Day33 MongoDB Use Case.

Create a Spring Boot Micro-service using MongoDB for ARS application.

Microservice vs Monolith.

ARS-v2 – Single Monolith application which contains all the modules in a Single Spring Boot project.

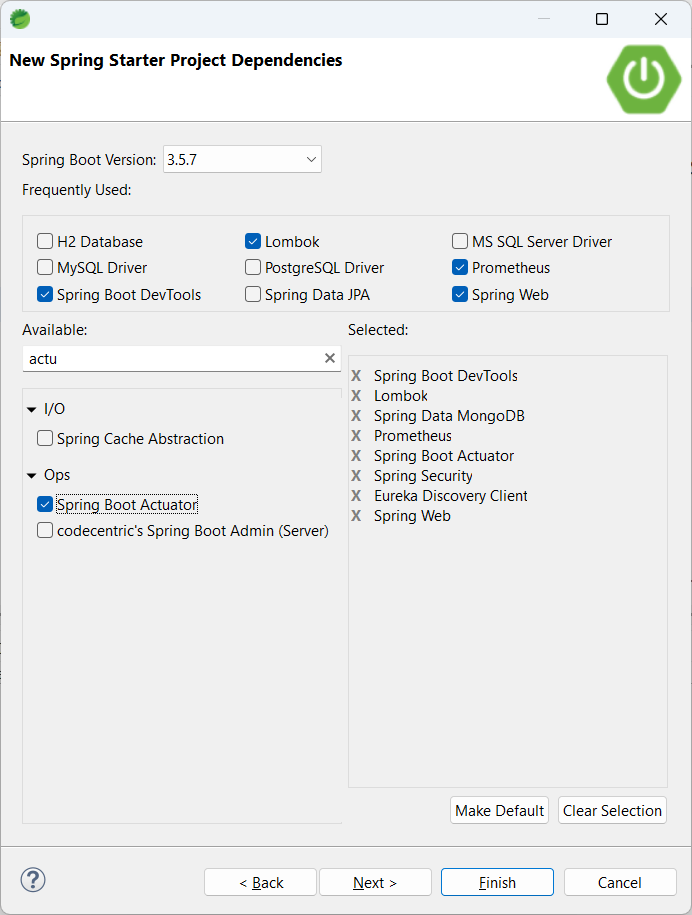
ARS-v3 – Microservice Architecture (Discovery Server, API Gateway, Config Server, Other micro-service)

JDK

DB (MS-SQL/My-SQL, MongoDB)

IDE (STS/VS Code/Eclipse/Intellij)

Booking micro-service



DTO vs Model/Entity

DTO is mainly used for Data transfer purpose only. Not persisted data. (Used only for client to server or service to service communication)

Model content is persisted.

Added all the must have, should have and good to have packages.

Configured Security. (Using JWT)

1. Base package
2. B.P+controller
3. B.P+model (DAO – Data Access Layer – Data carrying layer)
4. B.P+repo
5. B.P+service
6. B.P+config
7. B.P+enums
8. B.P+util
9. B.P+exceptions
10. B.P+dto (Data Transfer Objects) –
11. B.P+filter

Configured Application.properties file

DB config

Prometheus, actuator config

Unit & Integration Test Codes.

Application with 5 Micro-service. Excluding Discovery, API gateway and config server.

Docker – Single Runnable Image using the source\_code+lib+runtime\_env

To create docker image, we need dockerfile and docker cli/docker desktop running locally.

“docker build -t booking-service .”

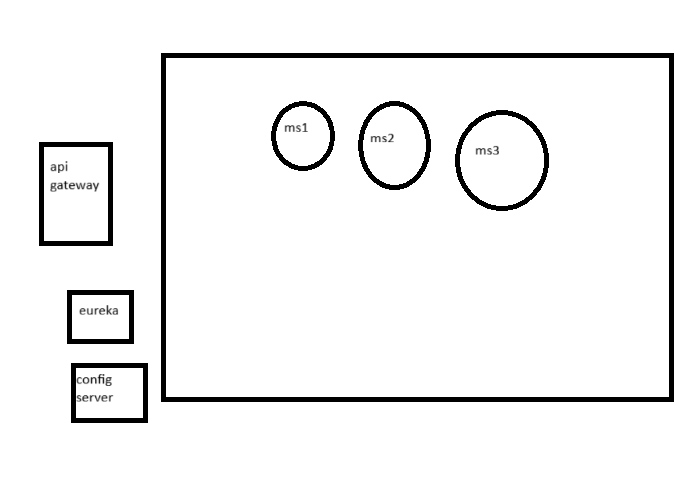
“docker login”

“docker tag booking-service <your\_dockerhub\_username>/booking-service”

“docker push <your\_dockerhub\_username>/booking-service”

<https://hub.docker.com/r/syskantechnosoft/booking-service/tags>

docker pull syskantechnosoft/booking-service:latest



Docker-compose -- Helps to run multiple micro-services in proper order.

“docker-compose” is a yml file.

“docker-compose up” -- docker-compose.yml

Ars-v3

-backend

– docker-compose.yml

* - user-service
* - booking-service
* - flight-service

-frontend

Github Actions

Jenkins - <https://github.com/jenkinsci/docker/blob/master/README.md>

<https://hub.docker.com/r/jenkins/jenkins>

docker run -d -v jenkins\_home:/var/jenkins\_home -p 8080:8080 -p 50000:50000 --restart=on-failure jenkins/jenkins:lts-jdk21

9d3013680b414aa88adef928197d3f8f