

**ĐẠI** **HỌC** **PHENIKAA**

**TRƯỜNG CÔNG NGỆ THÔNG TIN**

**PHÂN TÍCH THIẾT KẾ PHẦN MỀM**

**(N05)**

**Nhóm 13**

**Dự án: Hệ thống Quản lý đăng kí học theo tín chỉ**

|  |  |
| --- | --- |
| Đỗ Hoài Nam | [22014002@st.phenikaa-uni.edu.vn](mailto:22010392@st.phenikaa-uni.edu.vn) |
| Nguyễn Văn Minh | [22010487@st.phenikaa-uni.edu.vn](mailto:22010508@st.phenikaa-uni.edu.vn) |

**Giảng viên hướng dẫn:** ThS. Vũ Quang Dũng, ThS. Nguyễn Xuân Quế

**Lời cảm ơn**

Đối với bất kì bất kì sinh viên nào thì bài tập lớn kết thúc môn luôn là thành quả quan trọng của một môn học. Và để bài tập lớn này được đạt kết quả tốt như hiện nay chúng em đã nhận được rất nhiều sự hỗ trợ, hướng dẫn vô cùng tận tâm và nhiệt huyết của ThS. Vũ Quang Dũng, ThS. Nguyễn Xuân Quế. Xuất phát từ sự chân thành, chúng em xin bày tỏ sự biết ơn sâu sắc nhất đến các thầy.

Qua những bài giảng vô cùng thú vị ở lớp, những tiết học cuốn hút, các thầy đã không ngừng tạo những điều kiện tốt để chúng em có được môi trường học tập tốt và là người trang bị cho chúng em tất cả những kiến thức, kỹ năng cần có để chúng em có thể hoàn thành tốt đồ án này.

Trong quá trình học tập và nghiên cứu, do bản thân chúng em vẫn còn chưa vững kiến thức về chuyên ngành và kinh nghiệm thực tế nên có nhiều điều chúng em còn thiếu sót và cần học hỏi nhiều hơn nữa.

Chúng em rất cảm ơn các thầy vẫn tận tâm, tỉ mỉ, dành nhiều thời gian để hướng dẫn, chỉ bảo cho chúng em, giúp chúng em ngày một hoàn thiện bản thân mình hơn. Chúng em kính chúc các thầy luôn luôn mạnh khỏe.

Chúng em xin chân thành cảm ơn!

**Chương 1. Giới thiệu đề tài**

**1.1. Mục tiêu**

Bản phân tích và thiết kế này cung cấp mô tả chi tiết về **Hệ thống Quản lý Đăng ký Tín chỉ Học phần** (sau đây gọi tắt là “Hệ thống”), bao gồm mục tiêu, phạm vi, nhóm người dùng chính, các chức năng cốt lõi và các tiêu chí chất lượng cần đạt. Hệ thống hướng đến việc chuẩn hóa và tự động hóa toàn bộ quy trình đăng ký học phần theo học chế tín chỉ, đảm bảo tính công bằng, minh bạch, chính xác và trải nghiệm sử dụng thuận tiện cho sinh viên, giảng viên và bộ phận quản lý đào tạo.

Trong bối cảnh số hóa giáo dục diễn ra mạnh mẽ, số lượng môn học, lớp học phần, ràng buộc tiên quyết – song hành – tương đương, cùng các chính sách học vụ ngày càng đa dạng khiến việc đăng ký thủ công, phân lớp và kiểm soát sĩ số trở nên phức tạp, dễ sai sót và tốn nguồn lực. Hệ thống được xây dựng nhằm giải quyết các vấn đề đó, hỗ trợ **đăng ký – kiểm tra điều kiện – xếp lớp – theo dõi tiến độ học tập** một cách chính xác theo thời gian thực, đồng thời giảm tải cho phòng đào tạo.

Với mục tiêu trở thành nền tảng đăng ký tín chỉ trực tuyến **hữu ích – tiện lợi – mở rộng linh hoạt**, Hệ thống cung cấp giao diện thân thiện, quy trình đơn giản và phản hồi tức thì để sinh viên dễ dàng tra cứu chương trình đào tạo, lập kế hoạch học kỳ, đăng ký/hủy/muộn đăng ký theo quy định; giảng viên quản lý lớp học phần, sĩ số, lịch học – lịch thi; và phòng đào tạo kiểm soát năng lực phòng học, lịch biểu toàn trường, chỉ tiêu, chuẩn đầu ra cùng các báo cáo phục vụ ra quyết định.

**1.1.1. Mục tiêu tổng quát**

* **Chuẩn hóa quy trình** đăng ký tín chỉ theo quy chế học vụ và chương trình đào tạo của nhà trường.
* **Tự động hóa kiểm tra ràng buộc** (tiên quyết, học trước, song hành, tương đương, trùng lịch, giới hạn tín chỉ/học kỳ, trùng lớp, trùng ca thi…).
* **Tối ưu phân bổ nguồn lực** (chỉ tiêu lớp, phòng học, lịch giảng, lịch thi) và giảm thiểu tình trạng nghẽn hệ thống trong “giờ cao điểm”.
* **Minh bạch & thời gian thực**: cập nhật sĩ số, danh sách chờ (waitlist), trạng thái đăng ký, thông báo thay đổi tức thì.
* **Theo dõi tiến độ học tập** từng sinh viên so với khung chương trình, cảnh báo nợ môn/thiếu tín chỉ/vi phạm điều kiện.
* **Báo cáo – phân tích** nhiều chiều cho quản trị: tình hình đăng ký, tỷ lệ đầy lớp, nhu cầu mở thêm lớp, dự báo tải phòng học.

