

## Quick review of SDTproject "etapa 4"

Team:

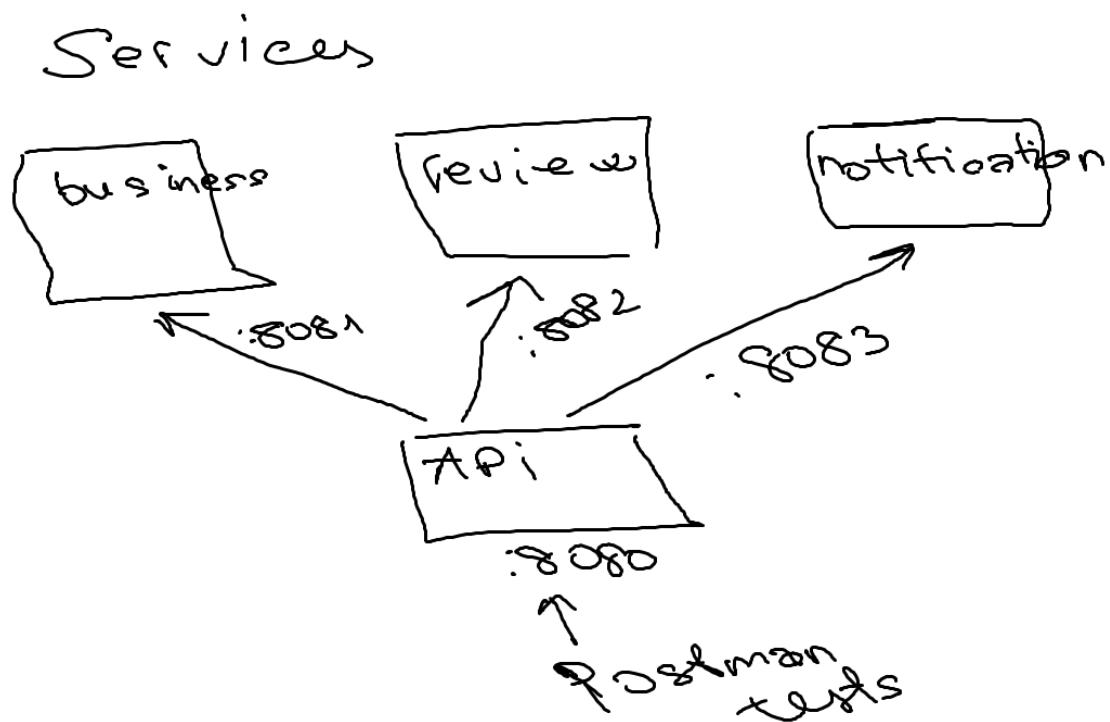
Chiorean Rebeca 1241EA sgr2

Lita Naomi 1241EA sgr2

Adascalitei Ana-Maria 1241EA sgr1

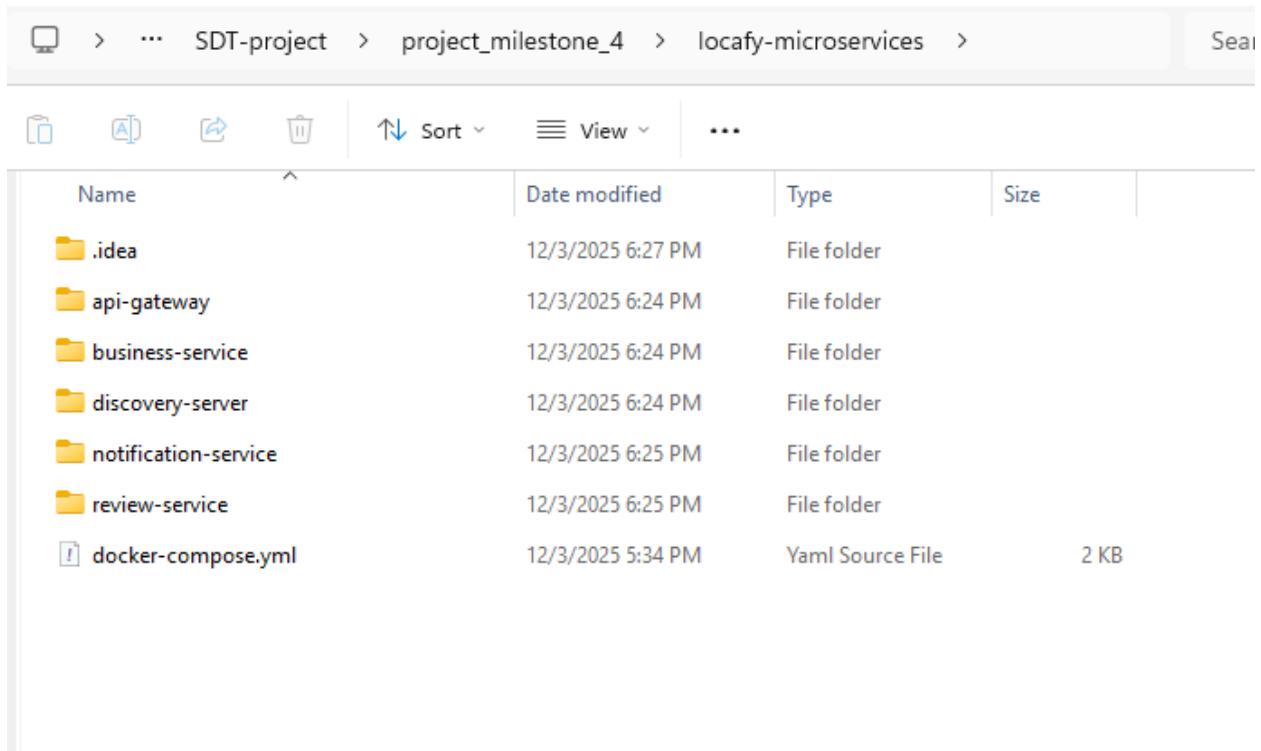
### The Project Structure:

