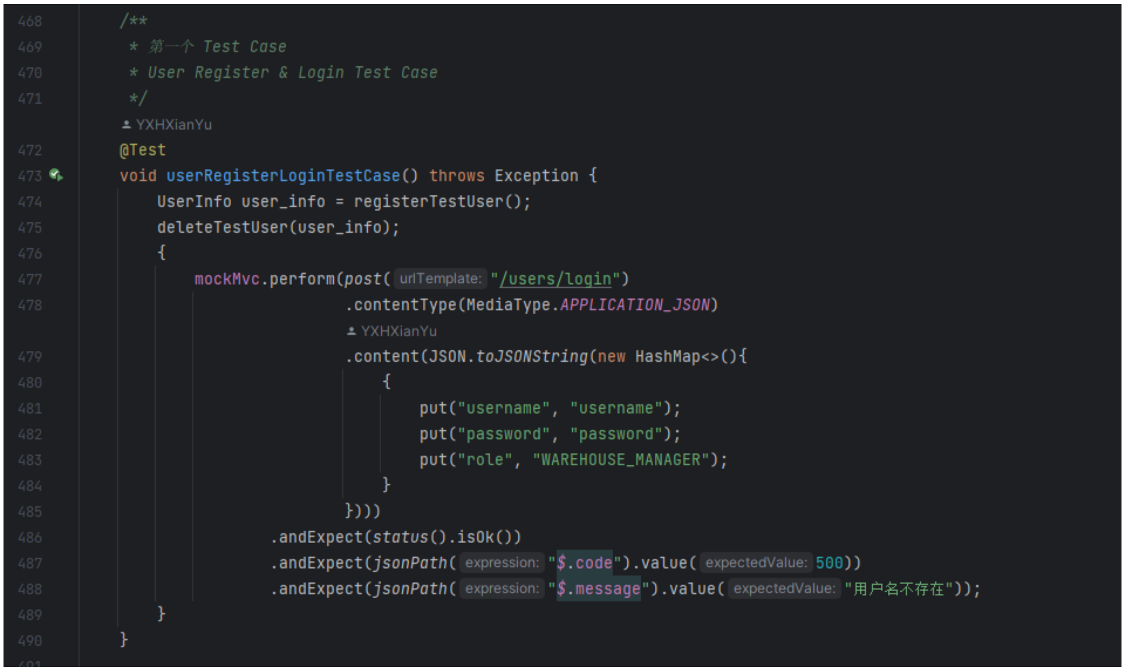
```
# 使用 OpenJDK 为基础镜像
FROM maven:3.8.5-openjdk-17

# 设置工作目录
WORKDIR /app

# 将 Maven 项目复制到容器中
COPY . ./

# 使用 Maven 构建应用
RUN mvn -s ./settings.xml clean package -X

# 指定运行时的端口
EXPOSE 8080

# 运行 SpringBoot 应用
CMD ["java", "-jar", "target/backend-0.0.1-SNAPSHOT.jar"]
```

- Frontend Dockerfile

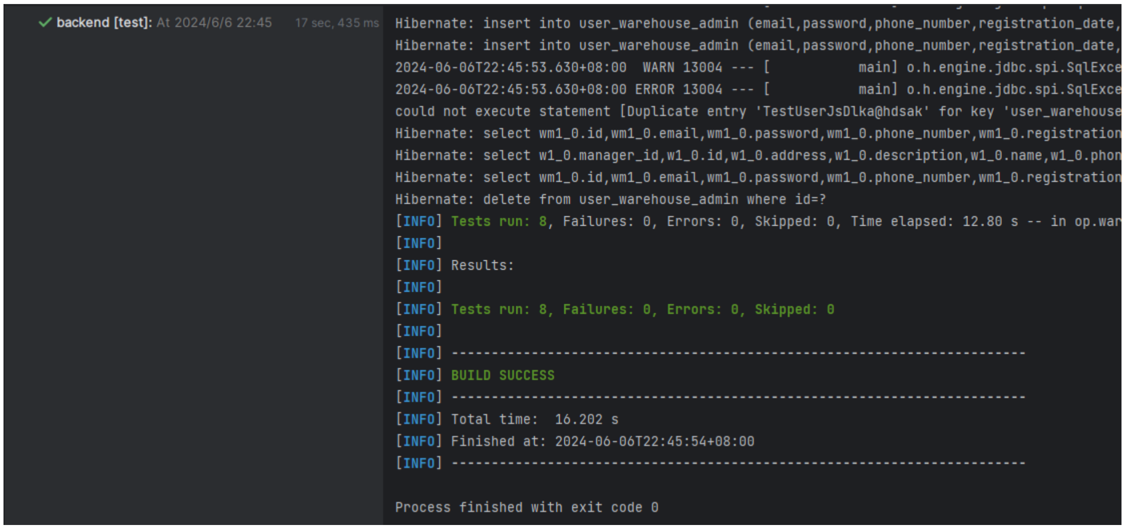
```
# 构建阶段
FROM node:18

WORKDIR /app

# 安装依赖
COPY package*.json ./
RUN npm install

# 构建应用
COPY ./ ./

# 指定运行时的端口
EXPOSE 5173

CMD ["npm", "run", "dev"]
```

