UỶ BAN NHÂN DÂN

THÀNH PHỐ HỒ CHÍ MINH

**TRƯỜNG ĐẠI HỌC SÀI GÒN**



**HỌC PHẦN KIỂM THỬ PHẦN MỀM**

Giáo viên hướng dẫn: Thầy Đỗ Như Tài

Thành viên nhóm:

1. Nguyễn Văn Khánh Duy – 3122411027-kduy9180@gmail.com
2. Âu Thái Khang – 3122411086-authaikhang123@gmail.com
3. Nguyễn Đức Gia Bảo – 3122411011-ducbao8a1@gmail.com
4. Huỳnh Hữu Tứ- 3122411240- huynhhuutuwork@gmail.com

Mục lục

[Phân công việc 3](#_Toc209727397)

[Bài tập minh họa 4](#_Toc209727398)

[Bài 1 4](#_Toc209727399)

[Bài 2 4](#_Toc209727400)

[Bài 3 5](#_Toc209727401)

[Bài 4 5](#_Toc209727402)

[Bài 5 8](#_Toc209727403)

[Bài 6 9](#_Toc209727404)

[Bài 7 11](#_Toc209727405)

[Bài 8 12](#_Toc209727406)

[Bài tập vận dụng 13](#_Toc209727407)

[Level 1 system context 13](#_Toc209727408)

[Level 2 container 14](#_Toc209727409)

[Level 3 component 14](#_Toc209727410)

[Level 4 code 15](#_Toc209727411)

# Phân công việc

|  |  |
| --- | --- |
| Họ tên | Công việc |
| Nguyễn Đức Giao Bảo | Bài tập mình họa: 4,5  Bài tập vận dụng: level 1, level 4 |
| Huỳnh Hữu Tứ | Bài tập mình họa: 2, 8  Bài tập vận dụng: level 3 |
| Âu Thái Khang | Bài tập mình họa: 1,6,7  Bài tập vận dụng: level 2 |
| Nguyễn Văn Khánh Duy | Bài tập mình họa: 3 |