**1.1.2. Mục tiêu chức năng**

* **Quản lý danh mục đào tạo**
  + Chương trình/khóa học, học phần, phiên bản CTĐT theo niên khóa.
  + Điều kiện ràng buộc: tiên quyết, song hành, tương đương, tái đăng ký, khống chế số lần học/thi lại.
* **Kế hoạch & lịch biểu**
  + Kế hoạch mở lớp học phần theo học kỳ (giảng viên, thời khóa biểu, phòng học, chỉ tiêu, hình thức dạy-học).
  + Lịch thi giữa kỳ/cuối kỳ, ràng buộc trùng lịch thi và quy tắc ưu tiên.
* **Đăng ký học phần cho sinh viên**
  + Tra cứu CTĐT cá nhân hóa, lộ trình gợi ý, giỏ đăng ký (cart), khóa học yêu thích.
  + Đăng ký/hủy/đổi lớp, danh sách chờ tự động khi đầy lớp; áp dụng cửa sổ thời gian theo đợt/mốc ưu tiên.
  + Kiểm tra ràng buộc tức thì, giải thích lý do từ chối rõ ràng, gợi ý lớp thay thế.
* **Quản lý lớp học phần**
  + Theo dõi sĩ số, danh sách sinh viên, phân nhóm thực hành/thảo luận, điểm danh, xuất danh sách.
  + Điều chỉnh chỉ tiêu (theo phê duyệt), chuyển phòng/đổi lịch có thông báo đến người liên quan.
* **Theo dõi tiến độ & tư vấn học tập**
  + Dashboard tiến độ cá nhân so với chuẩn đầu ra, kiểm đếm tín chỉ tích lũy theo khối kiến thức.
  + Cảnh báo rủi ro (GPA thấp, nợ tiên quyết, vượt/thiếu tín chỉ học kỳ, thời gian tối đa).
* **Báo cáo – thống kê**
  + Tải hệ thống theo thời gian thực, tỷ lệ lấp đầy lớp, nhu cầu mở thêm lớp, phân tích xu hướng đăng ký.
  + Kết xuất báo cáo (PDF/Excel) cho phòng đào tạo, khoa, bộ môn.
* **Tích hợp – mở rộng**
  + Tích hợp cổng định danh/SSO, học phí & hóa đơn (liên kết công nợ học phí thì ghi nhận tín chỉ đăng ký), LMS, thư viện.
  + API nội bộ phục vụ đồng bộ dữ liệu và phân tích BI.

**1.1.3. Mục tiêu phi chức năng**

* **Hiệu năng & khả dụng cao:** chịu tải giờ cao điểm (hàng nghìn yêu cầu/giây), thời gian phản hồi < 2s cho thao tác đăng ký; uptime ≥ 99.9% trong mùa đăng ký.
* **Mở rộng tuyến tính:** kiến trúc có thể mở rộng theo chiều ngang (scale-out), cân bằng tải, cache chiến lược.
* **Tính nhất quán & toàn vẹn dữ liệu:** đảm bảo giao dịch đăng ký nguyên tử (ACID) để tránh “overbooking”.
* **Bảo mật & riêng tư:** tuân thủ bảo vệ dữ liệu cá nhân, phân quyền theo vai trò (SV/giảng viên/cán bộ), ghi nhật ký (audit log), mã hóa dữ liệu nhạy cảm.
* **Khả năng khôi phục & liên tục dịch vụ:** cơ chế sao lưu – phục hồi (RPO/RTO rõ ràng), kế hoạch DR site.
* **Khả dụng đa nền tảng:** web responsive, hỗ trợ thiết bị di động; khả năng thông báo đẩy/email/SMS.
* **Khả năng quan sát:** logging, metrics, tracing để theo dõi sự cố và tối ưu hiệu năng.

**1.2. Phạm vi dự án (Rút gọn)**

**1.2.1. Phạm vi chức năng:**

* **Danh mục đào tạo:** Quản lý CTĐT (phiên bản theo khóa), khối kiến thức, chuẩn đầu ra; học phần & nhóm tương đương; ràng buộc học vụ (tiên quyết/học trước/song hành/cấm trùng, theo ngành/khóa).
* **Kế hoạch & đăng ký:** Lập kế hoạch/mở lớp (chỉ tiêu, GV, lịch–phòng), lịch thi tránh trùng; quy trình đăng ký của SV với giỏ đăng ký/hủy/đổi, kiểm tra ràng buộc tức thì, danh sách chờ & giữ chỗ, cảnh báo trùng lịch/giới hạn tín chỉ.
* **Vận hành & giám sát:** Quản lý lớp (sĩ số, phân nhóm, thông báo thay đổi), theo dõi tiến độ tốt nghiệp, (tuỳ áp dụng) học phí/công nợ; báo cáo–thống kê–audit; quản trị hệ thống và cấu hình học vụ.