4. 执行效果

4.1 测试用例

- 测试用例部分代码
  - 

```java
/**
 * 第一个 Test Case
 * User Register & Login Test Case
 */
YXHXianYu
@Test
void userRegisterLoginTestCase() throws Exception {
    UserInfo user_info = registerTestUser();
    deleteTestUser(user_info);
    {
        mockMvc.perform(post( urlTemplate: "/users/login")
                        .contentType(MediaType.APPLICATION_JSON)
                        YXHXianYu
                        .content(JSON.toJSONString(new HashMap<>(){
                            {
                                put("username", "username");
                                put("password", "password");
                                put("role", "WAREHOUSE_MANAGER");
                            }
                        }))
                .andExpect(status().isOk())
                .andExpect(jsonPath( expression: "$.code").value( expectedValue: 500))
                .andExpect(jsonPath( expression: "$.message").value( expectedValue: "用户名不存在"));
    }
}
```

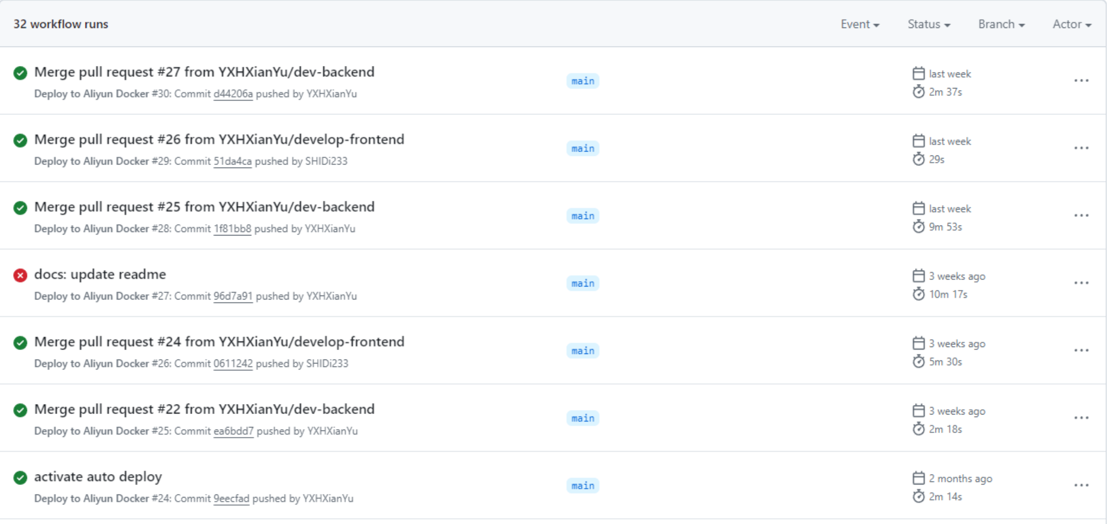
- 测试用例结果
  - 

✓ backend [test]: At 2024/6/6 22:45 17 sec, 435 ms

```
Hibernate: insert into user_warehouse_admin (email,password,phone_number,registration_date,
Hibernate: insert into user_warehouse_admin (email,password,phone_number,registration_date,
2024-06-06T22:45:53.630+08:00  WARN 13004 --- [           main] o.h.engine.jdbc.spi.SqlExce
2024-06-06T22:45:53.630+08:00 ERROR 13004 --- [           main] o.h.engine.jdbc.spi.SqlExce
could not execute statement [Duplicate entry 'TestUserJsDlka@hdsak' for key 'user_warehouse
Hibernate: select wm1_0.id,wm1_0.email,wm1_0.password,wm1_0.phone_number,wm1_0.registration
