# CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI

## 1.1 Thực trạng

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ trong lĩnh vực giáo dục, việc ứng dụng công nghệ thông tin vào công tác quản lý đào tạo là một xu hướng tất yếu. Tuy nhiên, tại nhiều trường đại học ở Việt Nam, trong đó có Trường Đại học Thủy Lợi, quá trình đăng ký học phần – một mắt xích quan trọng trong chu trình đào tạo – vẫn còn nhiều tồn tại, gây ảnh hưởng không nhỏ đến trải nghiệm học tập của sinh viên cũng như hiệu quả quản lý của nhà trường.

Hằng năm, vào mỗi đợt đăng ký học phần, hàng nghìn sinh viên cùng lúc truy cập hệ thống nhằm lựa chọn các môn học, lớp học phù hợp với kế hoạch học tập cá nhân. Tuy nhiên, do hệ thống hiện tại còn hạn chế về hiệu năng và khả năng mở rộng, hiện tượng nghẽn mạng, treo hệ thống, mất kết nối hoặc xử lý chậm thường xuyên xảy ra. Điều này không chỉ gây bức xúc, hoang mang cho sinh viên mà còn tạo áp lực rất lớn cho đội ngũ kỹ thuật và phòng Đào tạo của nhà trường.

Ngoài ra, giao diện hệ thống còn thiếu thân thiện, khó thao tác đối với sinh viên mới, không cung cấp đầy đủ thông tin như: số lượng sinh viên còn có thể đăng ký trong lớp, lịch học chi tiết, cảnh báo trùng lịch, hay gợi ý đăng ký tối ưu. Các chức năng quan trọng như thay đổi nguyện vọng, huỷ học phần, đăng ký lại, hoặc thống kê thời khóa biểu cá nhân thường được thực hiện một cách thủ công hoặc qua nhiều bước phức tạp.

Đối với cán bộ quản lý, việc theo dõi tiến độ đăng ký học, kiểm soát số lượng lớp, phân công giảng viên, sắp xếp phòng học… vẫn phụ thuộc nhiều vào thao tác thủ công và các hệ thống rời rạc. Việc tổng hợp dữ liệu, xuất báo cáo hay đưa ra quyết định điều chỉnh thường chậm trễ và thiếu chính xác, đặc biệt là trong những giai đoạn cao điểm.

Thực trạng này không chỉ ảnh hưởng đến chất lượng phục vụ sinh viên mà còn làm giảm hiệu suất điều hành và khả năng phản ứng nhanh của hệ thống đào tạo khi có thay đổi. Trong bối cảnh sinh viên ngày càng đòi hỏi tính minh bạch, công bằng và cá nhân hóa cao trong lộ trình học tập, thì hệ thống đăng ký học truyền thống hiện tại rõ ràng không còn đáp ứng được yêu cầu thực tiễn.

Chính vì vậy, nhu cầu cấp thiết đặt ra là phải xây dựng một hệ thống đăng ký học hiện đại, linh hoạt, dễ mở rộng, có khả năng xử lý đồng thời hàng nghìn yêu cầu, đồng thời cung cấp các tính năng thông minh như kiểm tra trùng lịch học, tối ưu thời khóa biểu, theo dõi tình trạng lớp học theo thời gian thực, và hỗ trợ ra quyết định cho cả sinh viên lẫn nhà quản lý. Đây không chỉ là một bài toán về kỹ thuật, mà còn là thách thức trong việc nâng cao chất lượng đào tạo, hướng tới một mô hình quản trị đại học thông minh và hiệu quả trong thời đại số hóa.

## 1.2 Lý do chọn đề tài

Là một sinh viên chuyên ngành Công nghệ thông tin, em nhận thức sâu sắc rằng công nghệ không chỉ đơn thuần là công cụ hỗ trợ, mà còn là yếu tố quyết định trong việc cải tiến và tái cấu trúc các quy trình quản lý giáo dục hiện nay. Trong quá trình học tập và tiếp xúc thực tế với hệ thống đào tạo tại Trường Đại học Thủy Lợi, em nhận thấy rằng quá trình đăng ký học phần – vốn là một nghiệp vụ diễn ra định kỳ và có tác động trực tiếp đến toàn bộ sinh viên – vẫn tồn tại nhiều bất cập cả về kỹ thuật lẫn trải nghiệm người dùng.