**1.2.2. Phạm vi dữ liệu**

Gồm người dùng (SV/GV/Cán bộ), CTĐT, học phần/nhóm tương đương, ràng buộc, lớp học phần, lịch học–thi, phòng, đăng ký, công nợ/học phí (nếu có), nhật ký. Bảo đảm toàn vẹn (khóa ngoại), phiên bản hóa CTĐT, lịch sử thay đổi, kiểm soát đồng thời. Lưu trữ dữ liệu cho các khóa đang học và tối thiểu N năm sau tốt nghiệp.

**1.2.4. Triển khai & môi trường**

DEV, UAT/Staging, PROD với dữ liệu ẩn danh; hạ tầng on-prem hoặc cloud, ưu tiên container, autoscale, HA; web responsive, có thể bổ sung app di động nhẹ; đảm bảo hiệu năng giờ cao điểm với CDN/cache đọc.

**1.2.5. Người dùng & bên liên quan**

Sinh viên theo học chế tín chỉ; giảng viên giảng dạy; cán bộ đào tạo/khoa/bộ môn lập kế hoạch, vận hành và báo cáo.

**1.3. Mô hình quy trình đã chọn và lý do**

**1.3.1. Mô hình quy trình**

Chọn mô hình lặp–tăng trưởng (Iterative & Incremental) theo Agile Scrum, có quản trị rủi ro kiểu Spiral nhẹ và kiểm thử theo V-Model trong từng vòng lặp. Cấu trúc chính:

* **Scrum** làm khung quản lý (kế hoạch, vai trò, nghi lễ, nhịp độ).
* **Phát hành tăng trưởng** theo từng **Increment** có thể triển khai thực tế.
* **Quản trị rủi ro** (hiệu năng giờ cao điểm, nhất quán giao dịch, tích hợp SSO/học phí) theo chu kỳ **Identify → Mitigate → Validate** trong mỗi sprint.
* **Kiểm thử sớm & liên tục**: mỗi hạng mục đều có kiểm thử đơn vị/tích hợp/phi chức năng kèm tiêu chí chấp nhận.

**1.3.2. Lý do lựa chọn**

* **Nhu cầu thay đổi thường xuyên** (quy chế học vụ, ràng buộc tiên quyết, cửa sổ đăng ký) ⇒ Agile thích ứng tốt, cấu hình hóa rule.
* **Ràng buộc kỹ thuật phức tạp** (ACID, chống overbooking, scale giờ cao điểm) ⇒ lặp–tăng trưởng giúp **thử–đo–học** sớm với dữ liệu giả lập, tránh dồn rủi ro về cuối.
* **Phạm vi rộng, nhiều bên liên quan** (SV, giảng viên, phòng đào tạo, tài vụ) ⇒ phát hành theo **module** để nhận phản hồi thật, giảm sai lệch kỳ vọng.
* **Yêu cầu phi chức năng nghiêm ngặt** (99.9% uptime mùa đăng ký, TTR<2s) ⇒ kiểm thử hiệu năng/độ tin cậy **lặp lại liên tục** trước mỗi mốc phát hành.
* **Cửa sổ thời gian gấp** (chuẩn bị trước mỗi học kỳ) ⇒ Increment có giá trị sử dụng sớm, tránh “big-bang”.