Hibernate: select w1_0.manager_id,w1_0.id,w1_0.address,w1_0.description,w1_0.name,w1_0.phon
Hibernate: select wm1_0.id,wm1_0.email,wm1_0.password,wm1_0.phone_number,wm1_0.registration
Hibernate: delete from user_warehouse_admin where id=?
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 12.80 s -- in op.war
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  16.202 s
[INFO] Finished at: 2024-06-06T22:45:54+08:00
[INFO] ------------------------------------------------------------------------

Process finished with exit code 0
```

4.2 Github Actions

- 部署代码见上文或此链接：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/blob/main/.github/workflows/deploy.yml
- 部分部署历史记录
  - 链接：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/actions
  - 注：本文档完成后，其他小组成员会对项目进行修改，所以本文档的截图并非提交时最新的部署结果。
  -



32 workflow runs

✓ Merge pull request #27 from YXHXianYu/dev-backend
Deploy to Aliyun Docker #30: Commit d44206a pushed by YXHXianYu
main
last week
2m 37s

✓ Merge pull request #26 from YXHXianYu/develop-frontend
Deploy to Aliyun Docker #29: Commit 51da4ca pushed by SHIDi233
main
last week
29s

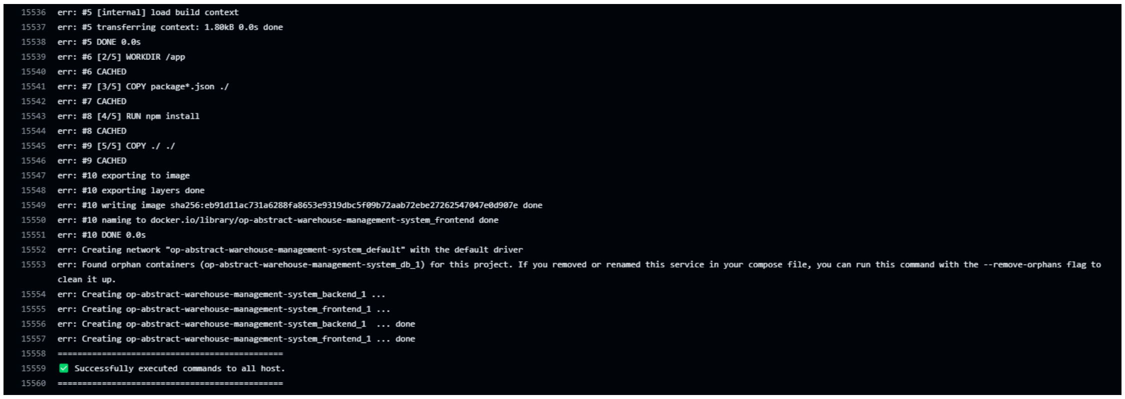
✓ Merge pull request #25 from YXHXianYu/dev-backend
Deploy to Aliyun Docker #28: Commit 1f81bb8 pushed by YXHXianYu
main
last week
9m 53s

✗ docs: update readme
Deploy to Aliyun Docker #27: Commit 96d7a91 pushed by YXHXianYu
main
3 weeks ago
10m 17s

✓ Merge pull request #24 from YXHXianYu/develop-frontend
Deploy to Aliyun Docker #26: Commit 0611242 pushed by SHIDi233
main
3 weeks ago
5m 30s

✓ Merge pull request #22 from YXHXianYu/dev-backend
Deploy to Aliyun Docker #25: Commit ea6bdd7 pushed by YXHXianYu
main
3 weeks ago
2m 18s

