**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



**HỆ THỐNG QUẢN LÝ Lớp HỌC TÍN CHỈ**

***Nhóm 01 - Sinh viên thực hiện***

1. Võ Tấn Đạt –
2. Trần Thanh Đại – 20077931

**Giảng viên hướng dẫn: TS. Võ Văn Hải**

# Phần 1: Mô tả yêu cầu bài toán

Một trường đại học cần quản lý hệ thống đăng ký học phần của sinh viên theo theo cơ chế tín chỉ. Các sinh viên sẽ được biên chế ở các khoa theo mỗi ngành học của mình đăng ký.

Trong mỗi học kỳ, nhà trường sẽ chọn một số môn học của từng ngành học cho sinh viên đăng ký học. Sinh viên đăng ký theo học vào một lớp nhất định (lớp tín chỉ) với số sinh viên được đăng ký tối đa được qui định cho từng lớp.

Môn học được mở có thể là môn học tự chọn

Mỗi môn học có một hoặc nhiều môn học tiên quyết. Sinh viên sẽ không được đăng ký môn học mà có môn tiên quyết mình chưa học.

Mỗi học kỳ sinh viên được đăng ký tối đa 30 tín chỉ. Nếu quá số này, hệ thống sẽ không cho đăng ký.

Để phòng ngừa trường hợp sinh viên đăng ký xong rồi hủy bỏ, nhà trường yêu cầu sinh viên phải xác nhận trước khi đăng ký.

Vào ngày mở đăng ký, sinh viên sẽ đăng nhập vào hệ thống và sẽ nhìn thấy danh sách các môn học mà mình có khả năng đăng ký. Sinh viên chọn các môn học và tiến hành đăng ký. Trường hợp các lớp đã đầy, sinh viên sẽ được đưa vào một danh sách dự bị để nhà trường cân nhắc có mở thêm lớp hay không. Nếu không mở thêm lớp, sinh viên sẽ bị hủy đăng ký môn đó.

Sau khi đăng ký thành công, một email thông báo sẽ được gửi cho sinh viên xác nhận việc đăng ký và nhận quyết định đóng học phí.

Ngoài ra, hệ thống còn có khả năng cho phép sinh viên xem thông tin học tập của mình (số tín chỉ đã đạt, số môn đã học, điểm môn học, điểm trung bình tích lũy…), thời khóa biểu theo tuần và các tiện ích khác.

Sau khi số tín chỉ đã đạt theo từng ngành, sinh viên có quyền đăng ký xét tốt nghiệp. Nếu mọi tiêu chuẩn đều thỏa mãn, sinh viên sẽ được cấp bằng tốt nghiệp và sẽ được đưa vào danh sách các cựu sinh viên. Thông tin về bằng cấp sẽ được công khai trên trang web của nhà trường. Thông tin của cựu sinh viên sẽ được lưu giữ để theo dõi quá trình làm việc (nếu sinh viên đồng ý), làm các cuộc survey, cũng như nhiều hoạt động khác.

# Phần 2: Lựa chọn kiến trúc thực hiện bài toán

**2.1 Chọn kiến trúc**

***Kiến trúc được lựa chọn:*** Event-driven Architecture (Kiến trúc hướng sự kiện)

**2.2 Lý do chọn kiến trúc**

Việc lựa chọn kiến trúc event-driven có vài lý do nhự sau:

***2.2.1. Khả năng mở rộng và phản ứng nhanh***

**Mở rộng hệ thống dễ dàng:** Khi số lượng sinh viên tăng hoặc yêu cầu hệ thống thay đổi, kiến trúc event-driven cho phép thêm mới các dịch vụ hoặc thành phần mà không ảnh hưởng đến toàn bộ hệ thống.

**Phản ứng nhanh với sự kiện:** Sinh viên có thể đăng ký môn học, nhận thông báo và xác nhận đăng ký theo thời gian thực. Hệ thống có thể xử lý nhiều yêu cầu đồng thời mà không làm giảm hiệu suất.

***2.2.2. Quản lý sự kiện phức tạp***

**Xử lý đăng ký:** Khi sinh viên đăng ký môn học, hệ thống sẽ phát sự kiện "đăng ký môn học" và các dịch vụ khác nhau (xác nhận đăng ký, kiểm tra điều kiện tiên quyết, kiểm tra số tín chỉ, gửi email) sẽ xử lý sự kiện này một cách độc lập nhưng đồng bộ.

**Danh sách dự bị:** Khi một lớp đầy, sự kiện "lớp đầy" được phát, và hệ thống có thể đưa sinh viên vào danh sách dự bị hoặc quyết định mở thêm lớp học dựa trên số lượng sinh viên trong danh sách dự bị.

***2.2.3. Tính linh hoạt và khả năng mở rộng***

**Thêm tính năng mới dễ dàng:** Khi cần thêm tính năng mới (ví dụ, tích hợp thêm hệ thống thanh toán), chỉ cần thêm dịch vụ mới để lắng nghe và xử lý sự kiện liên quan mà không cần thay đổi cấu trúc hiện tại.