- Spring Eureka discovery system (port 8071) -> discovers all available services
- API Gateway (port 8080) -> entrypoint for all other services



- The Services:
  - Business service (port 8081) -> has builder
  - Review service (port 8082)
  - Notification service (port 8083)

Everything is a new project created with Spring Boot(<https://start.spring.io/>):



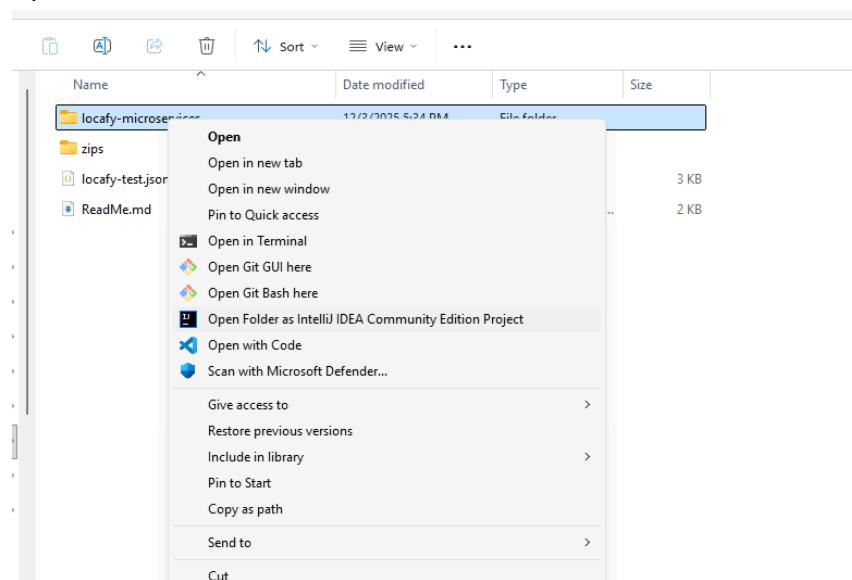
A screenshot of a Windows File Explorer window. The path is displayed at the top: Desktop > SDT-project > project\_milestone\_4 > locafy-microservices >. The main area shows a list of files and folders:

Name	Date modified	Type	Size
.idea	12/3/2025 6:27 PM	File folder	
api-gateway	12/3/2025 6:24 PM	File folder	
business-service	12/3/2025 6:24 PM	File folder	
discovery-server	12/3/2025 6:24 PM	File folder	
notification-service	12/3/2025 6:25 PM	File folder	
review-service	12/3/2025 6:25 PM	File folder	
docker-compose.yml	12/3/2025 5:34 PM	Yaml Source File	2 KB

