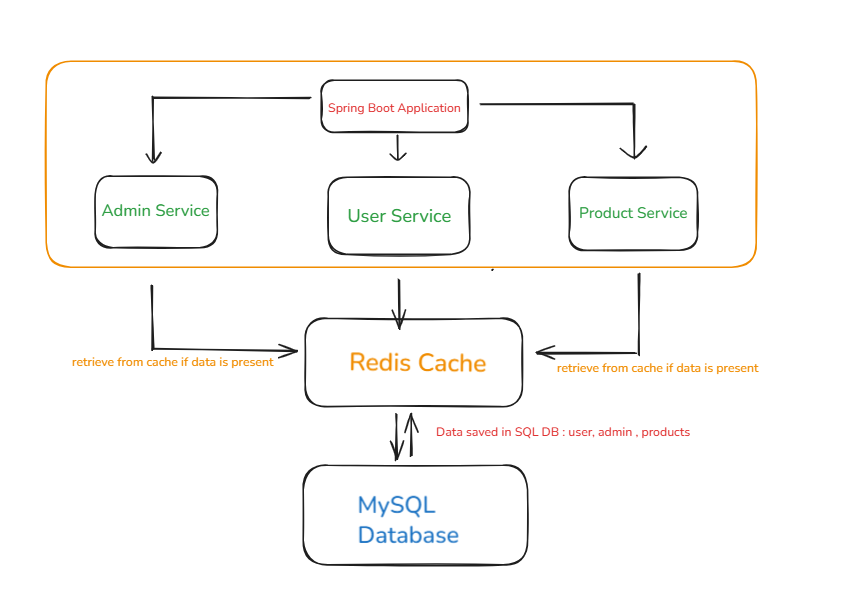
**Microservice Architecture Beauty App Project using Spring Boot, Redis and MySQL**

This document describes a proposed microservice architecture for your BeautyApp project, leveraging Redis (as a cache) and MySQL (as the primary data store).  
  
  


---## 1. Overview

# BeautyApp Spring Boot Application: Docker Compose Deployment Guide

This guide explains how to deploy your [pranjaljoshinagarro/beautyappproject] (<https://github.com/pranjaljoshinagarro/beautyappproject>)

Spring Boot application using the provided `docker-compose.yml`. The architecture includes your main app, MySQL as the relational database, and Redis as the caching layer.

\*\*Key Components:\*\*

- User Service

- Admin Service

- Product Service

- MySQL (Database)

- Redis (Cache)

## Technology Stack

| Component | Technology |

|--------------------|-------------------|

| Microservices | Spring Boot |

| Cache | Redis |

| Database | MySQL |

| Containerization | Docker, Docker Compose/Kubernetes |  
  
  
  
## Prerequisites

- [Docker](https://docs.docker.com/get-docker/) and [Docker Compose](https://docs.docker.com/compose/) installed.

- Your project folder should contain:

- `docker-compose.yml` (as shown above)

- `init.sql` (if you want to pre-populate your MySQL database)

- (Optional) Access to the Docker registry for the prebuilt images:

- `pranjalnagarro/beautyappprojectmain:beautyappprojectapplication` (app)

- `pranjalnagarro/beautyappproject:beautyappredis` (redis)  
  
  
  
Clone the repository:   
git clone <https://github.com/pranjaljoshinagarro/beautyappproject.git>

cd beautyappproject/

## 1. Start All Services

Build and run all services in detached mode:

```sh

docker compose up --build -d

```

- The `--build` flag is only required if you use a local Dockerfile; as your `app` service uses a prebuilt image, it is optional.

- The `-d` flag runs the containers in the background.  
  
  
  
### What happens?

- \*\*App service\*\* runs your Spring Boot application.

- \*\*db\*\* service runs MySQL 8.0, with the database `beautyproject`.

- \*\*redis\*\* service runs Redis for caching.  
  
