

The logo consists of a large red square. Inside this square is a smaller white square. Within the white square is a red square, and inside that is a white square. The word "iCommerce" is written in white, sans-serif font, centered within the innermost white square.

iCommerce

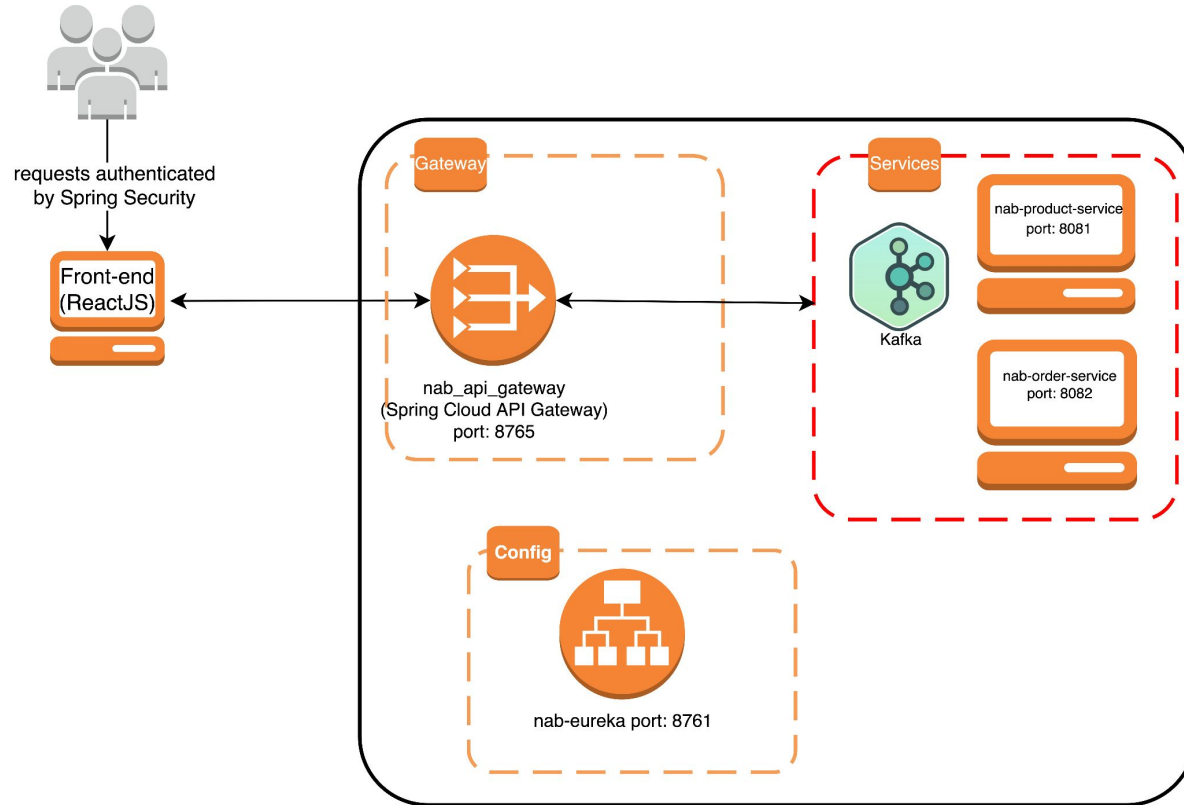
Installation

Follow steps in order

1. Install & run Kafka service with port: 9092 (you can follow this [link](#))
2. Install & run MySQL server.
3. Take file “nab-db.sql” in nab-product-service, and execute it in Mysql server. The step just initiates data.
4. Install & run nab-product-service (in README)
5. Install & run nab-order-service (in README)
6. Install & run nab-eureka (in README)
7. Install & run nab-api-gateway (in README)
8. Install & run nab-fe (in README)