**1.4. Các định nghĩa và từ viết tắt**

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuật ngữ** | **Viết đầy đủ** | **Mô tả ngắn** | **Dịch nghĩa / Giải thích** |
| **2FA** | Two-Factor Authentication | Xác thực đăng nhập cần 2 yếu tố (mật khẩu + mã OTP, v.v.) | Xác thực hai lớp |
| **5xx** | HTTP 5xx Status Codes | Mã lỗi máy chủ trong HTTP (500–599) | Lỗi phía máy chủ |
| **95th percentile** | — | Ngưỡng hiệu năng tính theo phân vị 95% | 95% yêu cầu nhanh hơn mốc này |
| **AC** | Acceptance Criteria | Tiêu chí chấp nhận cho mỗi chức năng/use case | Tiêu chí chấp nhận |
| **ACID** | Atomicity, Consistency, Isolation, Durability | Tính chất giao dịch bảo đảm toàn vẹn dữ liệu | Giao dịch nguyên tử/nhất quán/cô lập/bền vững |
| **Admin** | — | Vai trò quản trị hệ thống | Quản trị viên |
| **API** | Application Programming Interface | Giao diện lập trình cho tích hợp hệ thống | Giao tiếp lập trình ứng dụng |
| **Approved** | — | Trạng thái đăng ký đã được duyệt | Đã duyệt |
| **Audit log** | — | Nhật ký ghi lại thao tác/biến đổi dữ liệu | Nhật ký kiểm toán |
| **Autoscale** | — | Tự động tăng/giảm tài nguyên theo tải | Tự động mở rộng |
| **BI** | Business Intelligence | Phân tích dữ liệu, báo cáo hỗ trợ quyết định | Trí tuệ doanh nghiệp |
| **BR** | Business Rule | Quy tắc nghiệp vụ điều khiển hành vi hệ thống | Quy tắc nghiệp vụ |
| **Cache** | — | Lưu tạm dữ liệu để truy cập nhanh | Bộ nhớ đệm |
| **Capacity** | — | Sức chứa tối đa của lớp/phòng | Chỉ tiêu/sức chứa |
| **CDN** | Content Delivery Network | Mạng phân phối nội dung tĩnh để tăng tốc | Mạng phân phối nội dung |
| **Completed** | — | Trạng thái đăng ký đã hoàn tất môn | Đã hoàn thành |
| **CRUD** | Create, Read, Update, Delete | Tập thao tác cơ bản với dữ liệu | Tạo/Đọc/Cập nhật/Xóa |
| **CSV** | Comma-Separated Values | Định dạng tệp dữ liệu dạng bảng | Tệp giá trị phân tách bằng dấu phẩy |
| **CTĐT** | — | Chương trình đào tạo, có phiên bản theo niên khóa | Curriculum |
| **CVHT** | — | Cố vấn học tập | Academic advisor |
| **DEV** | Development | Môi trường phát triển | Môi trường phát triển |
| **Dis/Overbooking** | — | Vượt quá số chỗ cho phép khi đăng ký | Đăng ký quá chỗ |
| **DR (site)** | Disaster Recovery | Hạ tầng dự phòng khắc phục thảm họa | Trung tâm dự phòng |
| **Drop/ Dropped** | — | Hủy đăng ký môn | Hủy đăng ký |
| **Dashboard** | — | Trang tổng quan số liệu | Bảng điều khiển |
| **E-mail RFC 5322** | — | Chuẩn định dạng địa chỉ email | Tiêu chuẩn RFC 5322 |
| **GET/POST/PUT/DELETE** | — | Các phương thức HTTP dùng trong API | Phương thức HTTP |
| **GPA** | Grade Point Average | Điểm trung bình học tập | Điểm trung bình tích lũy |
| **GV** | — | Giảng viên | Teacher/Instructor |
| **HA** | High Availability | Khả dụng cao của hệ thống | Tính sẵn sàng cao |
| **HTTPOnly cookie** | — | Cookie chỉ truy cập được từ HTTP, chống XSS | Cookie an toàn (HTTPOnly) |
| **Iterative & Incremental** | — | Quy trình phát triển lặp – tăng trưởng | Phát triển lặp & tăng dần |
| **LMS** | Learning Management System | Hệ thống quản lý học tập | Hệ quản trị học tập |
| **Lockout** | — | Khóa tạm tài khoản sau nhiều lần sai | Khóa tạm |
| **LMS/SSO/SIS** | — | Nhóm hệ tích hợp: học tập / đăng nhập tập trung / thông tin sinh viên | Tích hợp hệ thống |
| **NFR** | Non-Functional Requirements | Yêu cầu phi chức năng (hiệu năng, bảo mật…) | Yêu cầu phi chức năng |
| **On-prem** | On-premises | Triển khai tại hạ tầng nội bộ | Tại chỗ |
| **Pending** | — | Trạng thái đăng ký chờ duyệt | Đang chờ |
| **PDF** | Portable Document Format | Định dạng tài liệu phổ biến | Tệp PDF |
| **PĐT** | — | Phòng Đào tạo | Academic Affairs |
| **PROD** | Production | Môi trường sản xuất | Môi trường vận hành |
| **RPO** | Recovery Point Objective | Mức mất dữ liệu tối đa cho phép khi sự cố | Mục tiêu điểm khôi phục |
| **RTO** | Recovery Time Objective | Thời gian khôi phục dịch vụ tối đa cho phép | Mục tiêu thời gian khôi phục |
| **Scale-out** | — | Mở rộng theo chiều ngang (thêm máy) | Mở rộng ngang |
| **SIS** | Student Information System | Hệ thống thông tin sinh viên | Hệ quản trị thông tin SV |
| **SSO** | Single Sign-On | Đăng nhập một lần cho nhiều hệ thống | Đăng nhập một lần |
| **Staging/UAT** | User Acceptance Testing | Môi trường kiểm thử chấp nhận người dùng | Môi trường UAT/Staging |
| **SV** | — | Sinh viên | Student |
| **Token/Access token** | — | Chuỗi định danh phiên/ủy quyền | Mã truy cập |
| **TKB** | — | Thời khóa biểu | Timetable |
| **UI** | User Interface | Giao diện người dùng | Giao diện |
| **Uptime 99.9%** | — | Cam kết sẵn sàng dịch vụ trong mùa đăng ký | Thời gian hoạt động |
| **UUID/ULID** | — | Định danh duy nhất cho bản ghi | Mã định danh duy nhất |
| **Use case (UC-xx)** | — | Trường hợp sử dụng; danh mục UC-01…UC-31 | Trường hợp sử dụng |
| **Waitlist** | — | Danh sách chờ khi lớp đầy | Danh sách chờ |

**Chương 2. Phân tích yêu cầu**

**2.1. Mô tả dự án (tóm tắt)**

**Mục tiêu:** Hệ thống quản lý đăng ký tín chỉ hỗ trợ trọn quy trình học vụ theo học chế tín chỉ: mở lớp → tư vấn lộ trình → đăng ký/huỷ/đổi → kiểm tra tiên quyết & xung đột TKB → tính/chốt học phí → theo dõi kết quả.  
Tập trung 3 đích đến:

1. Sinh viên đăng ký nhanh, chính xác, đúng lộ trình;
2. Giảng viên/Phòng Đào tạo vận hành, kiểm duyệt và tối ưu sĩ số;
3. Nhà trường giám sát số liệu thời gian thực để ra quyết định.