**Dễ dàng tích hợp với hệ thống khác:** Hệ thống có thể dễ dàng tích hợp với các dịch vụ bên ngoài như hệ thống thanh toán, dịch vụ email, hệ thống quản lý học tập khác.

***2.2.4. Quản lý lỗi và tính chịu lỗi cao***

**Khả năng phục hồi tự động:** Khi một dịch vụ gặp lỗi, hệ thống có thể phát hiện và xử lý lỗi mà không ảnh hưởng đến toàn bộ hệ thống. Các sự kiện có thể được lưu trữ và xử lý lại sau khi sự cố được khắc phục.

**Đảm bảo tính toàn vẹn dữ liệu:** Sự kiện được ghi nhận và lưu trữ, đảm bảo rằng không có sự kiện nào bị mất, giúp duy trì tính toàn vẹn của dữ liệu trong quá trình xử lý.

***2.2.5. Quản lý nghiệp vụ phức tạp***

**Điều kiện tiên quyết:** Khi sinh viên đăng ký môn học, hệ thống sẽ phát sự kiện "kiểm tra điều kiện tiên quyết". Dịch vụ kiểm tra sẽ lắng nghe sự kiện này và phản hồi về việc sinh viên có đủ điều kiện hay không.

**Giới hạn tín chỉ:** Khi sinh viên đăng ký môn học, hệ thống sẽ phát sự kiện "kiểm tra giới hạn tín chỉ". Dịch vụ kiểm tra sẽ tính toán số tín chỉ hiện tại và phản hồi lại về việc đăng ký có hợp lệ hay không.

***2.2.6. Cải thiện trải nghiệm người dùng***

**Thông báo thời gian thực:** Khi sinh viên đăng ký thành công hoặc bị hủy đăng ký, hệ thống sẽ ngay lập tức gửi thông báo qua email hoặc ứng dụng di động, giúp sinh viên cập nhật thông tin nhanh chóng.

**Giao diện người dùng động:** Sinh viên có thể thấy ngay lập tức các môn học khả dụng và trạng thái đăng ký, tăng cường trải nghiệm người dùng.

***2.2.7. Quản lý thông tin sinh viên và tốt nghiệp***

**Quản lý thông tin học tập:** Hệ thống phát sự kiện "cập nhật thông tin học tập" khi có thay đổi về số tín chỉ, điểm số, hoặc trạng thái học tập, giúp duy trì thông tin chính xác và cập nhật.

**Xét tốt nghiệp:** Khi sinh viên đạt đủ số tín chỉ và yêu cầu xét tốt nghiệp, hệ thống phát sự kiện "yêu cầu xét tốt nghiệp" và các dịch vụ liên quan sẽ xử lý yêu cầu này.

**2.3 Kết luận**

**Tóm lại:** Kiến trúc event-driven phù hợp cho hệ thống quản lý đăng ký học phần của sinh viên vì nó mang lại khả năng mở rộng, tính linh hoạt, tính chịu lỗi cao và cải thiện trải nghiệm người dùng. Việc xử lý các nghiệp vụ phức tạp như kiểm tra điều kiện tiên quyết, quản lý danh sách dự bị, và gửi thông báo thời gian thực sẽ trở nên dễ dàng và hiệu quả hơn trong mô hình này.