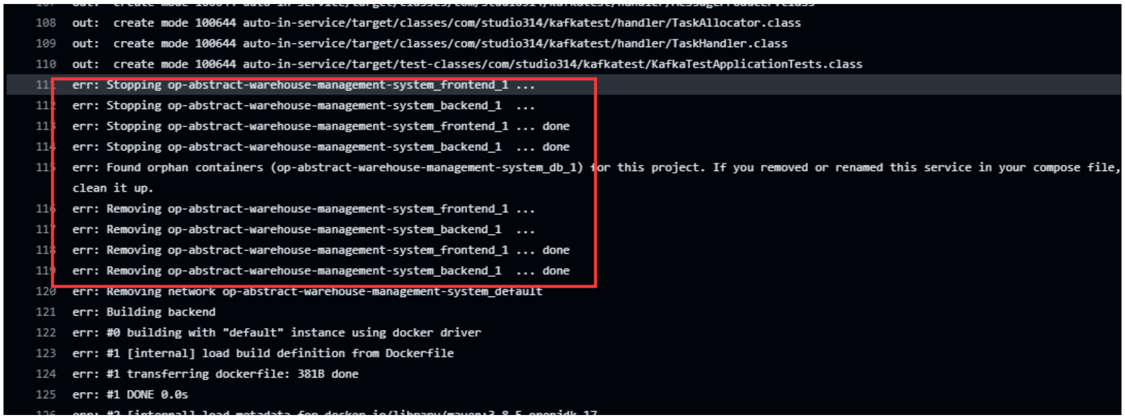
✓ activate auto deploy
Deploy to Aliyun Docker #24: Commit 9eecfad pushed by YXHXianYu
main
2 months ago
2m 14s

- `Merge pull request #27 from YXHXianYu/dev-backend`的部分详细信息
  - 链接：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/actions/runs/9263900374/job/25483099559
  - 

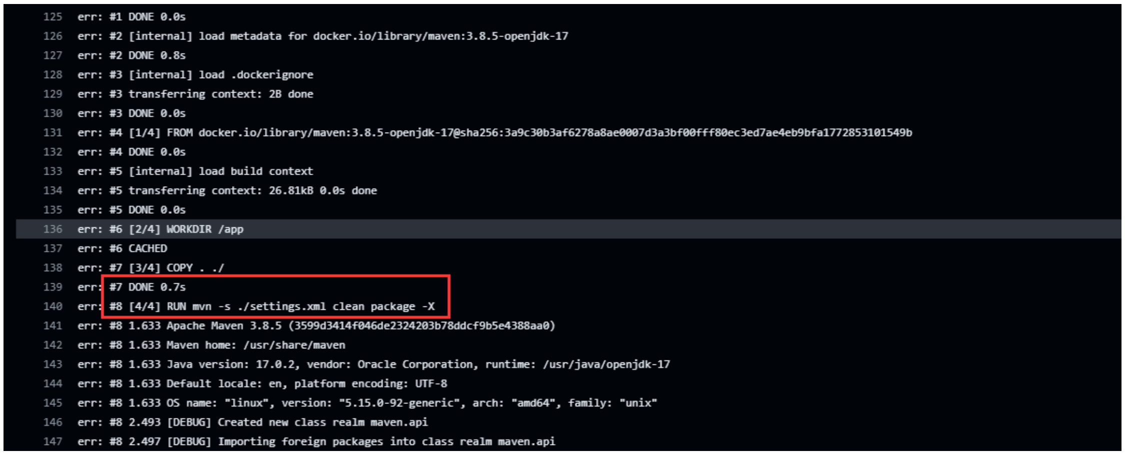
```
err: #5 [internal] load build context
err: #5 transferring context: 1.80kB 0.0s done
err: #5 DONE 0.0s
err: #6 [2/5] WORKDIR /app
err: #6 CACHED
err: #7 [3/5] COPY package*.json ./
err: #7 CACHED
err: #8 [4/5] RUN npm install
err: #8 CACHED
err: #9 [5/5] COPY ./ ./
err: #9 CACHED
err: #10 exporting to image
err: #10 exporting layers done
err: #10 writing image sha256:eb91d11ac731a6288fa8653e9319dbc5f09b72aab72ebe27262547047e0d907e done
err: #10 naming to docker.io/library/op-abstract-warehouse-management-system_frontend done
err: #10 DONE 0.0s
err: Creating network "op-abstract-warehouse-management-system_default" with the default driver
err: Found orphan containers (op-abstract-warehouse-management-system_db_1) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to
clean it up.
err: Creating op-abstract-warehouse-management-system_backend_1 ...
err: Creating op-abstract-warehouse-management-system_frontend_1 ...
err: Creating op-abstract-warehouse-management-system_backend_1  ... done
err: Creating op-abstract-warehouse-management-system_frontend_1 ... done
============================================
✅ Successfully executed commands to all host.
============================================
```