# Bài tập minh họa

## Bài 1

File Baitap3\_1.drawio

A diagram of a building

Description automatically generated

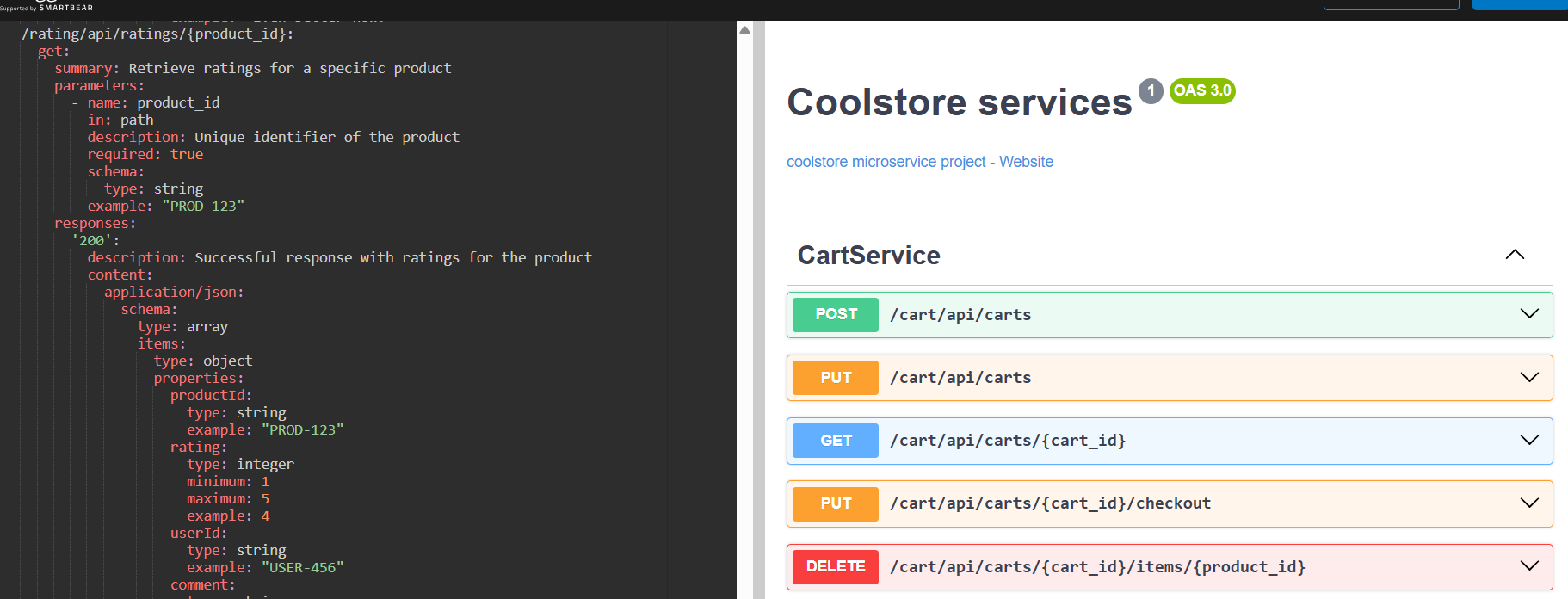
## Bài 2

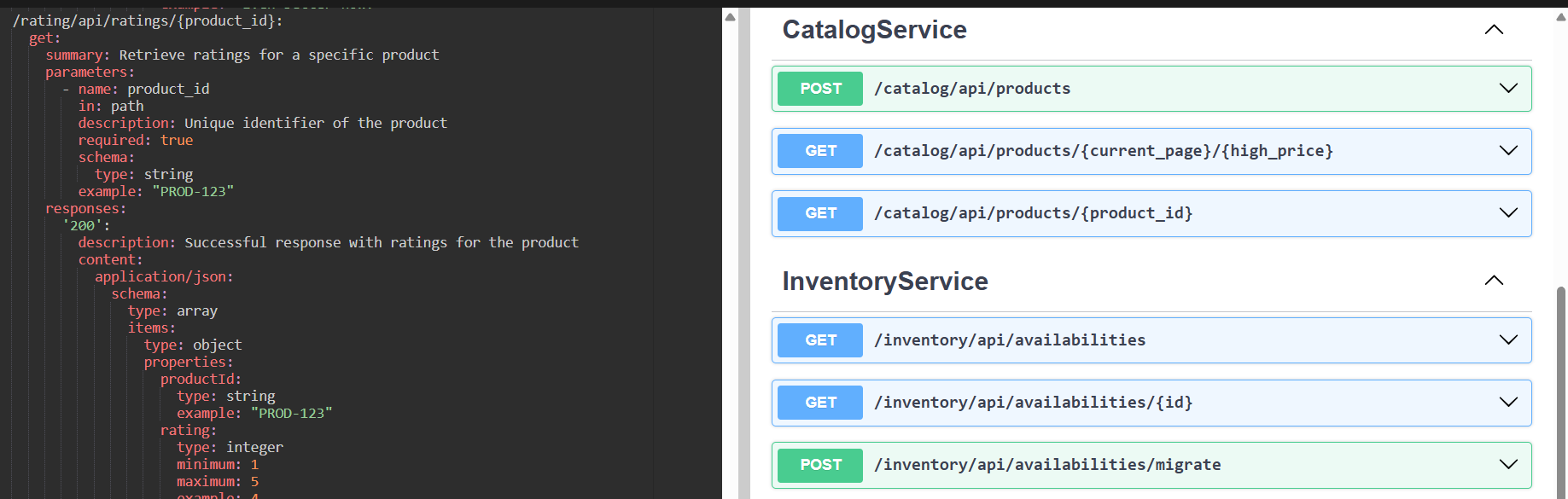
File Lab3\_2.drawio

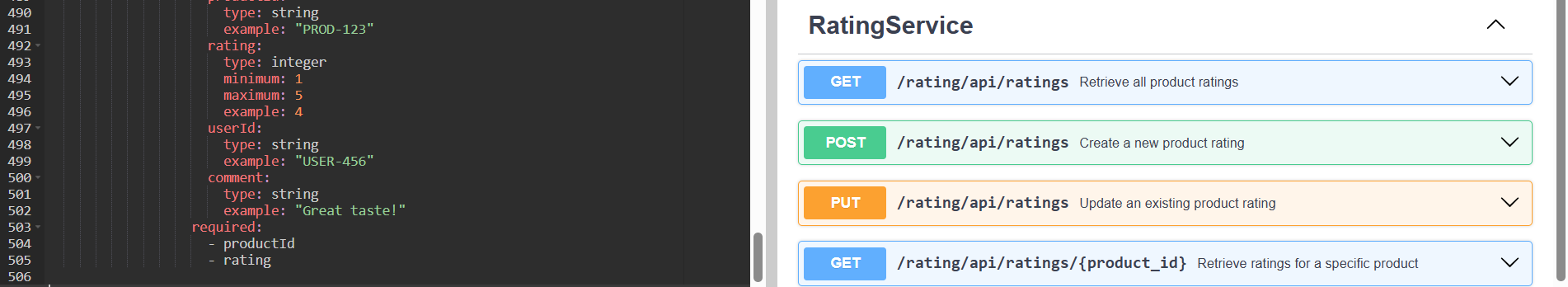
A diagram of a ship

Description automatically generated

## Bài 3







OpenAPI specification được lưu trong file swagger.yml

## Bài 4

File Cau4.drawio

A diagram of a meeting

Description automatically generated

Hệ thống gồm các tác nhân chính sau:

* Member (Person): Người sử dụng hệ thống, có thể là người tạo cuộc họp, có thể là người tham gia cuộc họp hoặc là thành viên trong nhóm.
* Adminstrator (Person): Người quản lý hệ thống phần mềm.
* My Meetings (System – Hệ thống chính): Hệ thống quản lý cuộc họp.
* Payment gateway (System – Hệ thống ngoại): Hệ thống quản lý cổng thanh toán.
* Email Service (System – Hệ thống ngoại): Hệ thống quản lý dịch vụ gửi mail

Quan hệ giữa các tác nhân:

* Giữa Member (Person) với My Meetings (System): Member có thể tổ chức hoặc tham gia vào những cuộc họp được tạo bên trong hệ thống.
* Giữa Member (Person) với Payment Service (System): Member sẽ thực hiện thanh toán các phí sử dụng dịch vụ thông qua hệ thống dịch vụ này.
* Giữa Member (Person) với Email Service (System): Member sẽ nhận mail thông báo của hệ thống thông qua hệ thống dịch vụ mail.
* Giữa Payment Gateway (System) với My Meetings (System):Hệ thống chính sẽ gửi các khoản phí cần thanh toán qua hệ thống thanh toán để khách hàng thanh toán các khoản phí. Sau đó hệ thống thanh toán sẽ gửi về thông tin xác thực người dùng đã thanh toán phí dịch vụ cho hệ thống chính để mở các dịch vụ cho người dùng.
