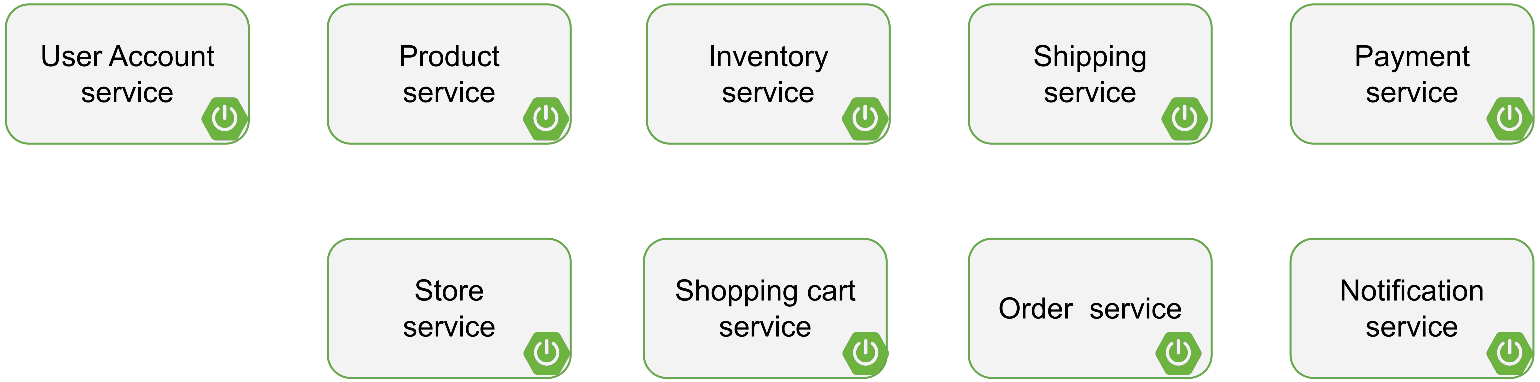
Shopping Portal Project

Hedi Abed ~ 15/30/2020

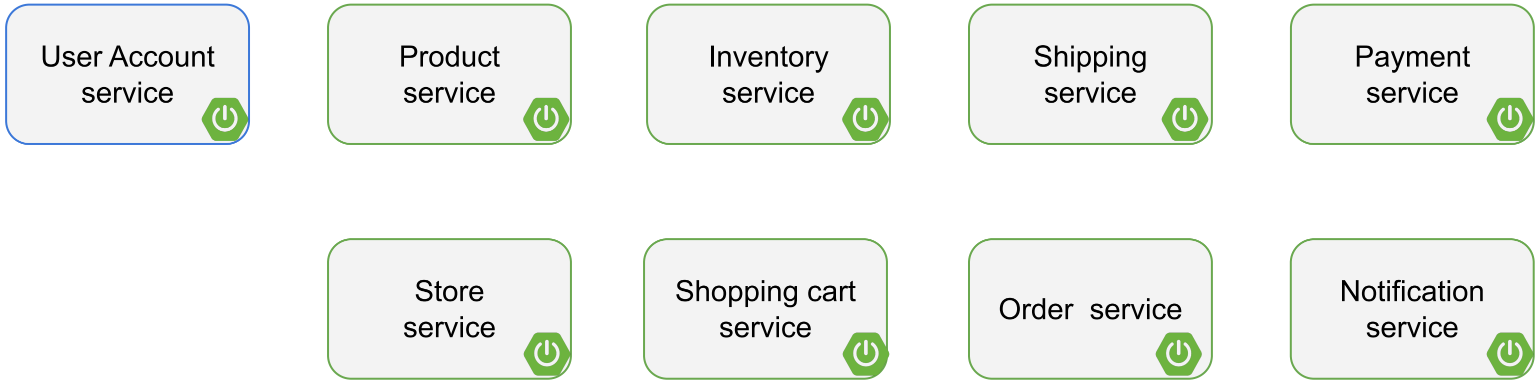
Shopping portal

Microservices :
The application contains a set of microservices:



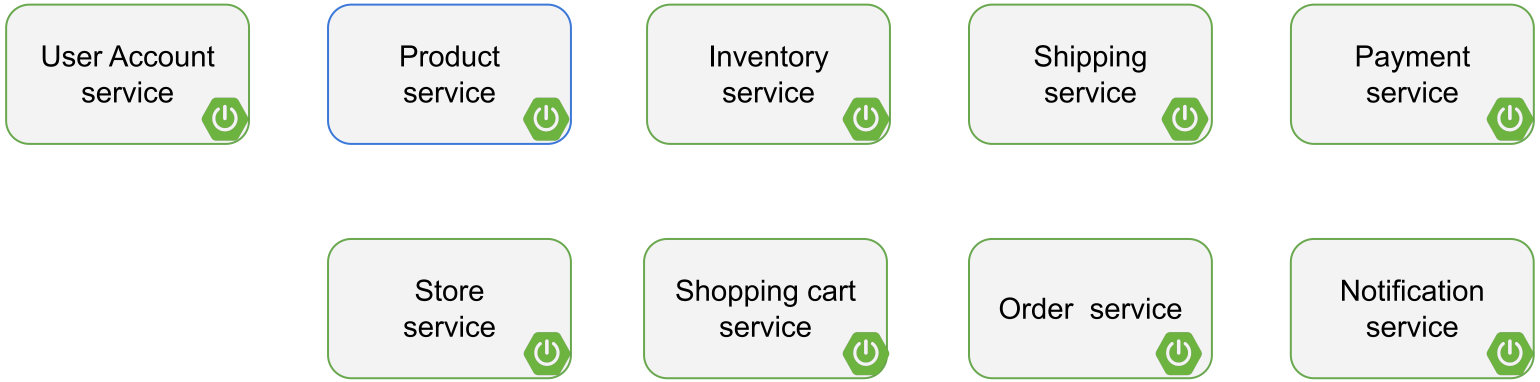
Shopping portal

Account microservice :
Provides user account operation functionality and attached to SQL database.
Ex: CRUD, Activate, Deactivate and Verified ...