4.3 Docker

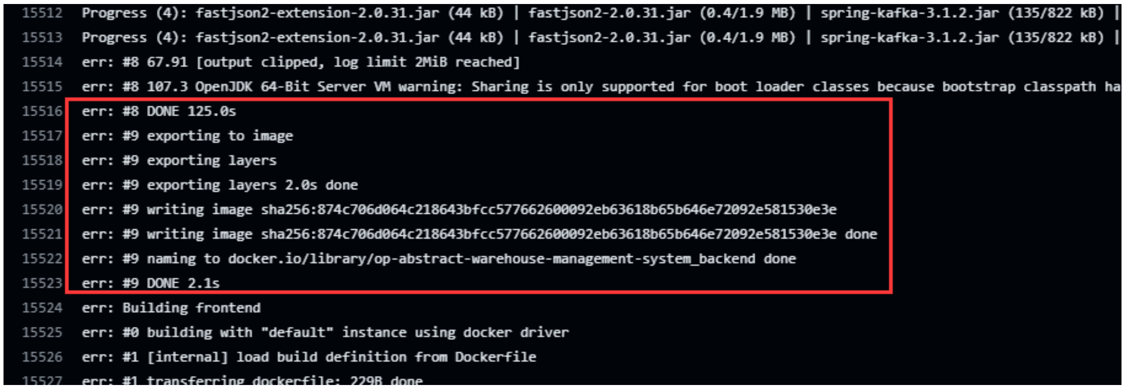
- 代码见上文，或如下链接
  - docker-compose.yml：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/blob/main/docker-compose.yml
  - 后端dockerfile：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/blob/main/backend/Dockerfile
  - 前端dockerfile：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/blob/main/frontend/Dockerfile
- 参考链接：https://github.com/YXHXianYu/OP-Abstract-Warehouse-Management-System/actions/runs/9263900374/job/25483099559
- 重置容器信息
  -



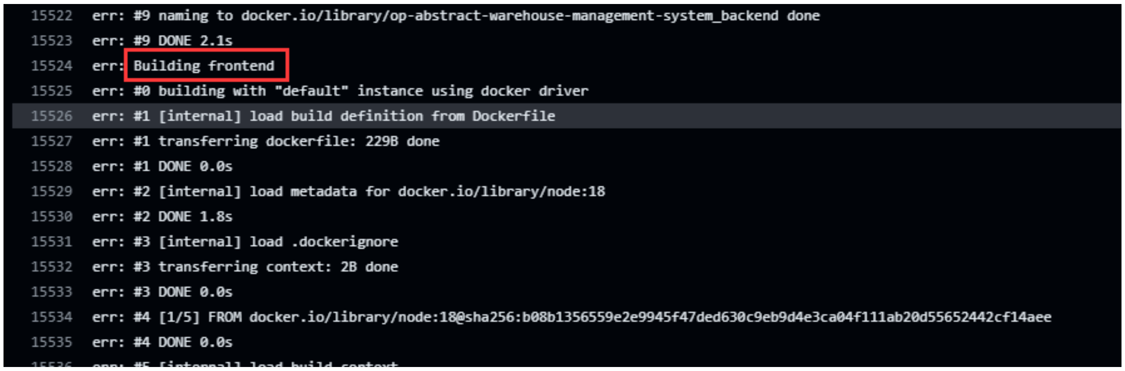
```
out:  create mode 100644 auto-in-service/target/classes/com/studio314/kafkatest/handler/TaskAllocator.class
out:  create mode 100644 auto-in-service/target/classes/com/studio314/kafkatest/handler/TaskHandler.class
out:  create mode 100644 auto-in-service/target/test-classes/com/studio314/kafkatest/KafkaTestApplicationTests.class
err: Stopping op-abstract-warehouse-management-system_frontend_1 ...
err: Stopping op-abstract-warehouse-management-system_backend_1  ...
err: Stopping op-abstract-warehouse-management-system_frontend_1 ... done
err: Stopping op-abstract-warehouse-management-system_backend_1  ... done
err: Found orphan containers (op-abstract-warehouse-management-system_db_1) for this project. If you removed or renamed this service in your compose file,
clean it up.
err: Removing op-abstract-warehouse-management-system_frontend_1 ...
err: Removing op-abstract-warehouse-management-system_backend_1  ...
err: Removing op-abstract-warehouse-management-system_frontend_1 ... done
err: Removing op-abstract-warehouse-management-system_backend_1  ... done
err: Removing network op-abstract-warehouse-management-system_default
err: Building backend
err: #0 building with "default" instance using docker driver
err: #1 [internal] load build definition from Dockerfile
err: #1 transferring dockerfile: 381B done
err: #1 DONE 0.0s
err: #2 [internal] load metadata for docker.io/library/maven:3.8.5-openjdk-17
```