* Giữa My Meetings (System) với Email Service (System): Hệ thống chính sẽ gửi yêu cầu xác thực cho người dùng thông qua dịch vụ Email.
* Giữa Adminstrator (Person) với My Meetings (System): Quản lý có thể thực hiện các thao tác điều khiển những cuộc họp và quản lý các thành viên thông qua quyền quản lý.
* Giữa Adminstator (Person) với Email Service (System): Người quản lý sẽ nhận các mail yêu cầu đăng ký thành viên của người dùng thông qua dịch vụ Email.

Legend – Ký hiệu:

* Person – người dùng: Màu xanh dương đậm
* System – hệ thống: Màu xanh dương nhạt
* External person – Người dùng bên ngoài hệ thống: Màu xám đậm
* External system – Hệ thống ngoại: Màu xám nhạt

## Bài 5

File Cau5.drawio

A diagram of a system

Description automatically generated

C2 – Container Context, giải thích:

* Khung viền nét đứt: Ký hiệu cho giới hạn của hệ thống cần được biểu diễn (Ở đây là hệ thống My Meetings)
* Hệ thống My Meetings khi được zoom in bao gồm có các container như sau:
  + SPA – Single-Page Application: là trang giao diện (GUI) cho phần mềm web sử dụng công nghệ ReactJS để xây dựng nên. Cả 2 người dùng (Member, Adminstrator) đều sử dụng SPA thông qua đường dẫn HTTP.
  + My Meetings API: là phần backend của hệ thống SPA gồm những API được gọi đến để chạy trang SPA thông qua sử dụng công nghệ .NET core.
  + Database: là kho dữ liệu dùng để lưu trữ thông tin các cuộc họp, thông tin của tất cả thành viên có tham gia phần mềm và thông tin các nhóm của thành viên. Được đọc và ghi qua API bằng ngôn ngữ SQL.
* Email System: là hệ thống dịch vụ bên thứ ba giao tiếp với hệ thống chính qua API bằng các giao thức SMTP dùng để gửi mail.
* Payment Gateway: là hệ thống thanh toán bên thứ ba giao tiếp với hệ thống chính qua API bằng các giao thức HTTP để uỷ quyền việc thanh toán cho người dùng.