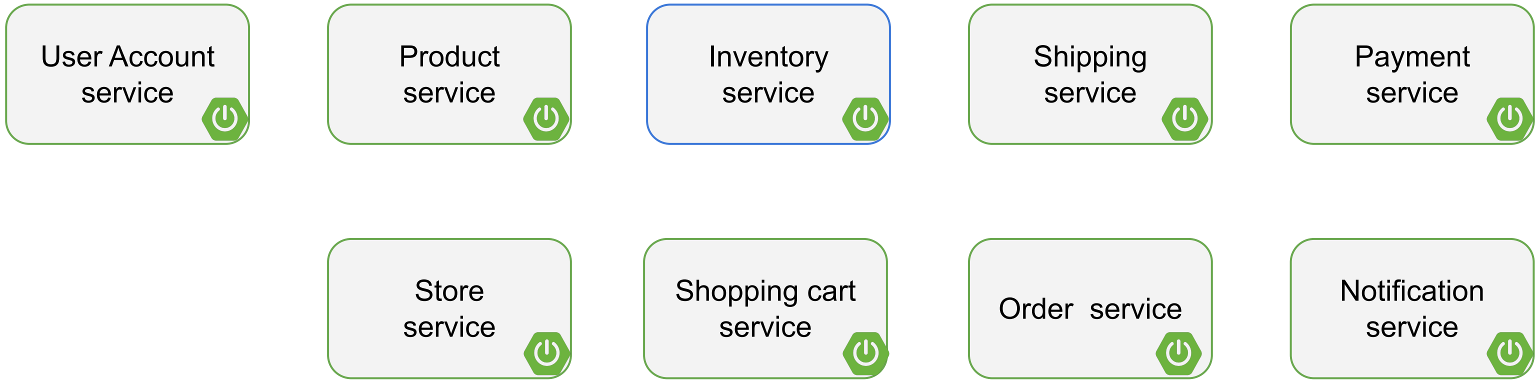
Shopping portal

Product microservice :
Provides product operation functionality and attached to SQL database.
Ex: CRUD, status ...



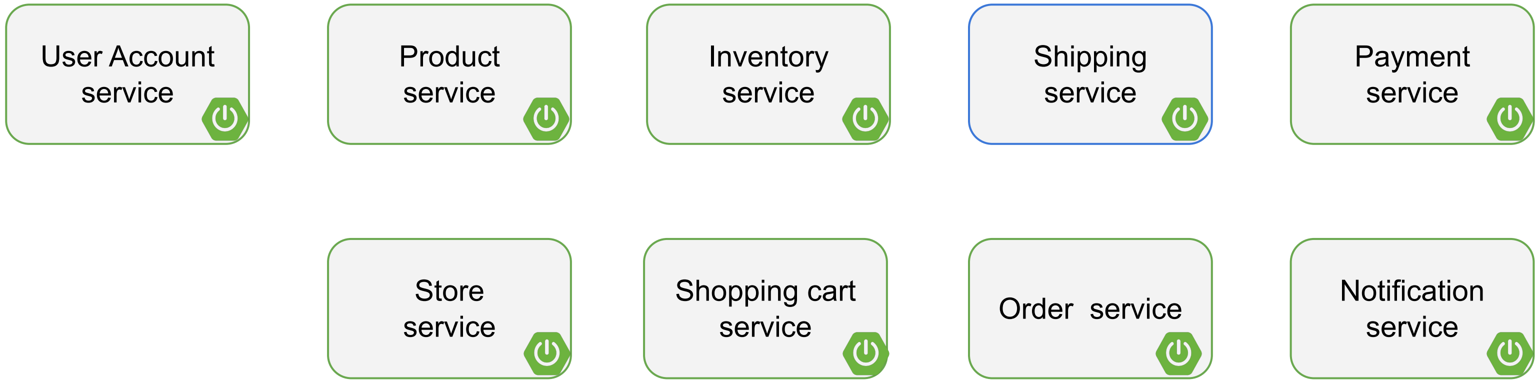
Shopping portal

Inventory microservice:
Provides product inventory operation functionality and attached to Redis database.
Ex: retrieve, increase, decrease and infinity.



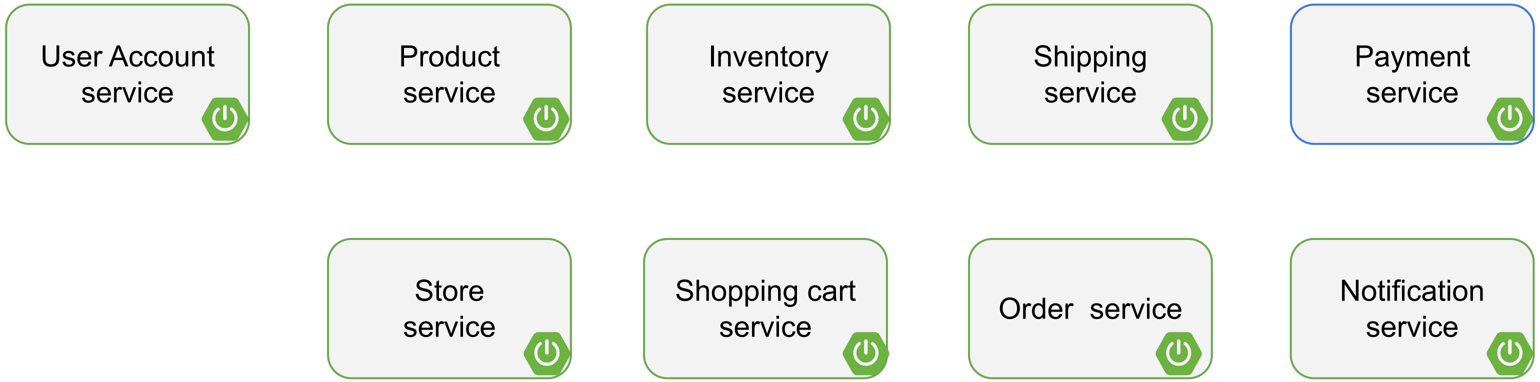
Shopping portal

Shipping microservice:
Provides shipping operation functionality and attached to NoSQL database.
Ex: CRU