Overview

Workflow



Technologies

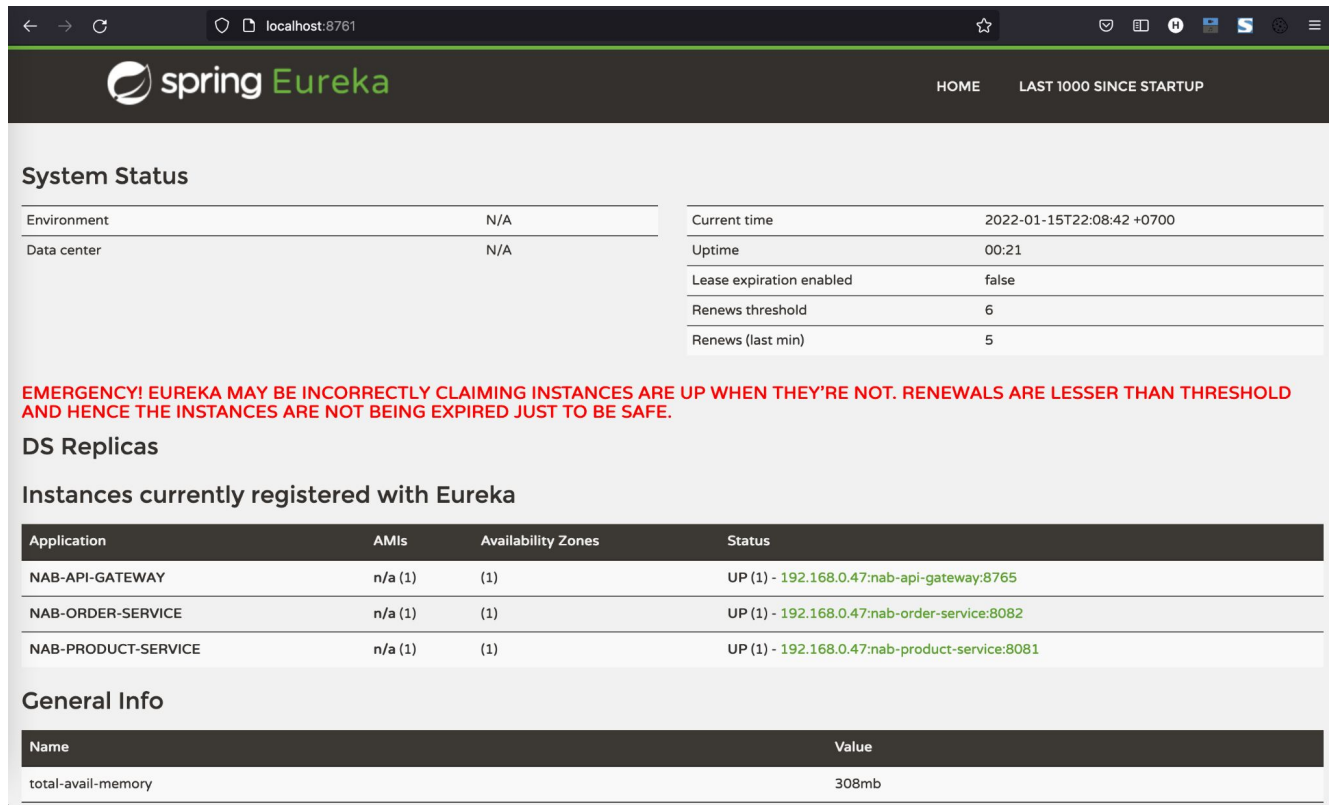
- Back-end:
 - Spring Boot
 - Spring Cloud: API-Gateway, Eureka, OpenFeign
 - Spring Data JPA, Spring JDBC
 - Kafka service
- Front-end:
 - ReactJS, NEXT
- DB: MySQL

Code quality

- Spotless Java check
- Unit Test
- Integration Test using H2 DB (to be updated)
- SOLID
- DRY