  
## 2. Verify Services

Check if all services are running:

```sh

docker compose ps

```

You should see containers named `beautyapp`, `beautyapp-mysql-db`, and `beautyapp-redis` all in the "Up" state.

## 3. Access the Application

- Open [http://localhost:8080](http://localhost:8080) in your browser.

- The app should now be live and connected to both MySQL and Redis.

## 4. Stopping and Removing Containers

To stop and remove all containers, networks, and the default network:

```sh

docker compose down

```  
  
##5. Useful Commands to execute in container   
- \*\*Connect to the MySQL CLI:\*\*

docker exec -it beautyapp-mysql-db mysql -u root -p

- \*\*Connect to the Redis CLI:\*\*

docker exec -it beautyapp-redis redis-cli

## 6. Troubleshooting

- \*\*App fails to connect to db or redis:\*\*

Ensure the `db` and `redis` services are healthy (`docker compose ps`).

- \*\*Templates not found:\*\*

Ensure your application JAR includes all Thymeleaf template files.

- \*\*MySQL schema doesn't exist:\*\*

Check your `init.sql`, or manually create the schema.

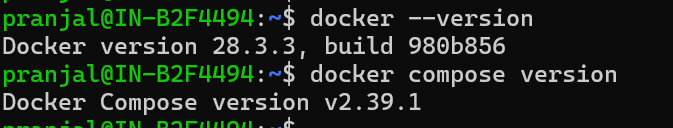
## 9. Reference

- [Sample Aplication Link:]

(https://github.com/SnehalAShinde/Spring\_Boot\_Project/tree/master)

- [Docker Compose documentation](https://docs.docker.com/compose/)

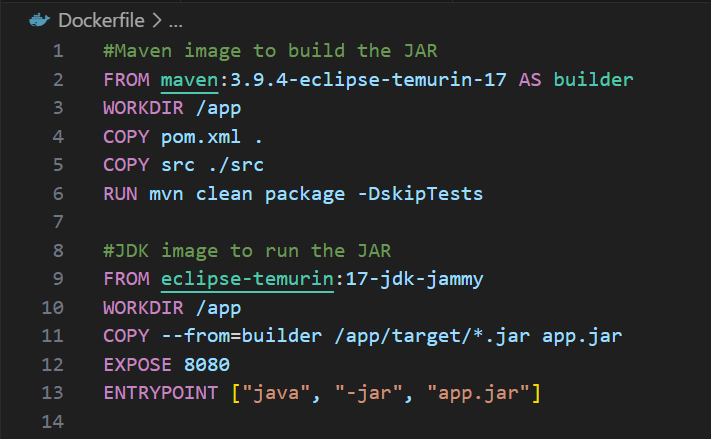
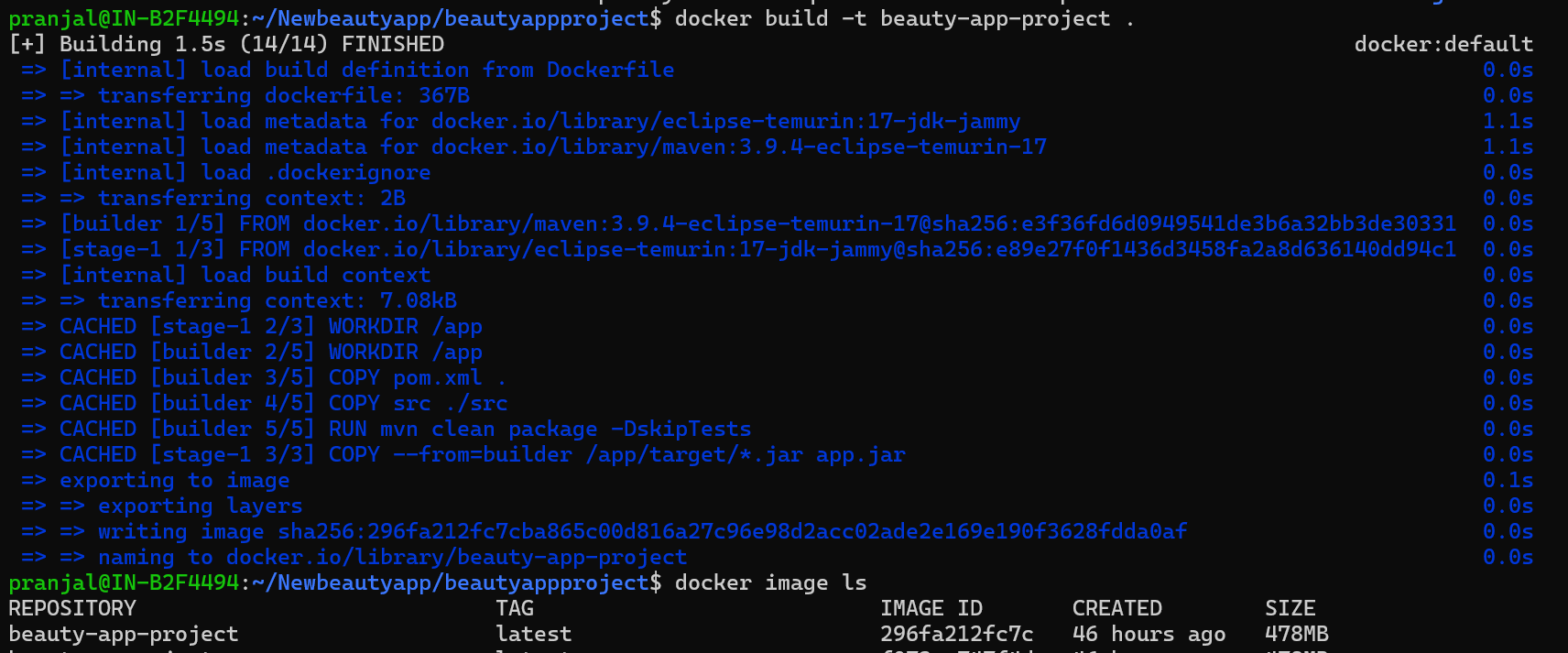
- [Spring Boot Docker guide](https://spring.io/guides/gs/spring-boot-docker/)