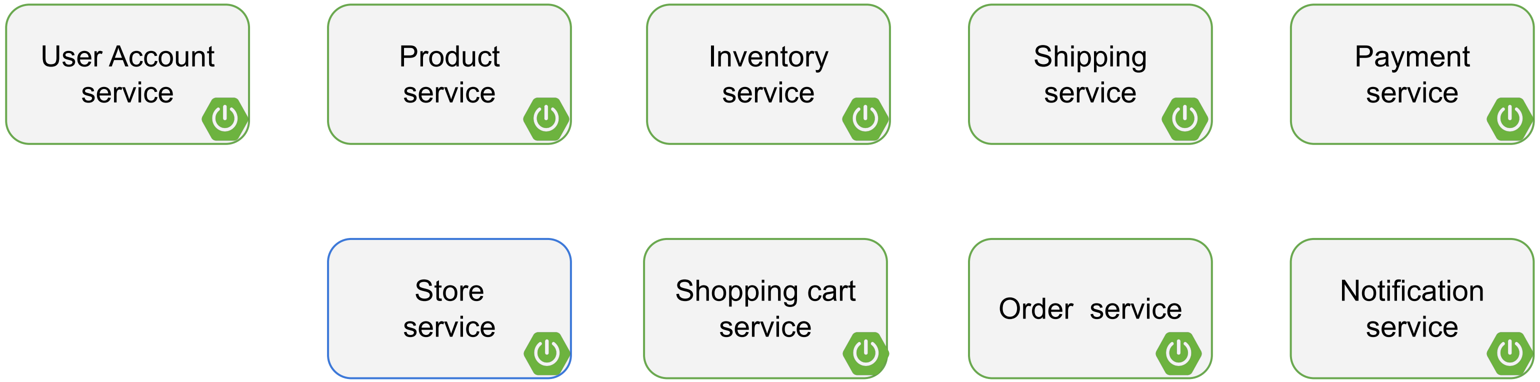
Shopping portal

Payment microservice:
Provides payment functionality and attached to online payment services.
Ex: Stripe, Paypal ...



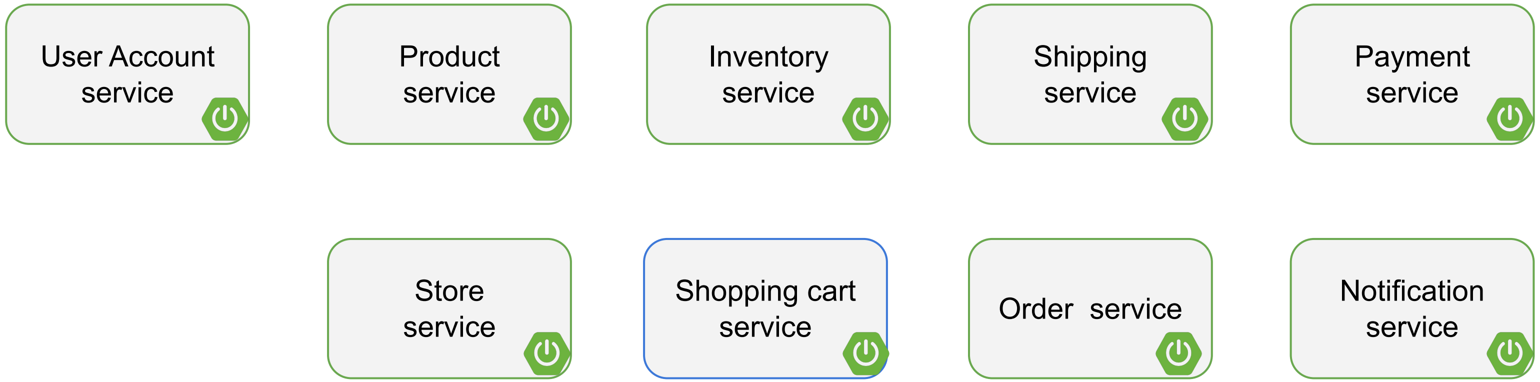
Shopping portal

Store microservice:
Provides personal shop operation functionality and attached to NoSQL database.
Ex: CRUD ...



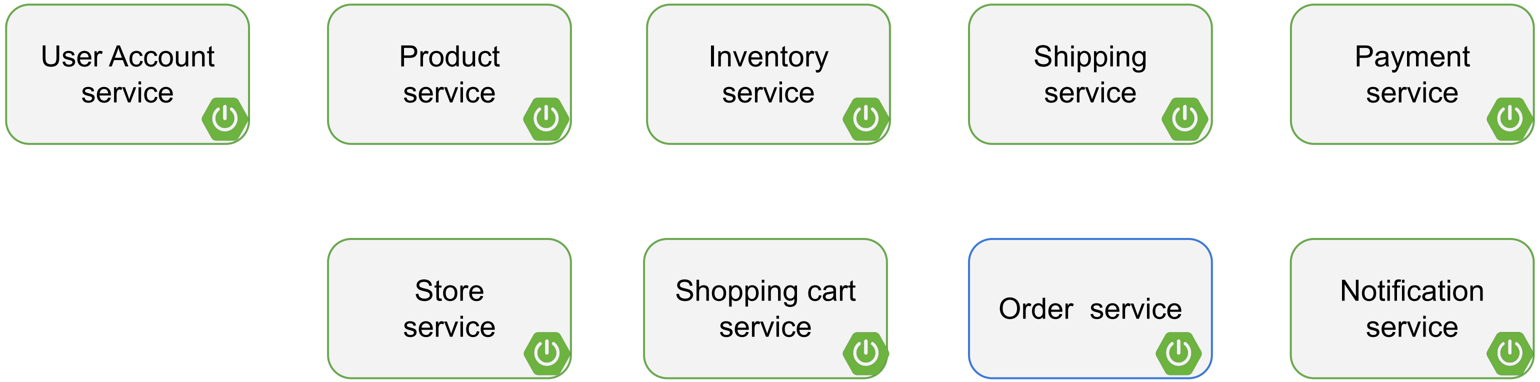
Shopping portal

Shopping cart microservice:
Provides shopping cart operation functionality, generates orders and attached to Redis database.
Ex: add, remove and checkout ...