# Phần 3: Cơ sở lý thuyết

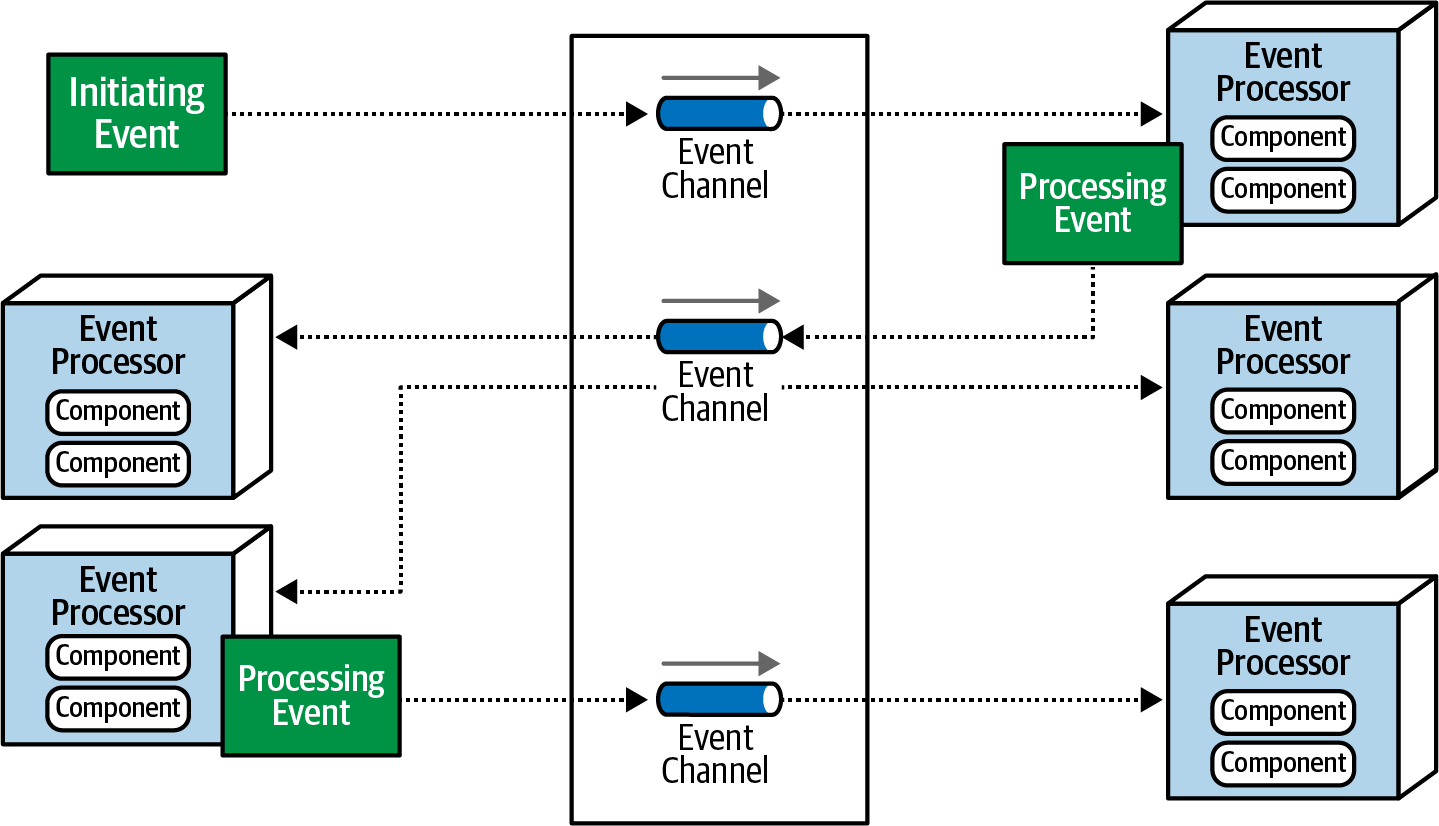
**3.1 Kiến trúc event-driven**

**- Sự kiện (Event)**

*Định nghĩa:* Một sự kiện là một hành động hoặc trạng thái đã xảy ra trong hệ thống, như một người dùng nhấp chuột, một giao dịch tài chính hoàn tất, hoặc một cảm biến phát hiện một thay đổi.

*Thuộc tính:* Sự kiện thường có các thuộc tính như tên sự kiện, thời gian xảy ra, dữ liệu liên quan đến sự kiện đó.

**- Thành phần chính của EDA**



*Event Producers:* Là các thực thể tạo ra các sự kiện. Ví dụ, một ứng dụng web có thể tạo ra một sự kiện khi người dùng đăng nhập.

Event Consumers: Là các thực thể nhận và xử lý các sự kiện. Ví dụ, một dịch vụ email có thể nhận sự kiện và gửi email xác nhận.

*Event Channels:* Là các đường dẫn thông qua đó các sự kiện được truyền từ nhà sản xuất tới người tiêu thụ. Các kênh sự kiện có thể là các hàng đợi tin nhắn, dịch vụ pub/sub, hoặc các hệ thống streaming.

*Event Processors:* Là các thực thể xử lý các sự kiện, bao gồm các bộ xử lý đơn giản hoặc các hệ thống phức tạp có thể thay đổi trạng thái của hệ thống hoặc kích hoạt các sự kiện khác.

**- Cơ chế hoạt động**

*Phát hiện sự kiện:* Nhà sản xuất phát hiện một sự kiện và gửi thông báo về sự kiện đó tới một hoặc nhiều kênh sự kiện.

*Phân phối sự kiện:* Kênh sự kiện chịu trách nhiệm phân phối sự kiện tới các nhà tiêu thụ đăng ký quan tâm.

*Xử lý sự kiện:* Nhà tiêu thụ nhận sự kiện và thực hiện hành động dựa trên sự kiện đó, có thể bao gồm việc cập nhật trạng thái hệ thống, gửi phản hồi, hoặc phát ra các sự kiện mới.

**3.2 Apache kafka**

**3.3 reactjs**

**3.4 nodejs**

**Phần 4: Cài đặt**

**4.1 Mô hình kiến trúc**

**4.2 Mô hình lớp**

**4.3**

# TÀI LIỆU THAM KHẢO

Các tài liệu Tiếng Anh

Các tài liệu từ Internet