**Chức năng theo vai trò**

Sinh viên

* Tìm & đăng ký học phần: Lọc theo kỳ/khoa/GV/khung giờ/cơ sở; giỏ đăng ký–hủy–đổi; trạng thái (đã đăng ký/chờ duyệt/waitlist/đã hủy).
* Kiểm tra ràng buộc & xung đột: Tự động kiểm tra tiên quyết/học trước/song hành, hạn mức tín chỉ; cảnh báo trùng lịch học/thi; gợi ý lớp thay thế.
* Lộ trình – học phí – thông báo: Gợi ý môn còn thiếu theo CTĐT; ước tính & thanh toán học phí (nếu áp dụng), hóa đơn điện tử; nhận thông báo đợt đăng ký/thay đổi lịch/phòng/thi; xem điểm & phản hồi môn học.

Giảng viên

* Quản lý lớp phụ trách: Xem sĩ số theo thời gian thực; lọc theo nhóm thực hành/buổi/tuần; đề xuất tăng/giảm sĩ số, gộp lớp, đổi phòng/ca.
* Tổ chức lớp học: (Nếu tích hợp) điểm danh/nhập minh chứng; đăng thông báo tới lớp; chia sẻ tài liệu cơ bản.
* Theo dõi & báo cáo nhanh: Tình trạng đăng ký, mức độ quan tâm học phần, tỉ lệ trùng lịch sinh viên.