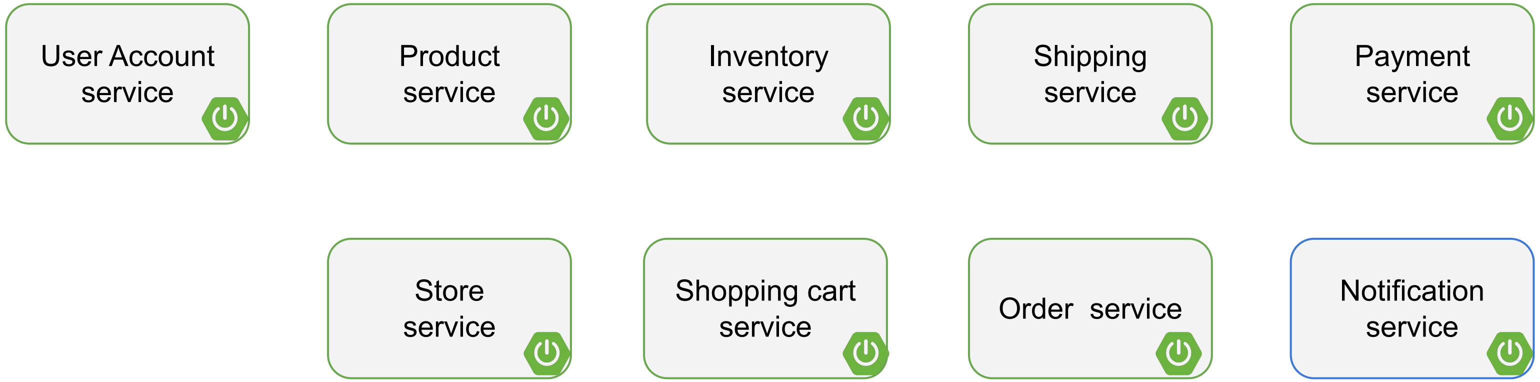
Shopping portal

Order microservice:
Receives order requests from the cart service via event bus and attached to Redis database.
Ex: Create, Updating Status...



Shopping portal

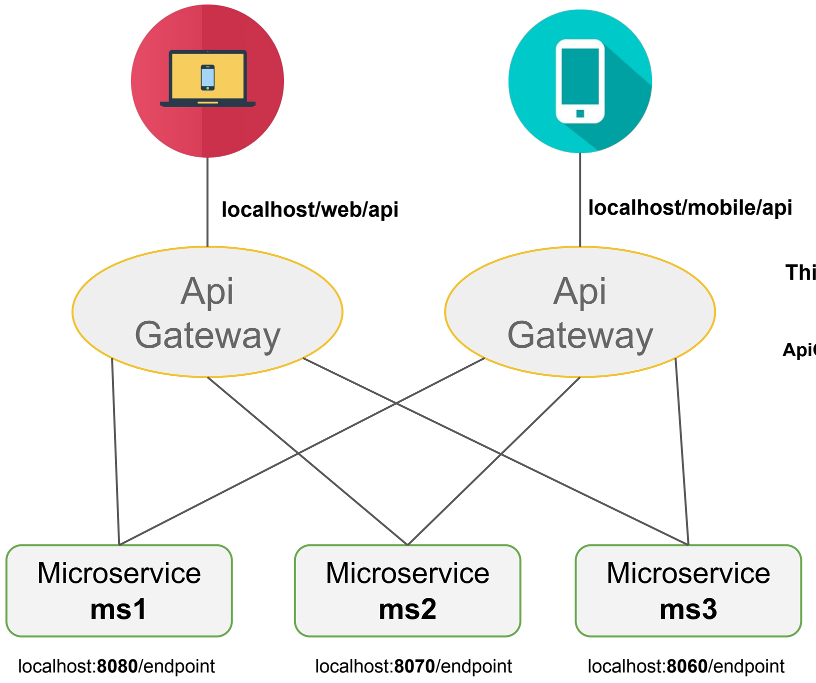
Notification microservice:
Provides notification operation functionality and attached to NoSQL database.
Ex: Send emails, Socket and firebase notifications...



Shopping portal

BFF vs API Gateway

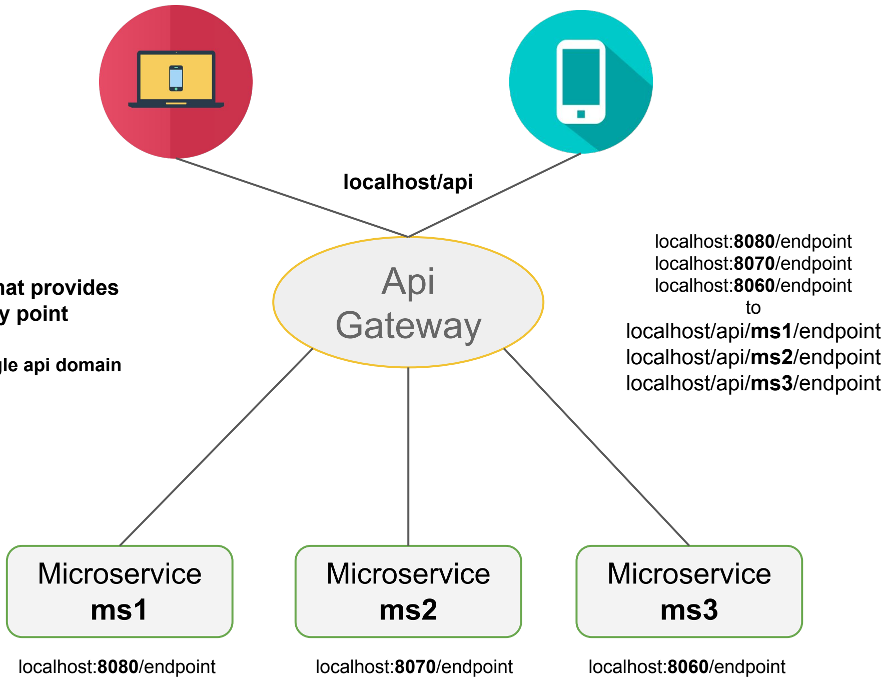
Multiple API Gateways provide separate APIs for each client



