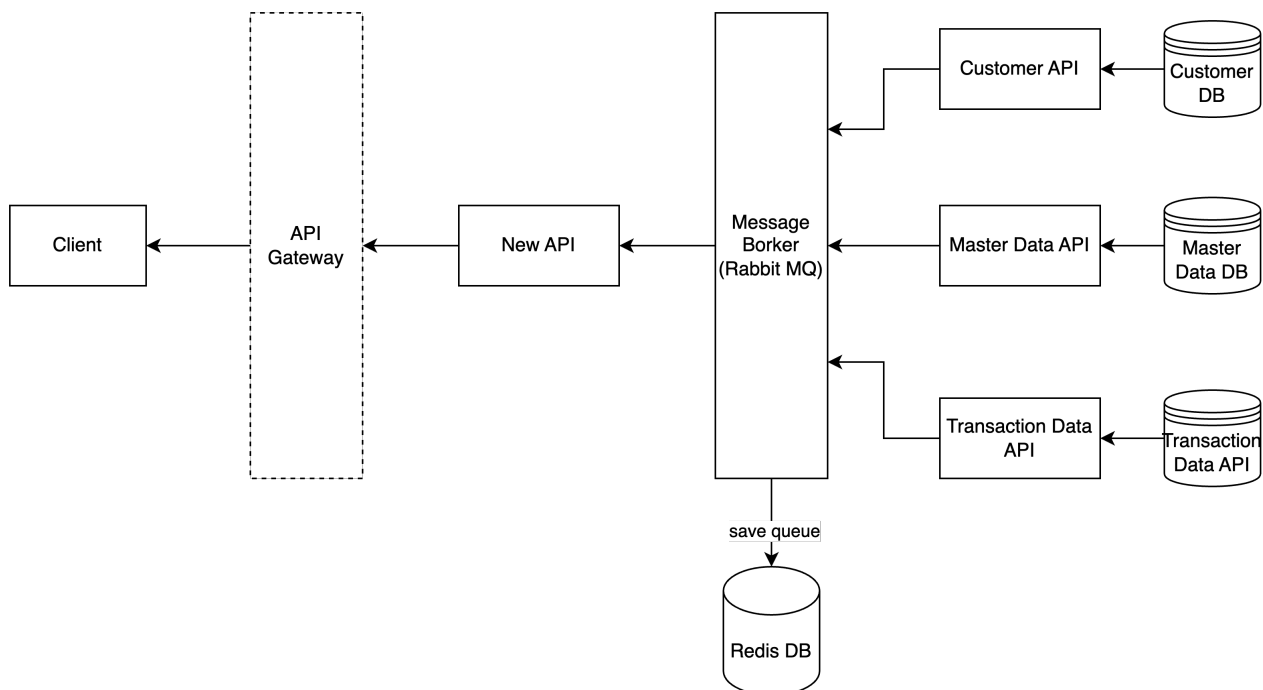


## Backend Questions

1. Assuming the system currently has three microservices: Customer API, Master Data API, and Transaction Data API, there is a new feature that requires data from all three microservices to be displayed in near real-time. The current technology stack includes REST APIs and an RDBMS database. How would you design a new API for this feature?

การออกแบบระบบ API เพื่อเชื่อมต่อ Customer API, Master Data API และ Transaction Data API ต้องมีการวางโครงสร้างระบบให้มีตัว Message Broker เพื่อเชื่อมต่อระหว่าง API เช่น RabbitMQ และอาจจะมี Redis Database เพื่อบันทึกคิว หากมีการติดต่อข้อมูลในปริมาณมาก ๆ และตัว API ที่ออกมาอาจจะต้องส่งอัปเดตข้อมูลส่งต่อไปยัง API Gateway อีกที



2. Assuming the team has started planning a new project, the project manager asks you for a performance test strategy plan for this release. How would you recommend proceeding to the project manager?

เริ่มจากการกำหนดเป้าหมายของ release ว่าเราทำการ release เพื่ออะไร มีตัวชี้วัดอะไรบ้าง สิ่งที่เราส่งไปนั้นมีส่วนประกอบอะไรบ้าง เช่น backend มี release เกี่ยวกับอะไร frontend เกี่ยวกับอะไร หลังจากนั้นก็จะเป็นการ test ในภาพรวมของระบบเช่น UAT Test หรือ End to End Test ในสิ่งที่ release ไป โดยอาจจะรวมกับสิ่งที่มีในระบบเดิมอยู่แล้ว แต่อาจจะมีผลกระทบด้วย หลังจากนั้นก็มีวางแผนเรื่อง performance test โดยอ้างอิงจากจำนวน load การใช้งานเดิมของระบบว่า ปกติจะมีการใช้มากที่สุดเท่าไร แล้วลองทดสอบดูว่าหากมีจำนวน load เท่าเดิมหรือมากกว่า ระบบจะมีปัญหาไหม แล้วก็แผนหลัง release ที่จะมีการติดตาม log การใช้งานระบบ โดยอาจจะมี tools รายงานเช่น Grafana ที่จะคอย report ให้