Bên cạnh các vấn đề như nghẽn hệ thống, thiếu cảnh báo trùng lịch, thao tác không trực quan hoặc giao diện chưa tối ưu cho thiết bị di động, hệ thống hiện tại còn chưa được thiết kế theo hướng mở rộng linh hoạt (scalable) hoặc hỗ trợ xử lý đồng thời hiệu quả. Điều này đặc biệt nghiêm trọng trong bối cảnh số lượng sinh viên đăng ký tăng cao theo từng năm học, và nhu cầu cá nhân hóa lịch học ngày càng lớn.

Việc nghiên cứu, thiết kế và triển khai một hệ thống đăng ký học hiện đại, áp dụng các mô hình kiến trúc tiên tiến như Microservices, kết hợp với CI/CD, cơ sở dữ liệu lai (hybrid database), triển khai trên nền tảng đám mây (Docker, Kubernetes) và giao diện thân thiện (React, TypeScript) không chỉ là một hướng đi phù hợp với xu thế công nghệ, mà còn giúp sinh viên tiếp cận sâu hơn với các bài toán thực tiễn và kỹ năng phát triển hệ thống phần mềm quy mô lớn.

Em lựa chọn đề tài này với mong muốn:

* Góp phần cải thiện một trong những quy trình quan trọng bậc nhất trong công tác đào tạo của nhà trường – đăng ký học phần.
* Vận dụng toàn diện kiến thức đã học trong suốt chương trình đào tạo để xây dựng một sản phẩm thực tế, có khả năng triển khai ứng dụng thật sự.
* Tiếp cận với những công nghệ hiện đại đang được sử dụng phổ biến trong các hệ thống doanh nghiệp lớn, từ đó nâng cao năng lực chuyên môn và chuẩn bị tốt hơn cho công việc sau khi tốt nghiệp.

Không chỉ dừng lại ở việc xây dựng một hệ thống kỹ thuật, em mong muốn đề tài này có thể trở thành một bước khởi đầu cho việc chuẩn hóa và hiện đại hóa quy trình đào tạo tại Trường Đại học Thủy Lợi nói riêng, và làm nền tảng tham khảo cho các cơ sở giáo dục khác nói chung trong công cuộc chuyển đổi số toàn diện ngành giáo dục đại học.

## 1.3 Mục tiêu

Mục tiêu tổng thể của đề tài là thiết kế và xây dựng một hệ thống đăng ký học trực tuyến hiện đại, có khả năng hỗ trợ toàn diện quy trình đăng ký học phần cho sinh viên Trường Đại học Thủy Lợi, đồng thời cung cấp các công cụ giám sát, điều phối và phân tích dữ liệu học tập cho cán bộ quản lý đào tạo. Hệ thống hướng đến việc thay thế phương thức đăng ký học truyền thống vốn còn nhiều hạn chế, bằng một nền tảng công nghệ thông minh, linh hoạt và dễ mở rộng.

Cụ thể, đề tài đặt ra các mục tiêu sau:

1. Tự động hóa quy trình đăng ký học phần

* Cho phép sinh viên đăng nhập, tìm kiếm, lựa chọn và đăng ký học phần nhanh chóng theo thời khóa biểu cá nhân.
* Tự động kiểm tra xung đột lịch học, hạn chế tình trạng trùng giờ, sai sót khi đăng ký.
* Cung cấp giao diện trực quan giúp sinh viên dễ dàng theo dõi danh sách học phần đã đăng ký, lịch học cá nhân, trạng thái lớp học (đã đầy, còn chỗ), và điều kiện tiên quyết của học phần.

2. Tối ưu hóa công tác quản lý đào tạo

* Hỗ trợ cán bộ đào tạo trong việc quản lý danh sách lớp học, số lượng sinh viên, phân bổ giảng viên – phòng học theo năng lực và nhu cầu thực tế.
* Cung cấp hệ thống thống kê, báo cáo theo thời gian thực về tình hình đăng ký học của sinh viên theo môn học, lớp, khoa, khóa,...
* Cho phép xuất dữ liệu phục vụ phân tích, dự báo và ra quyết định quản lý.

3. Đảm bảo hiệu năng và khả năng mở rộng

* Ứng dụng kiến trúc Microservices để chia nhỏ hệ thống thành các module độc lập, dễ dàng phát triển, bảo trì và mở rộng trong tương lai.
* Sử dụng công nghệ container (Docker) và nền tảng quản lý triển khai (Kubernetes) để đảm bảo khả năng triển khai linh hoạt và độ tin cậy cao.
* Tối ưu hiệu năng để hệ thống có thể xử lý đồng thời hàng nghìn yêu cầu mà không bị gián đoạn.