**Basic Assignment:  
  
• Install Docker and Docker Compose on your machine.  
  
  
  
  
• Verify the installation by running a test container.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**• Create a simple web application (e.g., a Python Flask, DotNet, Java or Node application).  
  
Github Repo Link:** [**https://github.com/pranjaljoshinagarro/beautyappproject.git**](https://github.com/pranjaljoshinagarro/beautyappproject.git) **• Write a Dockerfile to containerize the application.**

**  
  
  
• Build the Docker image and run a container from it  
  
  
  
  
• Use Docker commands to list, start, stop, and remove containers.  
A screenshot of a computer

AI-generated content may be incorrect.**

**• Inspect running containers and view logs.  
  
  
  
A screenshot of a computer program

AI-generated content may be incorrect.  
  
A screen shot of a computer program

AI-generated content may be incorrect.**

• Write a docker-compose.yml file to define a multi-container application.  
  
A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer program

AI-generated content may be incorrect.  
A screenshot of a computer program

AI-generated content may be incorrect.  
  
  
• Use Docker Compose to bring up the application and ensure all services are running correctly  
  
A screen shot of a computer program

AI-generated content may be incorrect.  
  
  
• Create a private Docker registry or use Docker Hub.

A screenshot of a computer

AI-generated content may be incorrect.

• Push your Docker images to the registry.  
  
A screen shot of a computer

AI-generated content may be incorrect.  
  
• Pull the images from the registry and run them locally.

A screenshot of a computer screen

AI-generated content may be incorrect.  
  
  
  
Applications Related Screenshots:   
A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.  
  
A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.  
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A computer screen shot of a black screen

AI-generated content may be incorrect.  
  
  
  
Thank you for reading.