- 后端部署
  - 

```
err: #1 DONE 0.0s
err: #2 [internal] load metadata for docker.io/library/maven:3.8.5-openjdk-17
err: #2 DONE 0.8s
err: #3 [internal] load .dockerignore
err: #3 transferring context: 2B done
err: #3 DONE 0.0s
err: #4 [1/4] FROM docker.io/library/maven:3.8.5-openjdk-17@sha256:3a9c30b3af6278a8ae0007d3a3bf00fff80ec3ed7ae4eb9bfa1772853101549b
err: #4 DONE 0.0s
err: #5 [internal] load build context
err: #5 transferring context: 26.81kB 0.0s done
err: #5 DONE 0.0s
err: #6 [2/4] WORKDIR /app
err: #6 CACHED
err: #7 [3/4] COPY . ./
err: #7 DONE 0.7s
err: #8 [4/4] RUN mvn -s ./settings.xml clean package -X
err: #8 1.633 Apache Maven 3.8.5 (3599d3414f046de2324203b78ddcf9b5e4388aa0)
err: #8 1.633 Maven home: /usr/share/maven
err: #8 1.633 Java version: 17.0.2, vendor: Oracle Corporation, runtime: /usr/java/openjdk-17
err: #8 1.633 Default locale: en, platform encoding: UTF-8
err: #8 1.633 OS name: "linux", version: "5.15.0-92-generic", arch: "amd64", family: "unix"
err: #8 2.493 [DEBUG] Created new class realm maven.api
err: #8 2.497 [DEBUG] Importing foreign packages into class realm maven.api
```