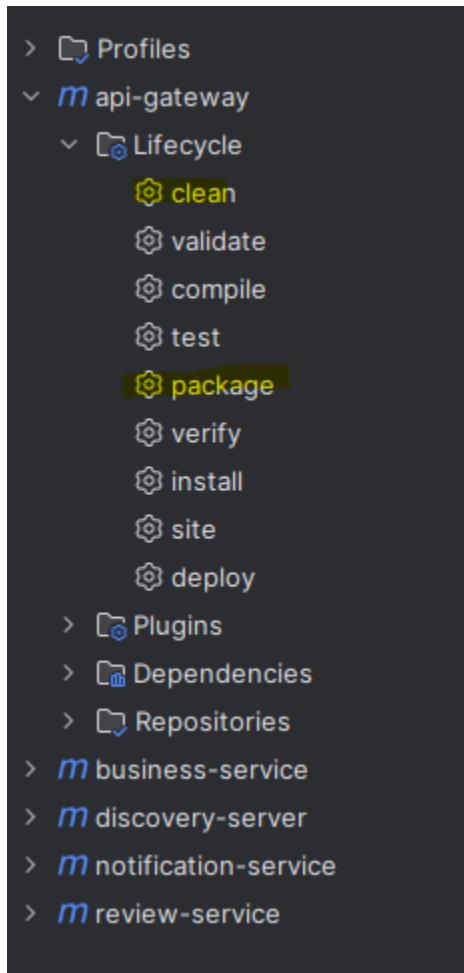
## Docker Containerization:

Step1:

- Open the main folder in IntelliJ



- Build all the projects with Maeven(use clean and then package)

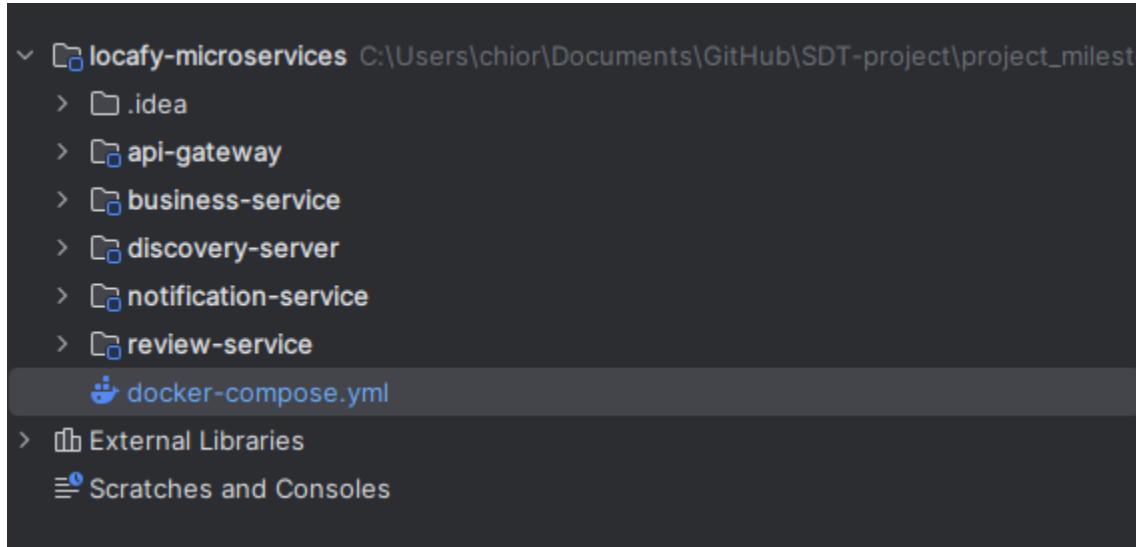


## Step 2:

- Have a “Docker” file inside each project (api-gateway, business-service, discovery-server, notification-service, review-service)



- Have a “docker-compose.yml” file



### Step 3:

- Open Administrator Prompt and move into the project directory:

```
PS C:\Users\chior> cd C:\Users\chior\Documents\GitHub\SDT-project\project_milestone_4\locafy-microservices
```

- Build everything:

```
locafy-microservices> docker compose up --build
```

- Admire:

locafy-microservices				
		review-service	2025-12-03T18:29:47.016Z	INFO 1 --- [review-service]
discover...	●	discovery-server	2025-12-03T18:30:44.146Z	INFO 1 --- [discovery-servi
locafy-micros	8761:8761		4ms	3ms
			2025-12-03T18:31:44.146Z	INFO 1 --- [discovery-servi
api-gate...	●	api-gateway	2025-12-03T18:32:44.147Z	INFO 1 --- [api-gateway] [
locafy-micros	8080:8080		4ms	4ms
			2025-12-03T18:33:44.147Z	INFO 1 --- [discovery-servi
notificat...	●		4ms	4ms
locafy-micros	8083:8083			
		api-gateway	2025-12-03T18:34:41.042Z	INFO 1 --- [api-gateway] [
busines...	●	discovery-server	2025-12-03T18:34:43.883Z	INFO 1 --- [discovery-servi
locafy-micros	8081:8081			configuration
		notification-service	2025-12-03T18:34:44.041Z	INFO 1 --- [notification-si
review-s...	●	discovery-server	2025-12-03T18:34:44.148Z	INFO 1 --- [discovery-servi
locafy-micros	8082:8082		13ms	13ms
		business-service	2025-12-03T18:34:46.583Z	INFO 1 --- [business-servi
		review-service	2025-12-03T18:34:46.988Z	INFO 1 --- [review-service]
		discovery-server	2025-12-03T18:35:44.149Z	INFO 1 --- [discovery-servi
			10ms	