This is a service that provides a single-entry point

ApiG: Maintain a single api domain

API Gateway provides the same API for all clients



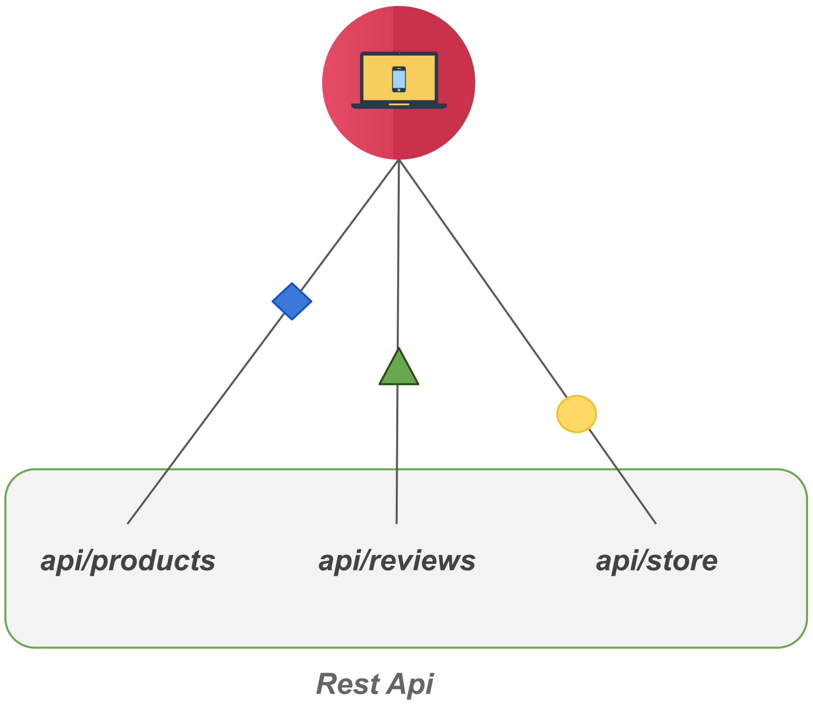
Shopping portal

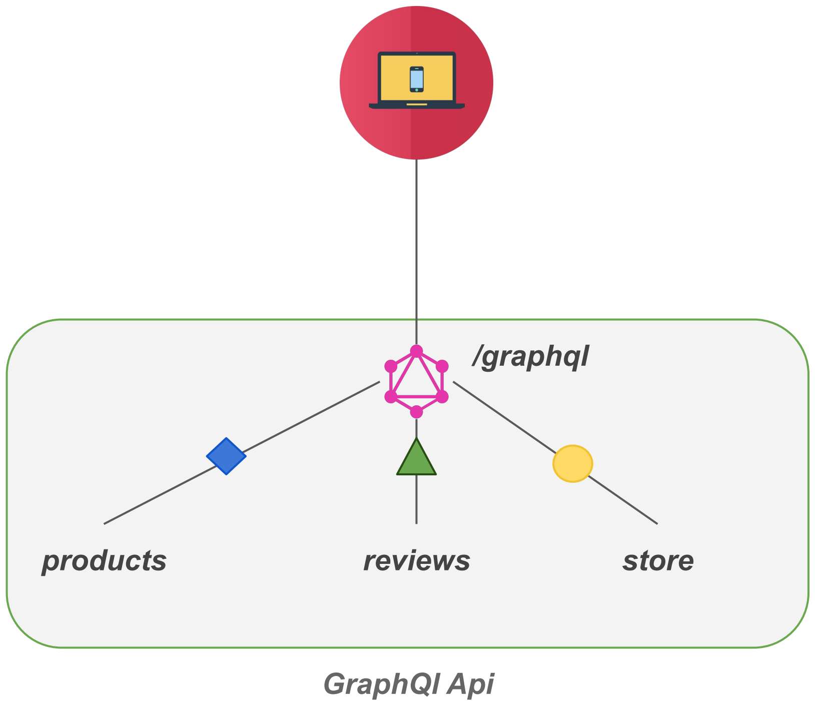
REST vs GraphQL

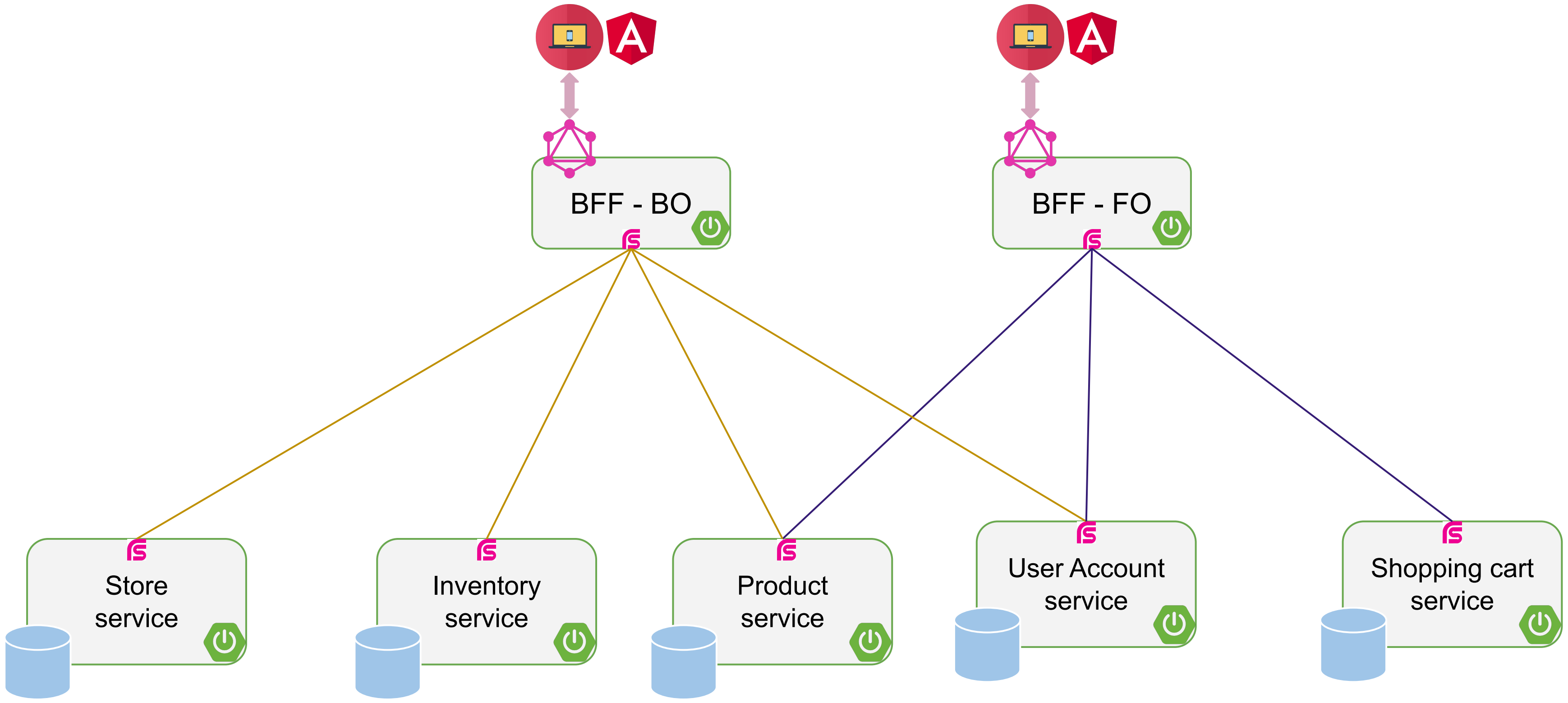
- One of the major differences is that with GraphQL, you only have one endpoint.
- With a single request. you can get an object and its related objects, while in REST you have different endpoints that can access different resources which means that if you need data from different resources you have to make different calls.
- For REst each call returning complete objetcs probably with data you don’t even need…