### 3. Design and develop two APIs using NestJS and Postgres

#### 1. Create a Multilingual Product API

- Method: POST
- Url: /api/product/create
- Example Body:

```
{
  "name": {
    "en": "Name",
    "th": "ชื่อ"
  },
  "description": {
    "en": "Name",
    "th": "ชื่อ"
  },
  "price": 10
}
```

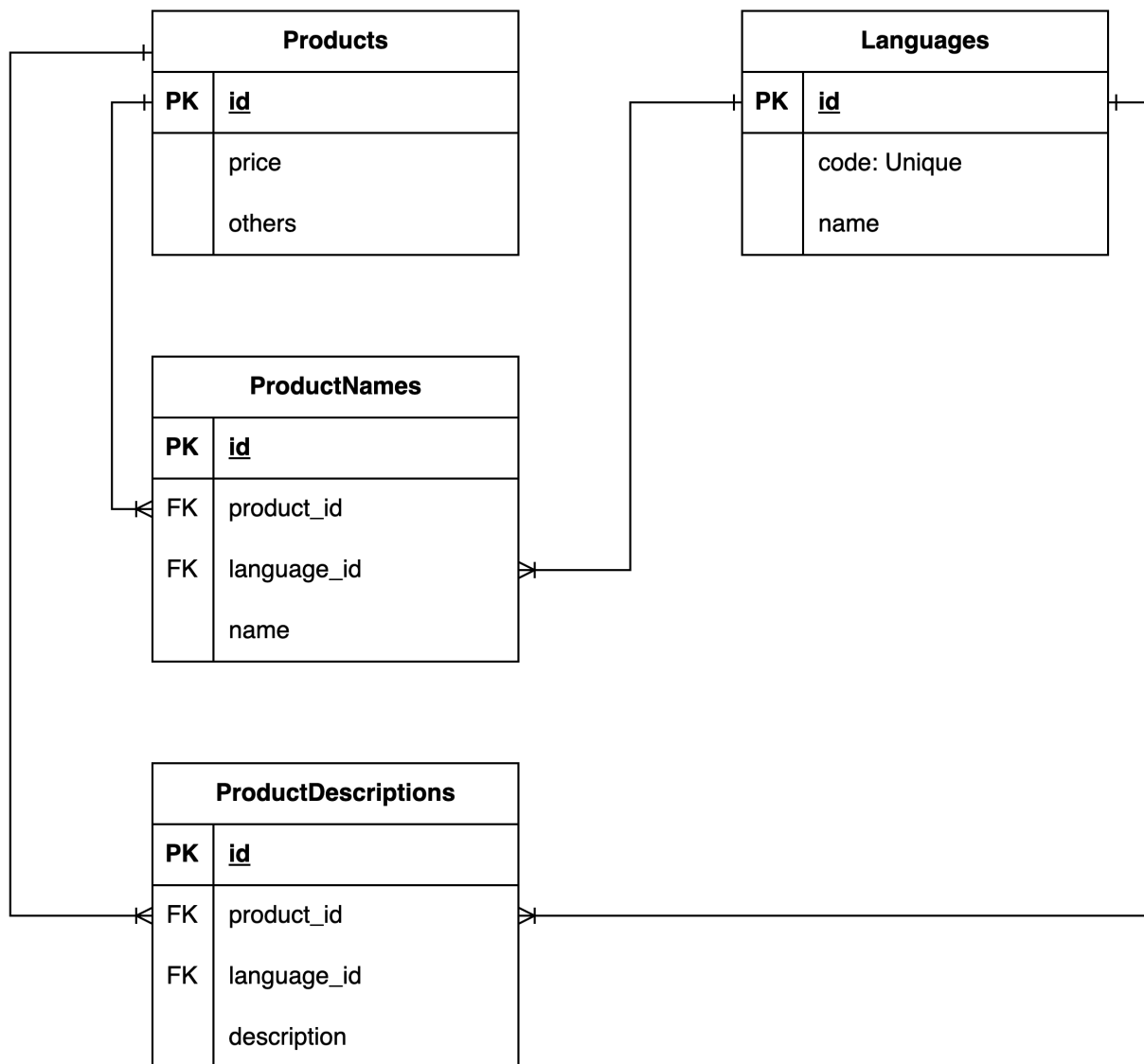
#### 2. Multilingual Product Search API

- Method: GET
- Url: /api/product
- Params: key

การ Validation:

- ใช้ DTO ของ Nestjs ในการ validation ดูว่า data type ตรงไหม ลักษณะโครงสร้างถูกไหม