Functionalities by screenshots

Eureka server (check to make sure all services ready)



The screenshot shows the Spring Eureka server interface. The top navigation bar includes the Spring Eureka logo and links for HOME and LAST 1000 SINCE STARTUP. The main content area is divided into several sections:

- System Status:** A table showing environment and data center details, and another table showing system metrics like current time, uptime, and lease expiration.
- Emergency Warning:** A red text block indicating a potential issue with instance status and renewals.
- DS Replicas:** A section for distributed system replicas.
- Instances currently registered with Eureka:** A table listing registered applications and their status.
- General Info:** A table showing general system information.

System Status	
Environment	N/A
Data center	N/A

System Metrics	
Current time	2022-01-15T22:08:42 +0700
Uptime	00:21
Lease expiration enabled	false
Renews threshold	6
Renews (last min)	5

EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE.

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
NAB-API-GATEWAY	n/a (1)	(1)	UP (1) - 192.168.0.47:nab-api-gateway:8765
NAB-ORDER-SERVICE	n/a (1)	(1)	UP (1) - 192.168.0.47:nab-order-service:8082
NAB-PRODUCT-SERVICE	n/a (1)	(1)	UP (1) - 192.168.0.47:nab-product-service:8081

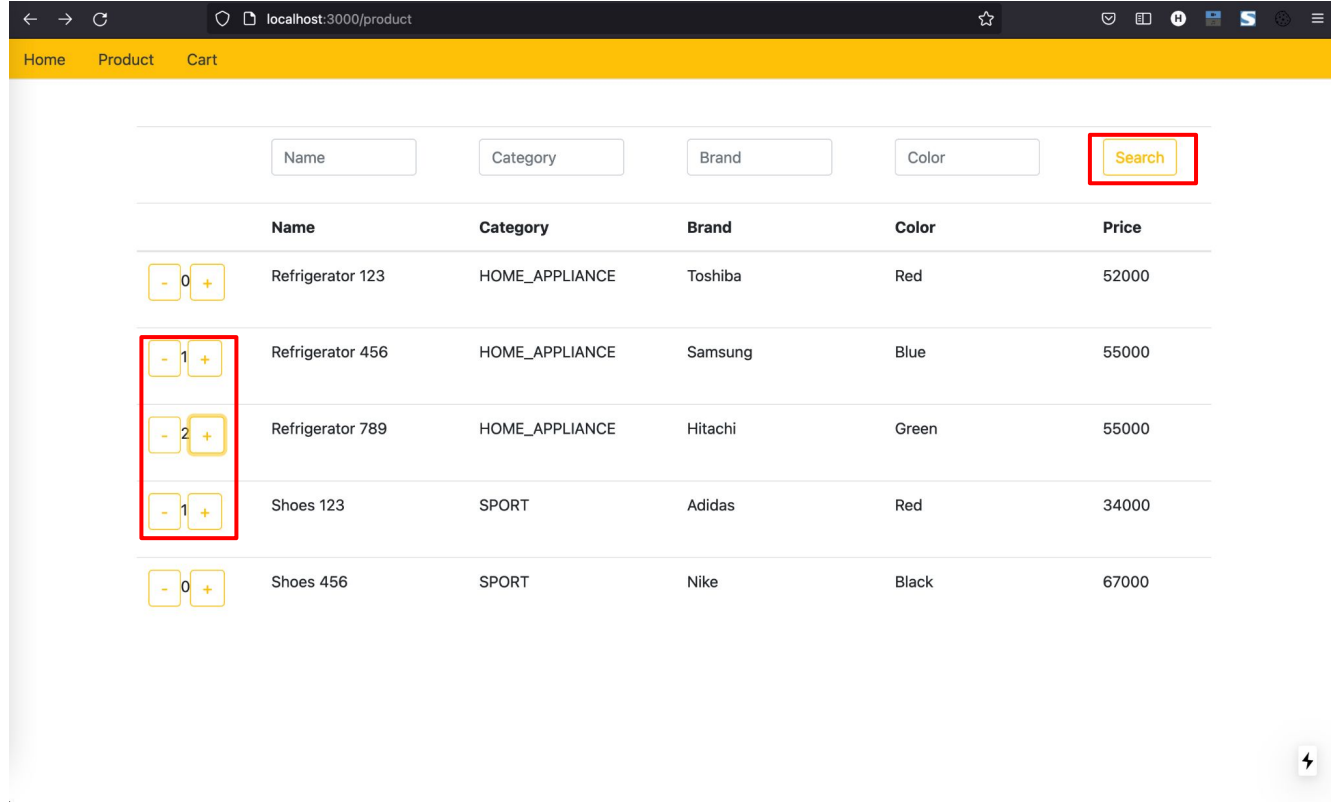
General Info

Name	Value
total-avail-memory	308mb

Products list

<div><div>← → ↺</div><div>localhost:3000/product</div><div>☆</div><div>📌 📄 🏠 🌐 📺 ⚙️ ☰</div></div>					
<div>HomeProductCart</div>					
<div><div><div>Name</div><div>Category</div><div>Brand</div><div>Color</div><div>Search</div></div></div>					
	Name	Category	Brand	Color	Price
<div><div>-</div><div>0</div><div>+</div></div>	Refrigerator 123	HOME_APPLIANCE	Toshiba	Red	52000
<div><div>-</div><div>0</div><div>+</div></div>	Refrigerator 456	HOME_APPLIANCE	Samsung	Blue	55000
<div><div>-</div><div>0</div><div>+</div></div>	Refrigerator 789	HOME_APPLIANCE	Hitachi	Green	55000
<div><div>-</div><div>0</div><div>+</div></div>	Shoes 123	SPORT	Adidas	Red	34000
<div><div>-</div><div>0</div><div>+</div></div>	Shoes 456	SPORT	Nike	Black	67000

Order products



Submit order

← → ↻

localhost:3000/cart

☆

📧 📅 🏠 🌐 📄 🌐 ☰

Home Product **Cart**

Submit

Product Id	Name	Price	Quantity	Total
2	Refrigerator 456	55000	1	55000
3	Refrigerator 789	55000	2	110000
4	Shoes 123	34000	1	34000

Checking DB (product table)

```
1 SELECT * FROM nab_db.nab_products;
```

100% 35:1

35:1

Result Grid



Filter Rows:



Search

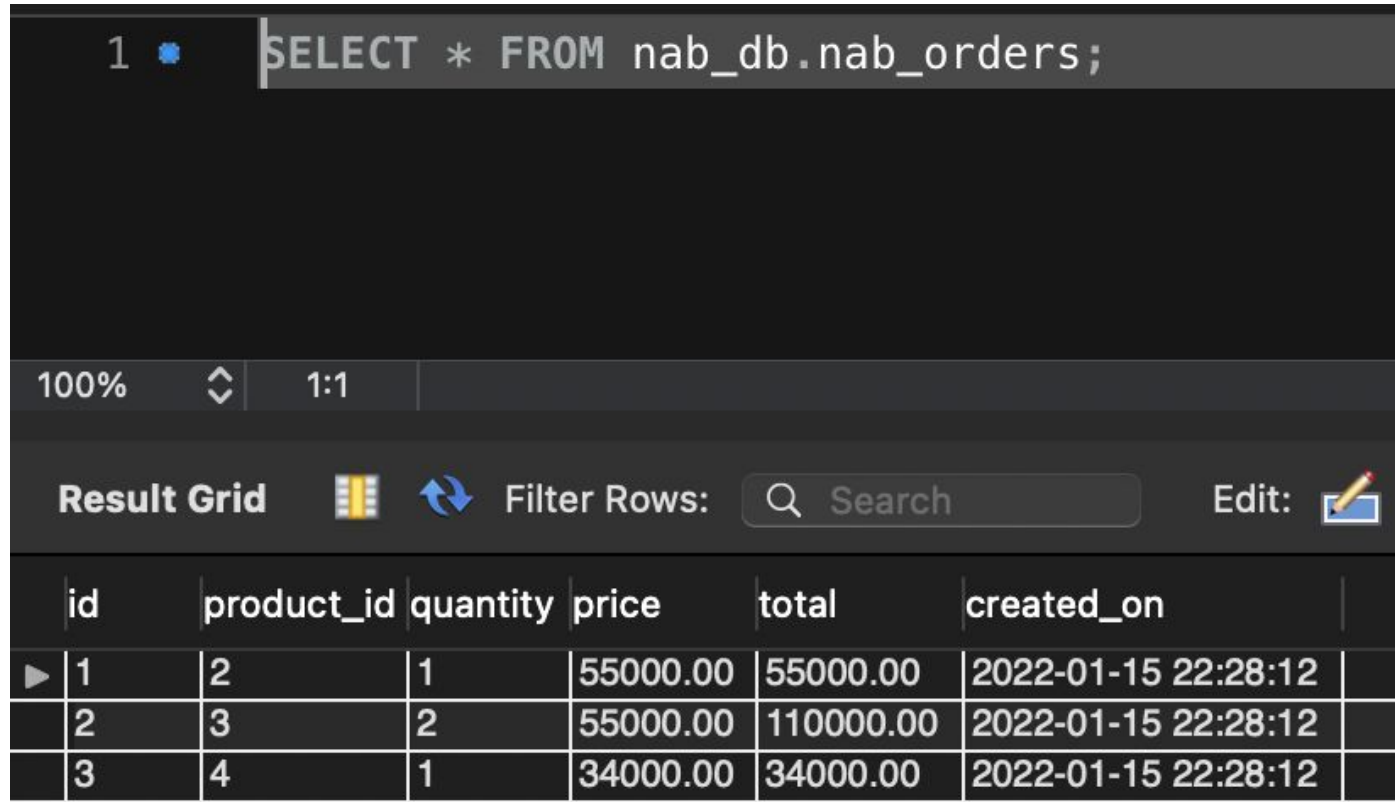
Edit:



Export/Import:

[illegible]

Checking DB (order table)



The screenshot shows a database client interface with a dark theme. At the top, a SQL query is entered in a text area: `SELECT * FROM nab_db.nab_orders;`. Below the query area, there are controls for zoom (100%), a refresh icon, and a zoom level (1:1). A toolbar contains the text "Result Grid", a grid icon, a refresh icon, the text "Filter Rows:", a search input field with a magnifying glass icon and the word "Search", and an "Edit:" label with a pencil icon. Below the toolbar is a table with 7 columns: `id`, `product_id`, `quantity`, `price`, `total`, and `created_on`. The table contains 3 rows of data. A play button icon is visible to the left of the first row.

```
1 SELECT * FROM nab_db.nab_orders;
```

100% 1:1

Result Grid Filter Rows: Search Edit:

	id	product_id	quantity	price	total	created_on
▶	1	2	1	55000.00	55000.00	2022-01-15 22:28:12
	2	3	2	55000.00	110000.00	2022-01-15 22:28:12
	3	4	1	34000.00	34000.00	2022-01-15 22:28:12

What need improving

Things should be improving

- Back-end technologies & functionalities:
 - Spring Security for API-Gateway.
 - Spring Config server with config files stored in GitHub.
 - Docker
 - Kubernetes
 - AWS CodePipeline
 - Zipkin server for tracking logs
 - Swagger or Spring REST docs.
 - Caching
 - More testing (integration test using H2 DB)
- Front-end
 - UT for every function
 - Common constants
 - Style checking

New Workflow (suggestion for next step)