Shopping portal

REST vs GraphQL

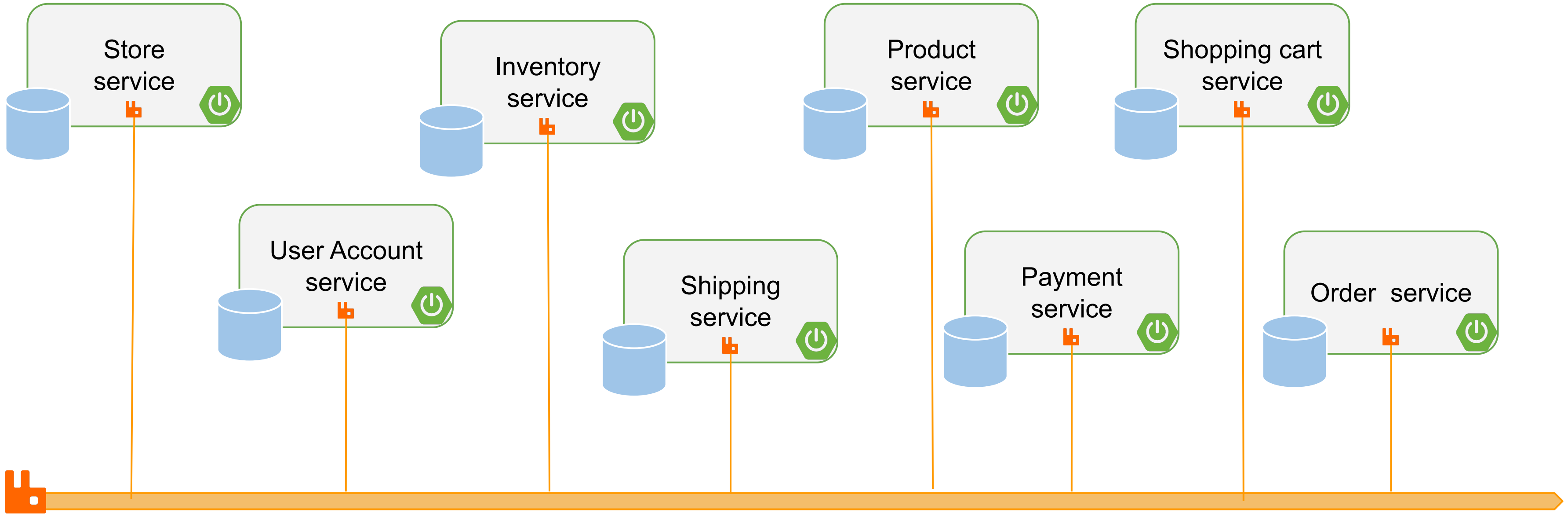


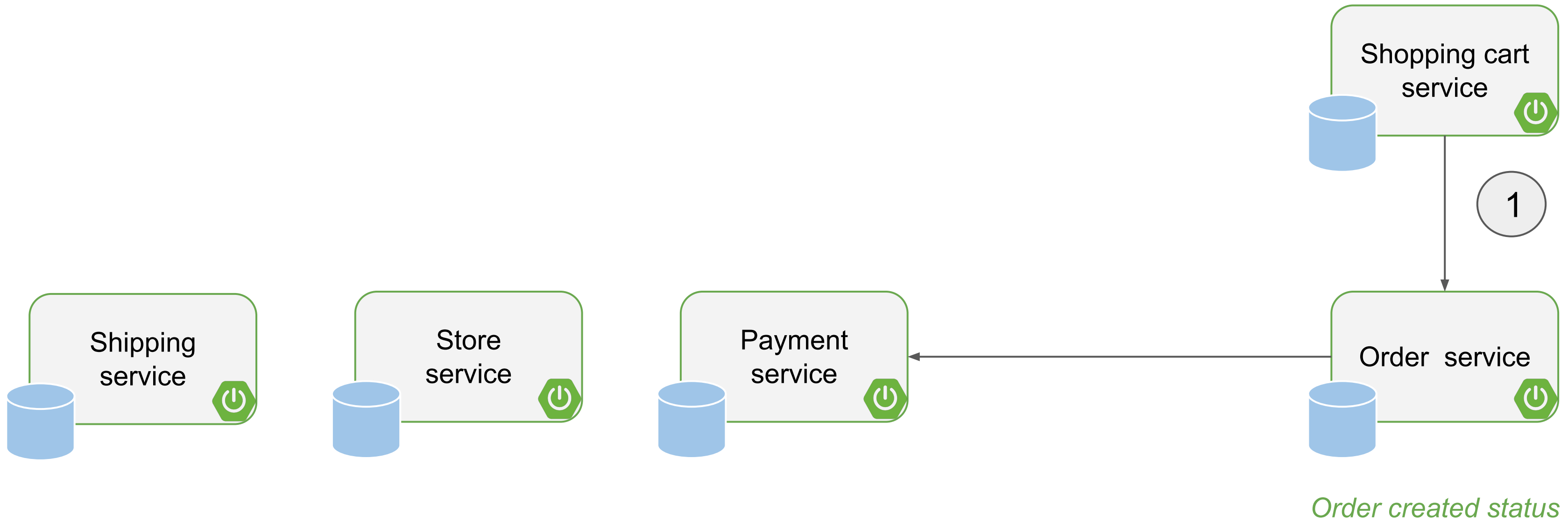




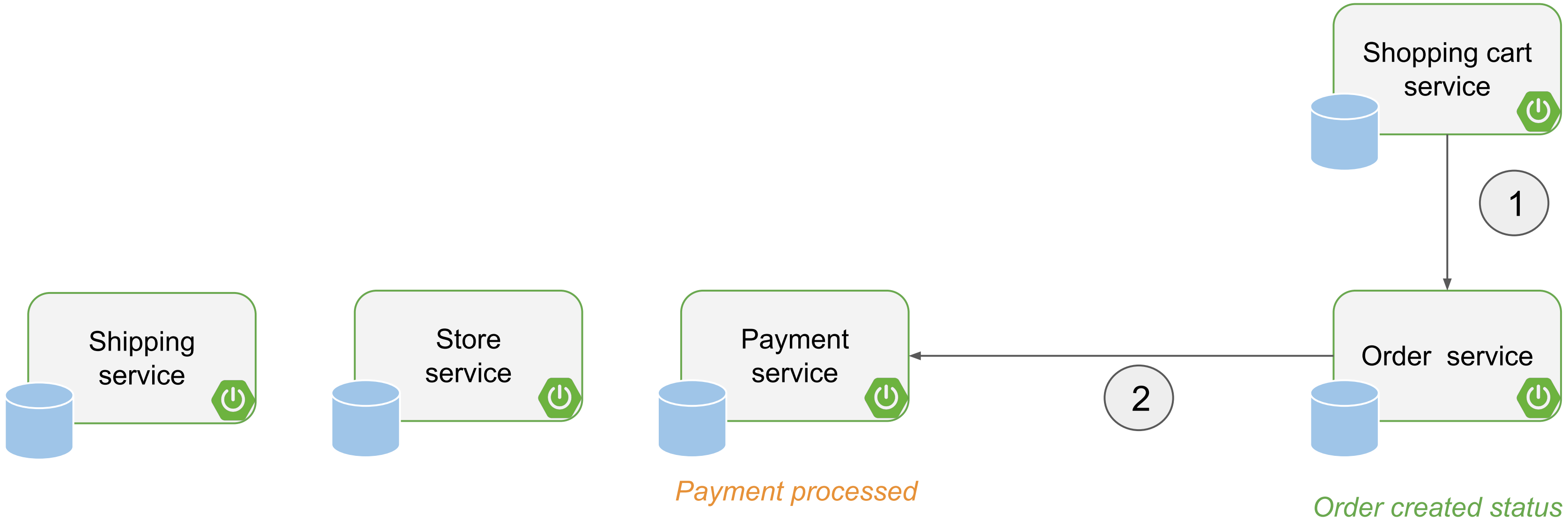
- RSocket is a new, message-driven, binary protocol that standardizes the approach to communication in between microservices. support for multiplexing and binarized payloads
- Interaction model of HTTP problems: the server has to send a response back to the client, even if the client is not interested in processing it. The size of data is higher than in the case of binary protocols.

- The communication is made by sending messages that contain information or commands that need to be processed. The sender is called the Producer.
- These messages are stored (in memory or persisted) in a queue and processed by another microservice (called the Consumer). Then, once a message is processed, it is removed or dequeued, which assures that it is processed only once.