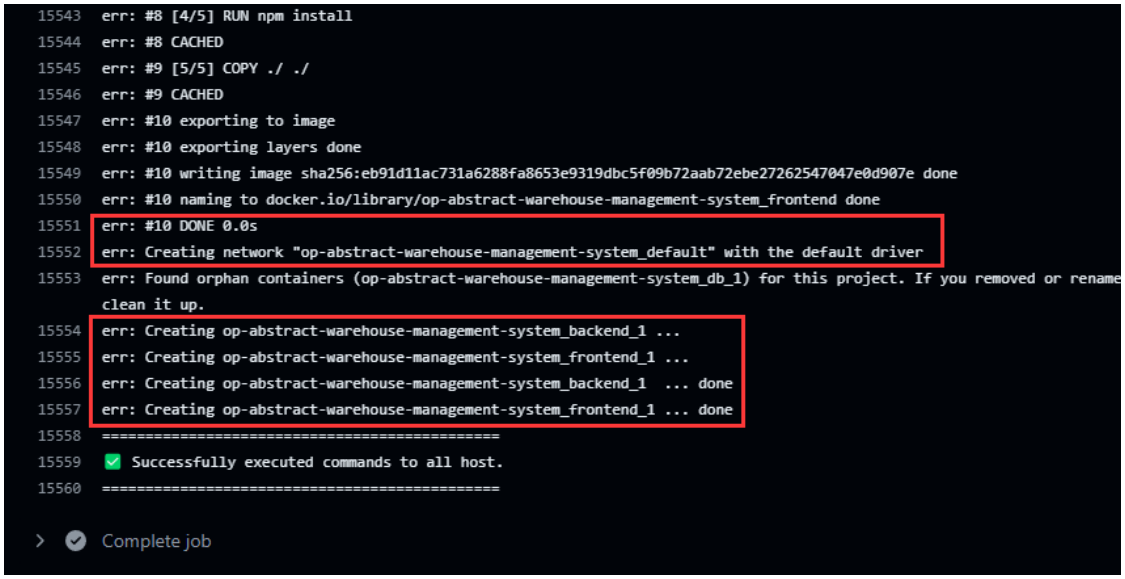
- 后端部署完成
  - 

```
Progress (4): fastjson2-extension-2.0.31.jar (44 kB) | fastjson2-2.0.31.jar (0.4/1.9 MB) | spring-kafka-3.1.2.jar (135/822 kB) |
Progress (4): fastjson2-extension-2.0.31.jar (44 kB) | fastjson2-2.0.31.jar (0.4/1.9 MB) | spring-kafka-3.1.2.jar (135/822 kB) |
err: #8 67.91 [output clipped, log limit 2MiB reached]
err: #8 107.3 OpenJDK 64-Bit Server VM warning: Sharing is only supported for boot loader classes because bootstrap classpath ha
err: #8 DONE 125.0s
err: #9 exporting to image
err: #9 exporting layers
err: #9 exporting layers 2.0s done
err: #9 writing image sha256:874c706d064c218643bfcc577662600092eb63618b65b646e72092e581530e3e
err: #9 writing image sha256:874c706d064c218643bfcc577662600092eb63618b65b646e72092e581530e3e done
err: #9 naming to docker.io/library/op-abstract-warehouse-management-system_backend done
err: #9 DONE 2.1s
err: Building frontend
err: #0 building with "default" instance using docker driver
err: #1 [internal] load build definition from Dockerfile
err: #1 transferring dockerfile: 229B done
```

- 前端部署
  - 

```
err: #9 naming to docker.io/library/op-abstract-warehouse-management-system_backend done
err: #9 DONE 2.1s
err: Building frontend
err: #0 building with "default" instance using docker driver
err: #1 [internal] load build definition from Dockerfile
err: #1 transferring dockerfile: 229B done
err: #1 DONE 0.0s
err: #2 [internal] load metadata for docker.io/library/node:18
err: #2 DONE 1.8s
err: #3 [internal] load .dockerignore
err: #3 transferring context: 2B done
err: #3 DONE 0.0s
err: #4 [1/5] FROM docker.io/library/node:18@sha256:b08b1356559e2e9945f47ded630c9eb9d4e3ca04f111ab20d55652442cf14aee
err: #4 DONE 0.0s
err: #5 [internal] load build context
```

- 前端部署完成
  -



```
err: #8 [4/5] RUN npm install
err: #8 CACHED
err: #9 [5/5] COPY ./ ./
err: #9 CACHED
err: #10 exporting to image
err: #10 exporting layers done
err: #10 writing image sha256:eb91d11ac731a6288fa8653e9319dbc5f09b72aab72ebe27262547047e0d907e done
err: #10 naming to docker.io/library/op-abstract-warehouse-management-system_frontend done
err: #10 DONE 0.0s
err: Creating network "op-abstract-warehouse-management-system_default" with the default driver
err: Found orphan containers (op-abstract-warehouse-management-system_db_1) for this project. If you removed or rename
clean it up.
err: Creating op-abstract-warehouse-management-system_backend_1 ...
err: Creating op-abstract-warehouse-management-system_frontend_1 ...
err: Creating op-abstract-warehouse-management-system_backend_1  ... done
err: Creating op-abstract-warehouse-management-system_frontend_1 ... done
==============================================
✅ Successfully executed commands to all host.
==============================================
```