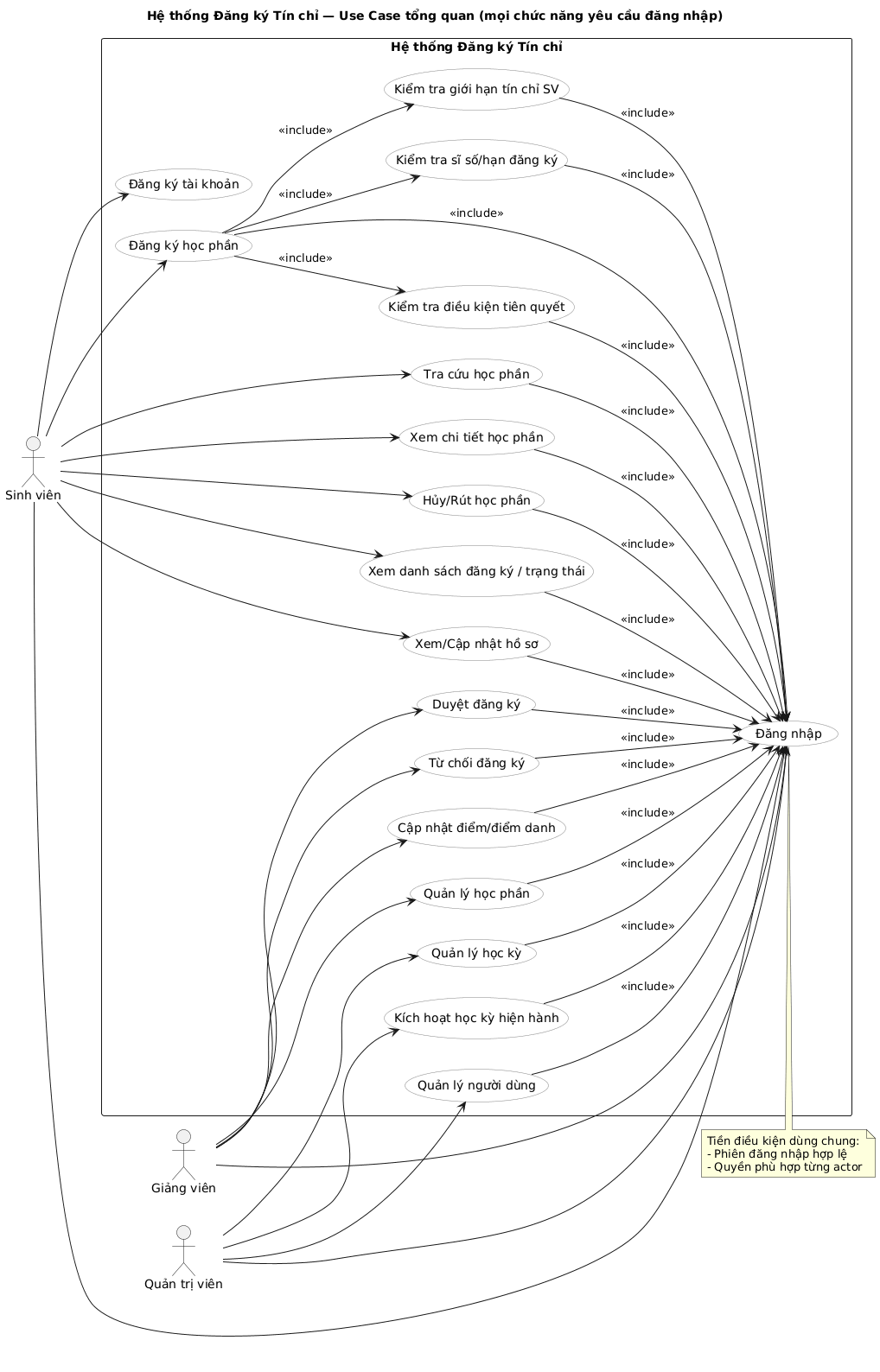
Quản trị viên

* Cấu hình & lập kế hoạch: Tạo/cập nhật kỳ học, mốc mở–đóng đăng ký; quản lý CTĐT, học phần, ràng buộc (tiên quyết/song hành/cấm trùng), chỉ tiêu–lịch–phòng; chính sách waitlist.
* Điều phối & kiểm duyệt: Phân công giảng viên; xét duyệt ngoại lệ (vượt trần tín chỉ, học chéo khoa… nếu áp dụng); phát hiện & xử lý xung đột phòng/GV; gộp/huỷ/mở thêm lớp; phát hành thông báo.

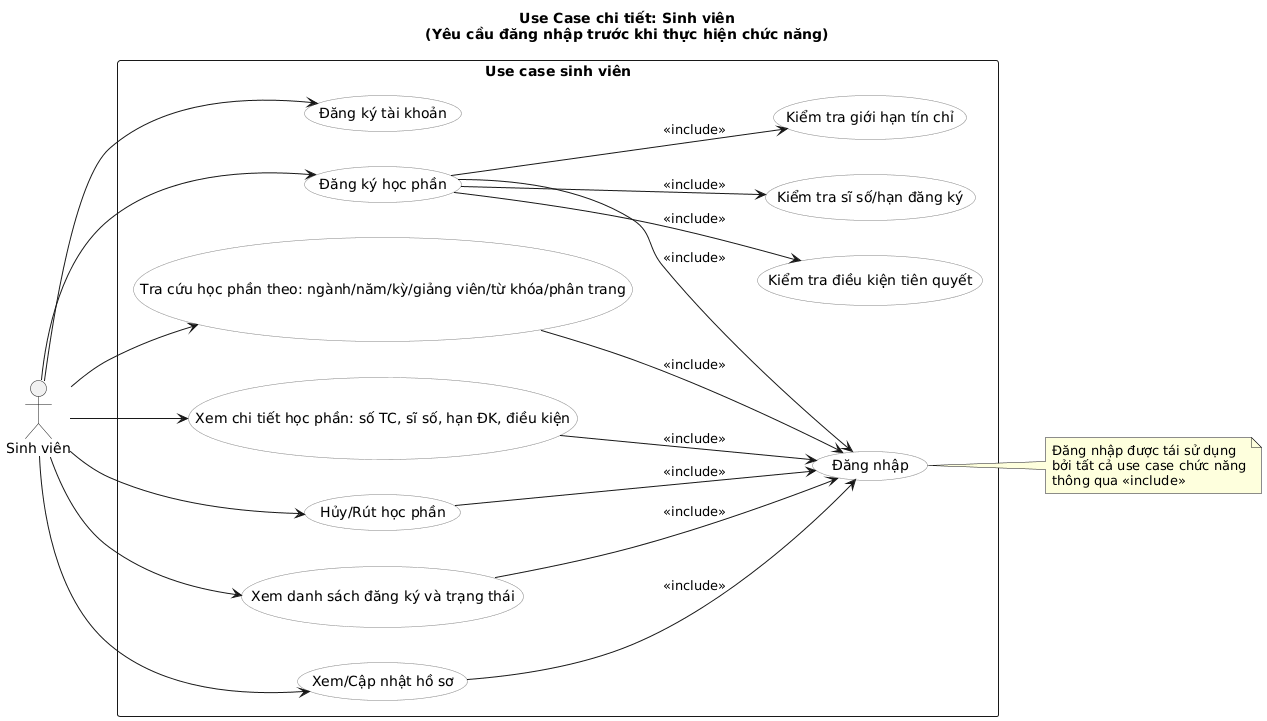
**Quy trình nghiệp vụ chính**

1. **Chuẩn bị học kỳ:** PĐT cấu hình kỳ, mở lớp, đặt mốc thời gian & quy tắc.
2. **Tư vấn & gợi ý:** Hệ thống đề xuất lộ trình; CVHT ghi chú khi cần.
3. **Đăng ký chính thức:** SV chọn lớp; hệ thống kiểm tra tiên quyết, trùng lịch, hạn mức; vào lớp hoặc waitlist.
4. **Chốt đăng ký & học phí:** Tính phí theo tín chỉ; SV thanh toán; kế toán đối soát & xuất hoá đơn.
5. **Điều phối lớp:** Theo dõi lấp đầy; mở thêm/gộp/huỷ; thông báo thay đổi.
6. **Vận hành trong kỳ:** Cập nhật danh sách, điểm danh, thông báo; xử lý ngoại lệ.
7. **Kết thúc học kỳ:** Khoá sửa đổi, tổng kết sĩ số, đồng bộ điểm (nếu có), lưu trữ & báo cáo.