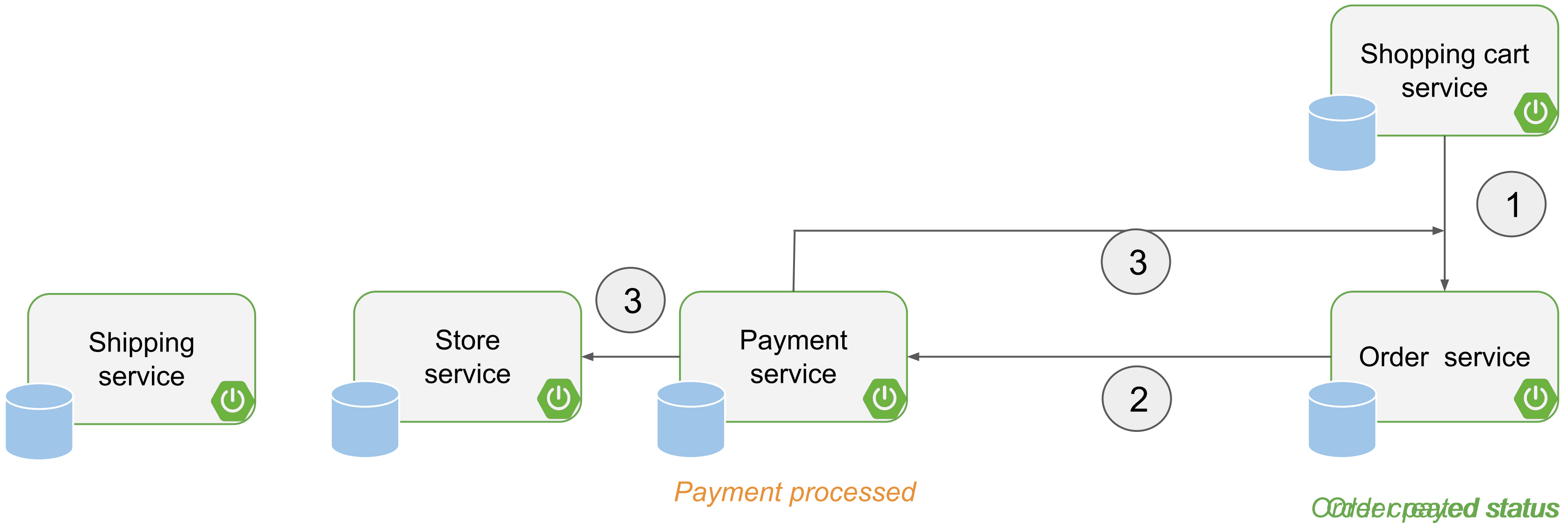




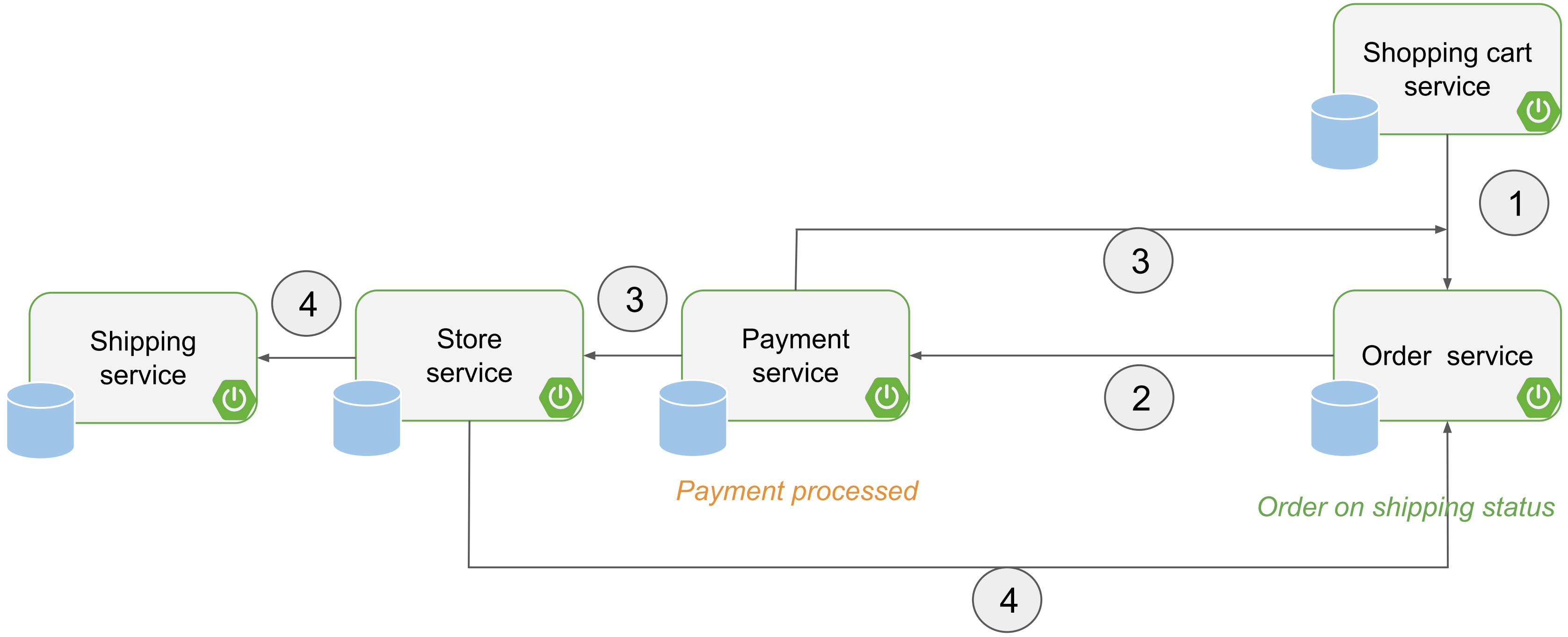
1. The order services triggered first after checkout



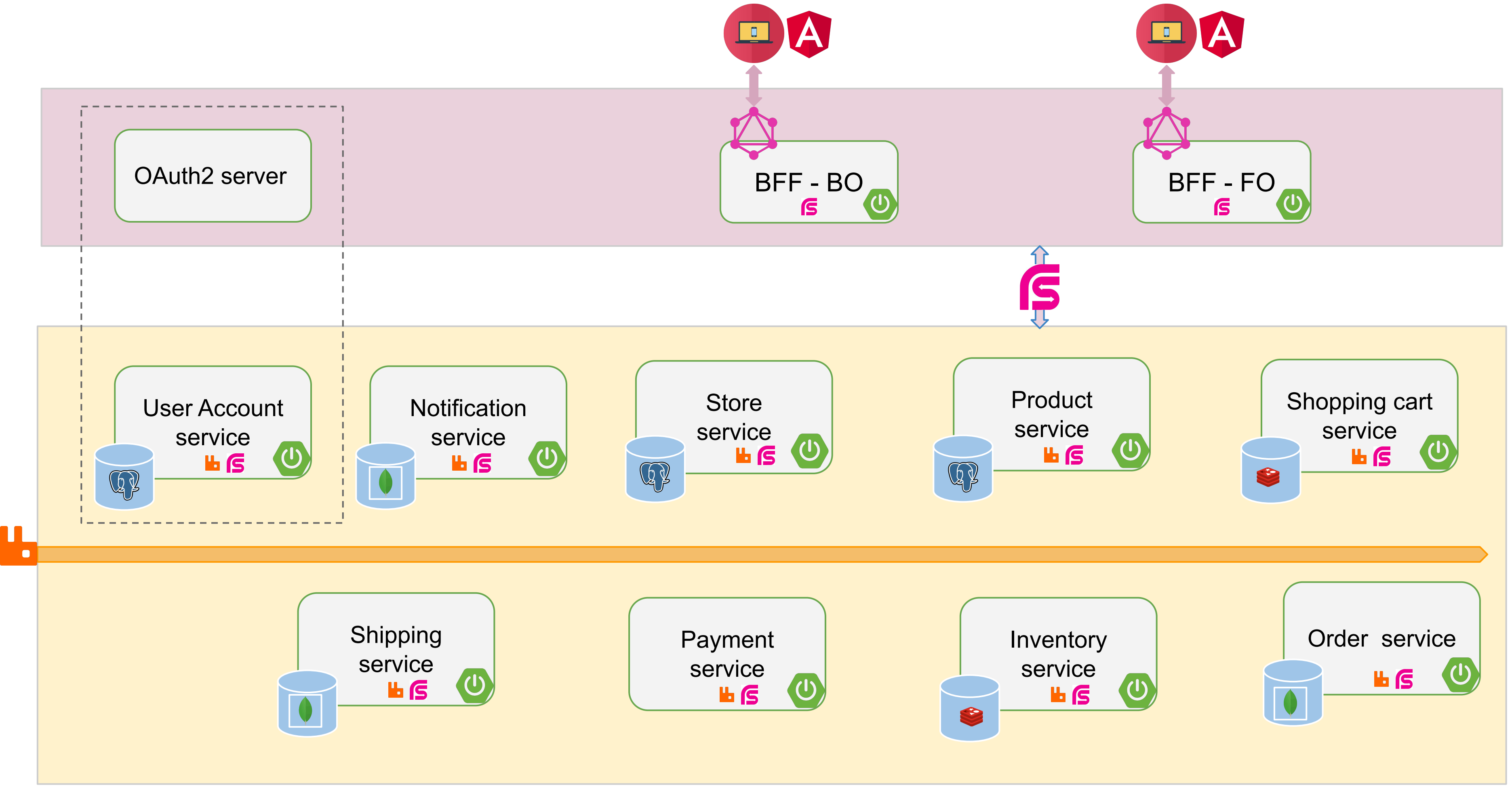
1. The order service triggered first after checkout
2. The order service trigger the payment service and payment service does the payment for this particular order



1. The order service triggered first after checkout
2. The order service trigger the payment service and payment service does the payment for this particular order.
3. Once the payment finishes the transaction,it notifies the order service and the store



1. The order service triggered first after checkout
2. The order service trigger the payment service and payment service does the payment for this particular order.
3. Once the payment received the transaction, it notifies the order service and the store.
4. After the store service finish booking this order, it trigger the shipping service and order service


OAuth2 server
BFF - BO
BFF - FO
User Account service
Notification service
Store service
Product service
Shopping cart service
Shipping service
Payment service
Inventory service
Order service

```
System.out.print(“THE END”);
```