## Database Design: ER Diagram



### Testing Strategy:

- unit test แต่ละฟังก์ชันควรมี coverage เกิน 80% หรือ ถ้ามีเวลาในการทำที่เหมาะสม
- integration หรือ e2e แต่ละ flow ควรจะครบตาม test case ที่ทาง QA กำหนดไว้

# React Questions

## 1. useCallback ใช้ทำอะไร

useCallback เป็น hooks ของ React ที่ประกอบไปด้วย 2 ส่วนคือ ตัว function เอง และ ตัวแปรที่เป็น dependencies โดยจะนำมาใช้ทำ

1. function ที่มีการจดจำ process ภายใน โดยจะสามารถเรียกใช้ต่อเมื่อตัวแปรที่เป็น dependencies เปลี่ยนแปลงเท่านั้น

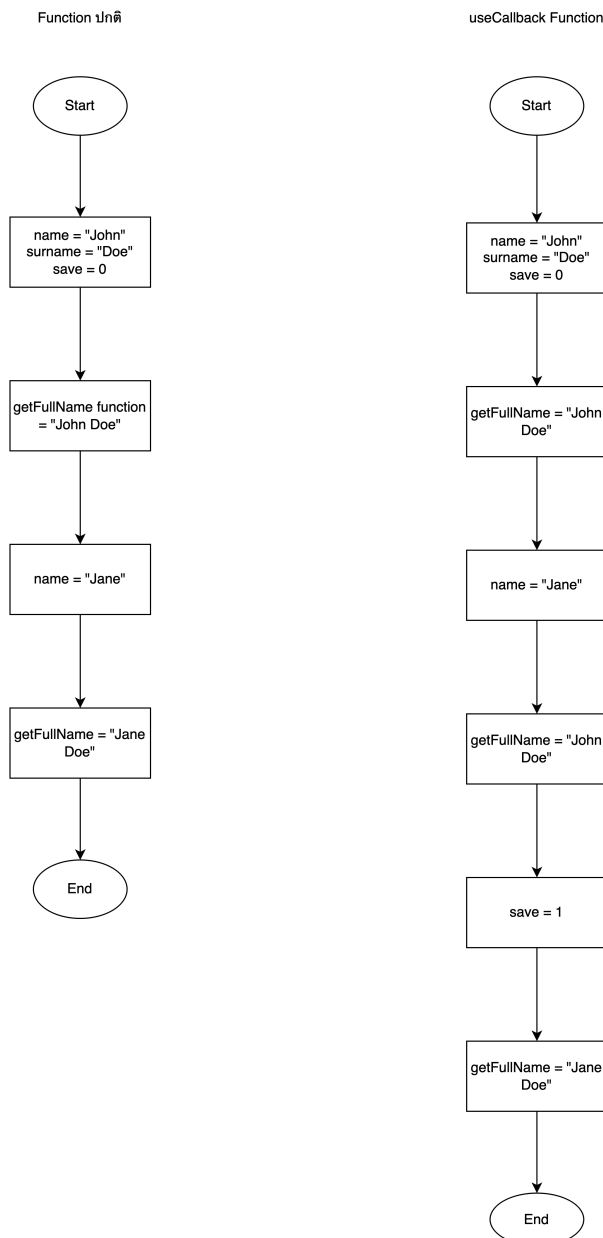
เช่น component Profile

- มี state ชื่อว่า name, surname, fullName และ save
- ต้องการ function getFullName() ที่จะนำ name มาต่อกับ surname แล้วนำมา setState

ของ fullName

- มีปุ่มที่เรียก function getFullName และมีปุ่มสำหรับเปลี่ยน state ของ save
- มี input สำหรับใส่ name, surname
- เมื่อมีการเปลี่ยนแปลงข้อมูลของ name หรือ surname จะต้องมีการกดปุ่ม save ก่อนเท่านั้น

ปุ่มเรียก getFullName ถึงจะการทำงาน



- ซึ่งถ้าเขียน function `getFullName` ตามปกติ ไม่ว่าข้อมูลอะไรเปลี่ยนแปลง เวลาเรียกใช้ก็จะนำค่าปัจจุบันมาใช้ แต่หากเขียน `getFullName` เป็น function ภายใน `useCallback` แล้วมี dependencies เป็น `save` เวลาเรียกใช้ก็จะใช้ค่าเดิมก่อนที่จะมีการเปลี่ยนแปลง state ของ `save`

2. function ที่มีขนาดใหญ่ที่หากมีการเรียกใช้ที่ไม่จำเป็นจะทำให้ระบบใช้ทรัพยากรมากเกินไป

## **2. Write a unit test for the UserProfile React component using Jest and React Testing Library.**

ไฟล์คำตอบอยู่ใน `react-2-solution/user-profile.spec.js`