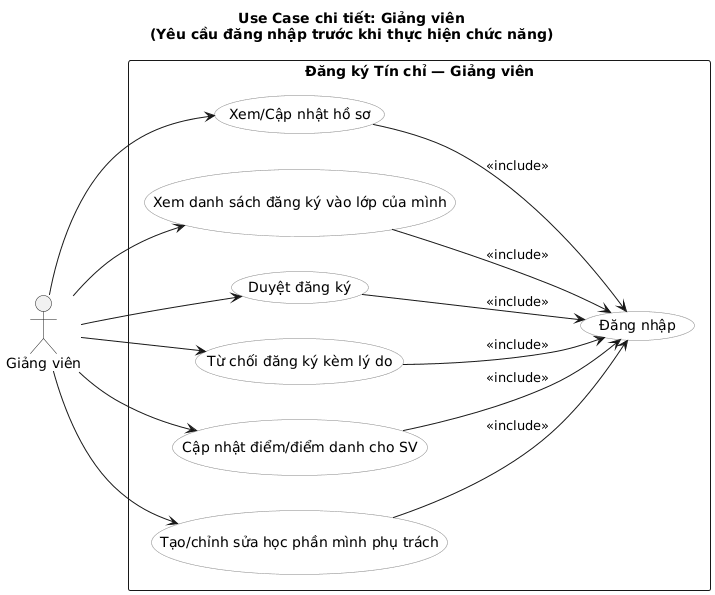
**2.1.1. Các use case Diagram**

****

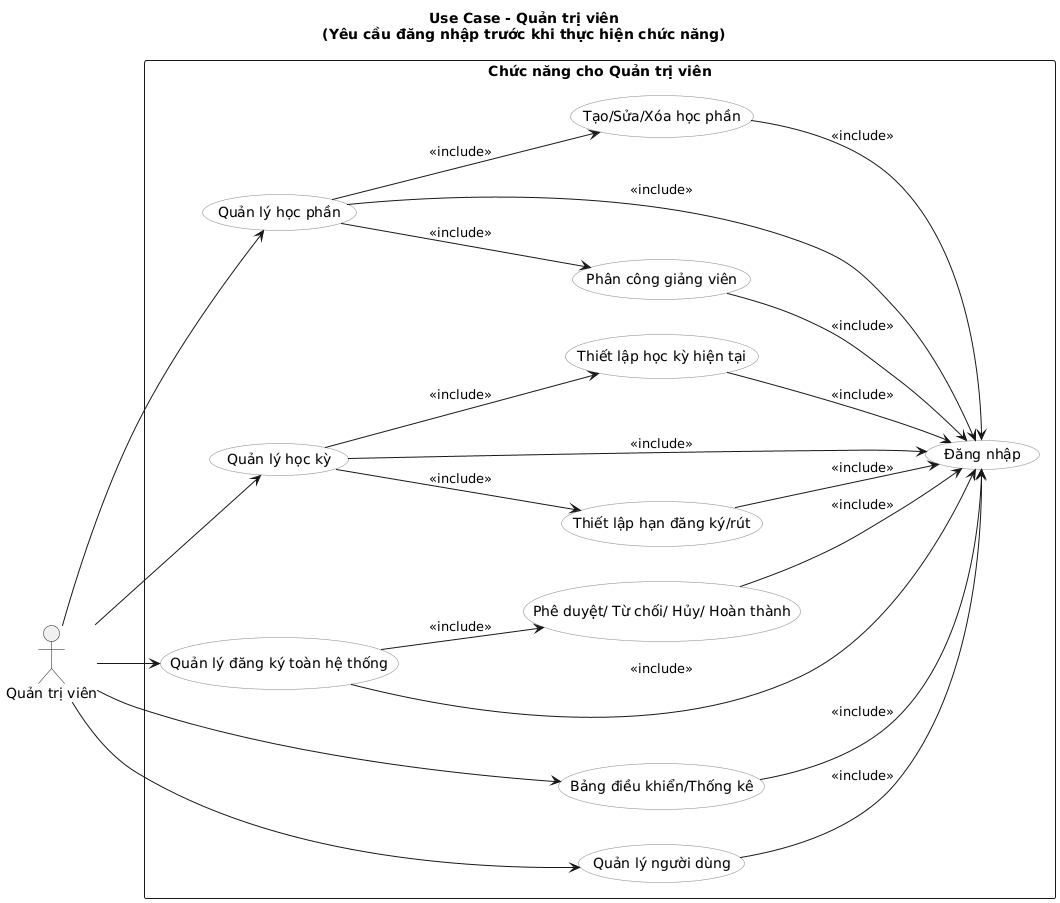
*Hình 1. Use case tổng quát hệ thống*

**

*Hình 2. Use case sinh viên*

**

*Hình 3. Use case giảng viên*

**

*Hình 4. Use case quản trị viên*

*2.1.2. Mô tả actors*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Actor** | **Nhiệm vụ chính** | **Quyền / Thao tác** | **Luồng thao tác** | **Ràng buộc / Quy tắc** | **Màn hình chính** |
| Sinh viên | Tìm & xem môn; đăng ký/hủy; theo dõi trạng thái; xem điểm. | Lọc/tìm môn; xem sĩ số, GV, lịch; Đăng ký; Hủy trước hạn; Xem điểm & biên nhận. | Đăng nhập → Tìm môn → Chi tiết → Đăng ký → Theo dõi 'Đăng ký của tôi' → Hủy (nếu cần) → Xem điểm khi hoàn tất. | Chỉ khi kỳ/môn mở; không vượt sĩ số; trong hạn đăng ký/rút; điểm xuất hiện khi completed. | Danh sách môn; Chi tiết môn; Đăng ký của tôi; Hồ sơ cá nhân. |
| Giảng viên | Quản lý lớp phụ trách; duyệt/từ chối; cập nhật tiến độ & điểm. | Cập nhật thông tin lớp (lịch, hạn, sĩ số); Duyệt/Từ chối yêu cầu; Nhập điểm; đánh dấu completed. | Vào quản lý → Xem yêu cầu → Phê duyệt/Từ chối → Cập nhật lớp → Nhập điểm & hoàn tất. | Chỉ lớp được phân công; tuân thủ hạn; trạng thái pending/approved/rejected/dropped/completed. | QL khóa học; QL đăng ký; Dashboard giảng viên. |
| Quản trị viên | Quản trị người dùng, học kỳ, học phần, đăng ký; theo dõi số liệu. | Quản lý người dùng & vai trò; tạo/sửa học kỳ; CRUD học phần; gán GV/kỳ; đặt sĩ số, hạn đăng ký/rút; duyệt/điều chỉnh đăng ký; xem dashboard. | Dashboard → Thiết lập học kỳ → Tạo/cấu hình học phần → Theo dõi/duyệt đăng ký → Quản lý tài khoản. | Toàn quyền dữ liệu; hạn & sĩ số chi phối SV/GV; đảm bảo nhất quán trạng thái & kỳ mở/đóng. | Dashboard quản trị; QL người dùng; QL học kỳ; QL học phần; QL đăng ký. |

**2.2 Yêu cầu chức năng**

**2.2.1. Danh mục Use Case**

**Nhóm Xác thực & Hồ sơ**

UC-01: Đăng nhập

UC-02: Đăng ký tài khoản

UC-03: Đăng xuất

UC-04: Xem & cập nhật hồ sơ cá nhân (bao gồm đổi mật khẩu)

**Nhóm Khóa học (Sinh viên/Khách xem; GV/QL trị quản lý)**

UC-05: Tra cứu danh sách học phần (lọc/tìm kiếm)  