4. Cung cấp trải nghiệm người dùng thân thiện

* Thiết kế giao diện web responsive, dễ sử dụng trên nhiều thiết bị (PC, máy tính bảng, điện thoại).
* Tích hợp các tiện ích như tìm kiếm nhanh, bộ lọc môn học, gợi ý đăng ký theo chương trình đào tạo, cảnh báo khi lớp đầy hoặc lịch học bị xung đột.

5. Đảm bảo an toàn và bảo mật dữ liệu

* Áp dụng cơ chế xác thực người dùng thông qua tài khoản Outlook sinh viên nhằm bảo đảm tính chính danh.
* Bảo vệ dữ liệu cá nhân và thông tin học phần bằng các chuẩn mã hóa và phân quyền truy cập hợp lý.

6. Hướng đến triển khai thực tế

* Xây dựng một hệ thống hoàn chỉnh có thể triển khai thử nghiệm hoặc tích hợp với hệ thống quản lý đào tạo hiện tại của nhà trường.
* Có khả năng mở rộng để tích hợp với các chức năng liên quan như quản lý điểm, lịch thi, đăng ký tín chỉ tốt nghiệp,...

## 1.4 Cơ sở lý thuyết

# CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ

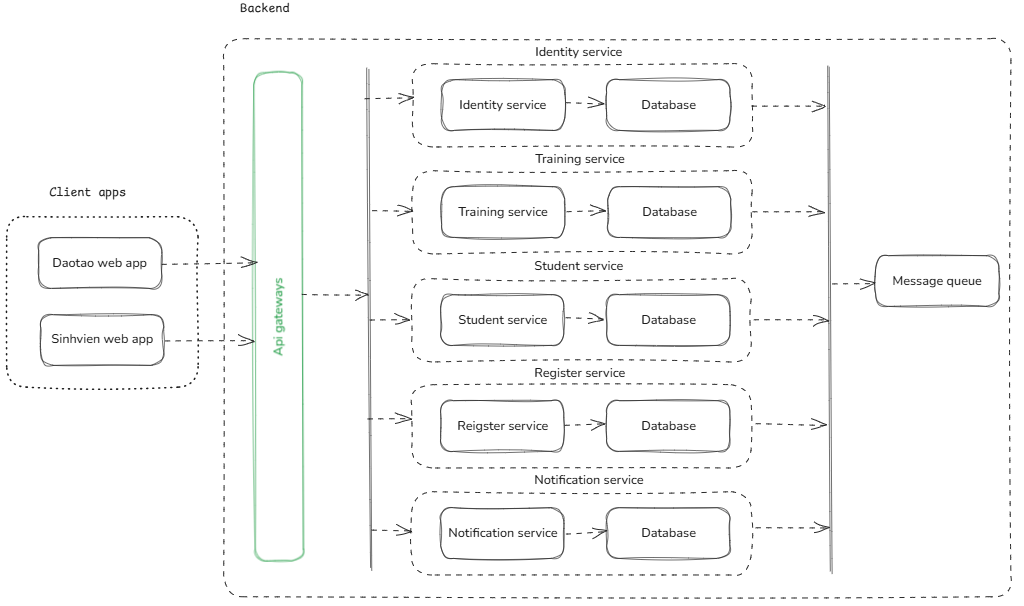
## 2.1 Thu thập yêu cầu

2.3 Đặc tả use case

2.6. Thiết kế api

# CHƯƠNG 3 TRIỂN KHAI

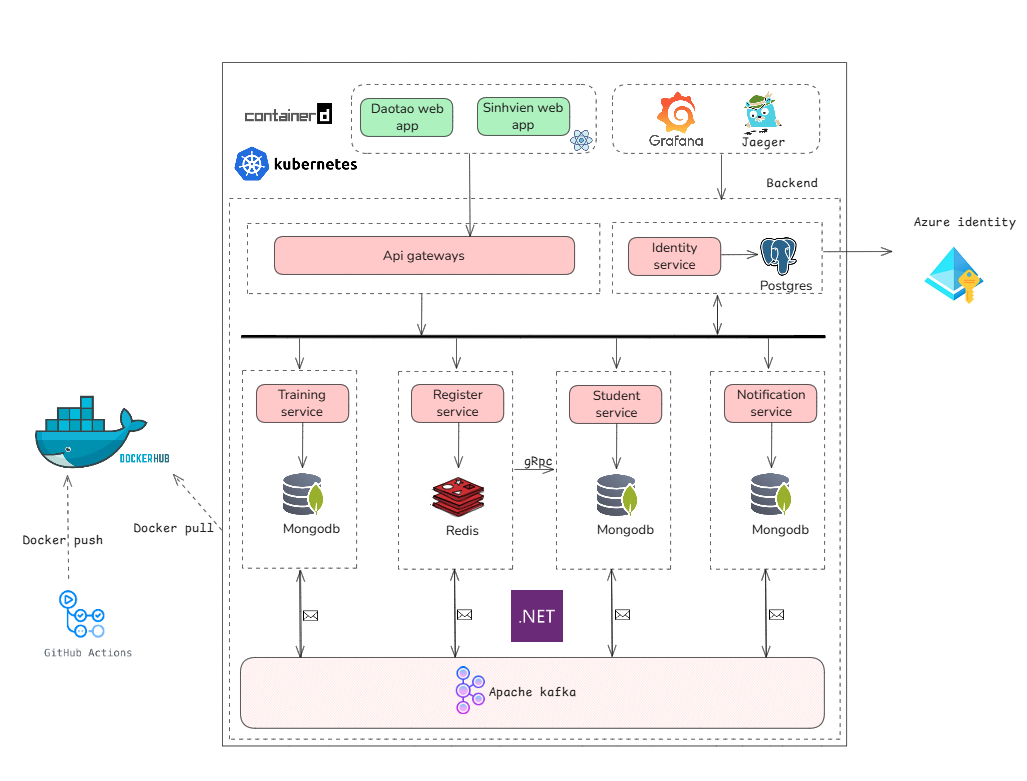
## 3.1 Sơ đồ thành phần



Chi tiết các thành phần:

* **Client apps:** 
  + **Daotao web app:** Web app dành cho PĐT và GV.
  + **Sinhvien web app:** Web app dành cho sinh viên.
* **API Gateways:**  đóng vai trò điều hướng các API request từ người dùng đến các service.
* **Identity Service:** Đóng vai trò xác thực và phân quyền người dùng trên hệ thống.
* **Training Service:** Đóng vài trò xử lý thông tin môn học và cấu hình lớp học phần môn học.
* **Student Service**: Đóng vai trò xử lý các tác vụ của sinh viên.
* **Register Service**: Đóng vai trò xử lý tác vụ đăng ký học.
* **Notification Service**: Đóng vai trò xử lý các tác vụ thông báo.