## Bài 6

File BaiTap3\_6.drawio

A diagram of a diagram

Description automatically generated

Tổng Quan Về My Meetings System: Hãy tưởng tượng My Meetings System như một "trung tâm tổ chức cuộc họp" thông minh. Nó giống như một văn phòng lớn với nhiều bộ phận khác nhau (như đăng ký, quản lý người dùng, lịch họp, v.v.).hệ thống này gồm có:

* Một giao diện đẹp mắt (SPA) để người dùng tương tác.
* API điều phối mọi thứ(meeting API).
* Nhiều "bộ phận" (Registrations, User Access, Meetings, Administration, Payments) làm việc riêng nhưng kết nối qua cơ sở dữ liệu và Event Bus.

Các Bộ Phận (Components) và Cơ Sở Dữ Liệu: Hệ thống có 5 bộ phận chính, mỗi bộ phận quản lý một loại dữ liệu riêng trong cơ sở dữ liệu (database) với các "kho" (schema) khác nhau. Chúng làm việc bằng cách lưu/truy xuất (store/retrieve) dữ liệu.

1. Registrations (Đăng Ký): Quản lý việc bạn đăng ký vào cuộc họp. Nó lưu thông tin như tên, email, cuộc họp bạn tham gia, và trạng thái (chờ, xác nhận, hủy).
2. User Access (Quyền Truy Cập): Quản lý thông tin người dùng và quyền truy cập. Nó giống như bộ phận nhân sự, kiểm tra bạn là ai (đăng nhập) và bạn được làm gì (xem hay quản lý).
3. Meetings (Lịch Họp): Quản lý thông tin về các cuộc họp, như thời gian, địa điểm, và danh sách tham gia. Nó giống như lịch trình của văn phòng.
4. Administration (Quản Trị): Hỗ trợ quản trị viên (admin) quản lý hệ thống, như thay đổi cấu hình hoặc xem báo cáo. Nó giống như bộ phận IT.
5. Payments (Thanh Toán): Xử lý thanh toán cho các cuộc họp có phí. Nó giống như quầy thu ngân, kết nối với ngân hàng hoặc ví điện tử.

Events Bus Là Gì?

Events Bus (hay còn gọi là Event Bus, Message Bus, hoặc Event Broker) là một thành phần trung gian (middleware) trong kiến trúc phần mềm, được thiết kế để quản lý và phân phối các sự kiện (events) hoặc thông điệp (messages) giữa các thành phần khác nhau trong hệ thống. Nó đóng vai trò như một "trạm trung chuyển" (hub) nơi các thành phần có thể xuất bản (publish) sự kiện và đăng ký (subscribe) để nhận các sự kiện mà họ quan tâm.

Định nghĩa chi tiết:

* Events Bus nhận các sự kiện từ các publishers (nhà xuất bản), chẳng hạn như khi một người dùng đăng ký một cuộc họp hoặc thanh toán hoàn tất.
* Nó phân phối (dispatch) các sự kiện này đến các subscribers (người đăng ký) dựa trên các chủ đề (topics) hoặc loại sự kiện (event types) mà họ đã đăng ký.
* Trong hệ thống này, nó là in-memory, nghĩa là tin nhắn được lưu tạm trong bộ nhớ máy chủ.

## Bài 7

File Baitap3\_7.drawio

A screenshot of a diagram

Description automatically generated

## Bài 8

File Lab3\_8.drawio

A diagram of a process

Description automatically generated

# Bài tập vận dụng

## Level 1 system context

A diagram of a company

Description automatically generated with medium confidence

Giải thích:

* Người đọc tương tác với hệ thống với các chức năng như tìm kiếm, mượn trả, gia hạn thành viên,...
* Admin có thể quản lý kho sách, xem yêu cầu mượn trả trực tiếp thông qua hệ thống.
* Khi Người đọc muốn mượn trả và được Admin duyệt -> hệ thống sẽ gửi phần phí và phương thức thanh toán qua hệ thống thanh toán (external system) và Người đọc sẽ thanh toán thông qua hệ thống này

## Level 2 container

A diagram of a group of people

Description automatically generated

## Level 3 component

A diagram of a network

Description automatically generated

## Level 4 code

A diagram of a computer

Description automatically generated