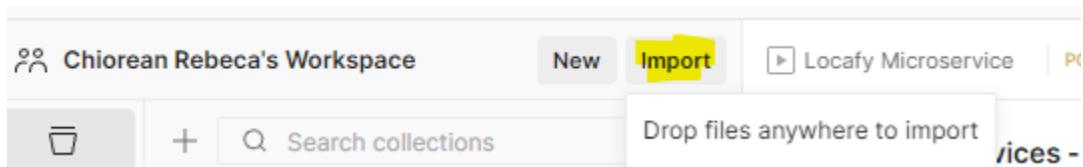
## Postman:

### Step1:

- Install the desktop application.

### Step 2:

- Import locafy-test.json



### Step 4:

- Run the tests

A screenshot of the Postman interface showing the "Locafy Microservices - Run results" section. It displays a list of test cases:

- POST 1. Create Business (Builder Pattern)
- POST 6. Create Review for Business that we triggered notifications for
- GET 2. Search Business (Name Strategy)
- POST 3. Create Review (Triggers Notifications)
- POST 4. Create Review for Non-Existent Business
- POST 5. Create Review for existing business

A context menu is open over the fourth item ("POST 4. Create Review for Non-Existent Business"). The menu options include "Add request", "Add folder", "Run" (which is highlighted with a yellow box), "Share", "Copy link", "Ask AI", "Move", "Fork", "Rename", "Duplicate", "Delete", and "More". On the right side of the screen, there is a summary table:

Environment	Iteration
One	1

Failed (0) Skipped (0)

Useful links (I suggest using these after Postman ran its first test suite so the db aren't empty):

- Checking business database entries based on ids:  
<http://localhost:8080/api/businesses/1>
- Checking reviews:  
<http://localhost:8080/api/reviews/business/1>

Looking for notifications (At the moment notifications are printed):

```
2025-12-03T18:22:14.900Z INFO 1 --- [notification-service] [nio-8083-exec-7] c.e.n.controller.NotificationController : -----
-----
2025-12-03T18:22:14.900Z INFO 1 --- [notification-service] [nio-8083-exec-7] c.e.n.controller.NotificationController : ⚠ NOTIFICATION RECEIVED: New Review
Posted for Business ID: 1
2025-12-03T18:22:14.901Z INFO 1 --- [notification-service] [nio-8083-exec-7] c.e.n.controller.NotificationController : -----
-----
```

← this notification appears after a review is being posted

**Files and their content:**

**Dockerfile (the same file in every folder):**

```
# 1. Start with Java 17
FROM eclipse-temurin:17-jdk-alpine

# 2. Set the working directory inside the container
WORKDIR /app

# 3. Copy the built JAR file from your computer's 'target' folder into the
#      container
#      (Maven builds into 'target', Gradle builds into 'build/libs')
COPY target/*.jar app.jar

# 4. Run the application
ENTRYPOINT ["java", "-jar", "app.jar"]
```

## docker-compose.yml

```
version: '3.8'

services:
  discovery-server:
    build: ./discovery-server
    container_name: discovery-server
    ports:
      - "8761:8761"
    environment:
      - SPRING_PROFILES_ACTIVE=docker

  api-gateway:
    build: ./api-gateway
    container_name: api-gateway
    ports:
      - "8080:8080"
    environment:
      -
      EUREKA_CLIENT_SERVICEURL_DEFAULTZONE=http://discovery-server:8761/eureka/
      - SPRING_PROFILES_ACTIVE=docker
    depends_on:
      - discovery-server

  business-service:
    build: ./business-service
    container_name: business-service
    ports:
      - "8081:8081"
    environment:
      -
      EUREKA_CLIENT_SERVICEURL_DEFAULTZONE=http://discovery-server:8761/eureka/
    depends_on:
      - discovery-server

  notification-service:
    build: ./notification-service
    container_name: notification-service
    ports:
      - "8083:8083"
    environment:
      -
      EUREKA_CLIENT_SERVICEURL_DEFAULTZONE=http://discovery-server:8761/eureka/
    depends_on:
      - discovery-server

  review-service:
    build: ./review-service
    container_name: review-service
```

```
ports:
  - "8082:8082"
environment:
  -
EUREKA_CLIENT_SERVICEURL_DEFAULTZONE=http://discovery-server:8761/eureka/
depends_on:
  - discovery-server
  - business-service
```