## 3.2 Sơ đồ triển khai



* Hạ tầng triển khai:
  + **Kubernetes + containerd**: Toàn bộ hệ thống chạy trên Kubernetes với containerd là container runtime.
  + **DockerHub**: Hình ảnh container được đẩy lên từ GitHub Actions và được Kubernetes pull về từ DockerHub để triển khai.
* CI/CD Pipeline:
  + **GitHub Actions**: Tự động build và đẩy image Docker lên DockerHub.
  + **Docker push/pull**: CI/CD đẩy hình ảnh lên DockerHub và Kubernetes sẽ pull về khi triển khai.
* Frontend:
  + **Daotao web app** và **Sinhvien web app**: Giao diện người dùng, chạy bên ngoài và giao tiếp với backend thông qua API Gateway.
* API Gateway và Identity:
  + **API Gateway**: Là điểm vào của tất cả các request, chuyển tiếp đến các dịch vụ tương ứng.
  + **Identity Service**: Dịch vụ xác thực, kết nối tới **PostgreSQL** để lưu trữ thông tin người dùng, đồng thời tích hợp với **Azure Identity** (AAD - Azure Active Directory).
* Các dịch vụ backend:
  + **Training Service**: Quản lý các tác vụ liên quan đến đào tạo, sử dụng **MongoDB** để lưu trữ dữ liệu.
  + **Register Service**: Xử lý các yêu cầu đăng ký, sử dụng **Redis** để lưu trữ dữ liệu tạm thời và hỗ trợ giao tiếp gRPC với các dịch vụ khác.
  + **Student Service**: Quản lý thông tin sinh viên, lưu trữ trên **MongoDB**.
  + **Notification Service**: Gửi thông báo, cũng sử dụng **MongoDB** để lưu trữ dữ liệu thông báo.
  + Các dịch vụ này giao tiếp với nhau thông qua **Apache Kafka**, một hệ thống hàng đợi tin nhắn, để xử lý các sự kiện bất đồng bộ.
* Quan sát hệ thống:
  + **Grafana**: Theo dõi metrics, giám sát hiệu năng hệ thống.
  + **Jaeger**: Theo dõi trace/phân tích đường đi của request trong hệ thống (distributed tracing).