UC-06: Xem chi tiết học phần  
UC-07: Đăng ký học phần  
UC-08: Hủy/Drop đăng ký học phần của tôi  
UC-09: Xem các đăng ký của tôi (lọc theo trạng thái)  
UC-10: Tải/xuất danh mục đăng ký của tôi (tệp/báo cáo)

**Nhóm Quản trị học phần/học kỳ (Admin/Giảng viên)**

UC-11: Tạo học phần  
 UC-12: Cập nhật học phần  
 UC-13: Xóa học phần  
 UC-14: Quản lý giảng viên phụ trách học phần  
 UC-15: Quản lý/hạch toán sức chứa, lịch học, phòng học

**Nhóm Quản trị đăng ký (Admin/Giảng viên)**

UC-16: Duyệt/Phê duyệt đăng ký  
 UC-17: Từ chối đăng ký  
 UC-18: Đánh dấu hoàn thành/hủy/báo cáo tình trạng đăng ký  
 UC-19: Tìm kiếm/lọc đăng ký theo sinh viên, học phần, trạng thái

**Nhóm Quản trị học kỳ (Admin)**

UC-20: Tạo học kỳ  
 UC-21: Cập nhật học kỳ  
 UC-22: Đóng/Mở học kỳ, thiết lập cửa sổ đăng ký  
 UC-23: Xem học kỳ hiện hành

**Nhóm Quản trị người dùng (Admin)**

UC-24: Tra cứu người dùng  
 UC-25: Xem chi tiết người dùng  
 UC-26: Lọc người dùng theo vai trò  
 UC-27: (Tuỳ chọn) Gán/đổi vai trò người dùng

**Nhóm Dashboard & Báo cáo**

UC-28: Dashboard Sinh viên (tổng tín chỉ, tình trạng đăng ký gần đây)  
 UC-29: Dashboard Giảng viên (lớp phụ trách, đăng ký môn mình dạy)  
 UC-30: Dashboard Quản trị (thống kê người dùng/khóa học/đăng ký)  
 UC-31: Xem báo cáo/tải số liệu tổng hợp (nếu có nút tải)

**2.2.2 Đặc tả chi tiết cho từng Use Case**

**UC-01: Đăng nhập**

* **Mục tiêu:** Cho phép người dùng hợp lệ truy cập hệ thống.
* **Phạm vi:** Xác thực phiên dựa trên email/mật khẩu.
* **Tác nhân:** Khách.
* **Tiền điều kiện:** Tài khoản tồn tại, trạng thái *active*.
* **Kích hoạt:** Người dùng nhấn **Login** trên trang Login.
* **Hậu điều kiện (Success):** Phiên hợp lệ được tạo; chuyển hướng **Dashboard** theo vai trò.  
  **(Failure):** Không tạo phiên; hiển thị thông báo lỗi cụ thể.
* **Luồng chính:**
  1. Người dùng nhập **email** và **mật khẩu**.
  2. Hệ thống kiểm tra định dạng email, độ dài mật khẩu (≥8 ký tự—cấu hình).
  3. Gửi yêu cầu xác thực (Auth service).
  4. Nếu hợp lệ: tạo accessToken/session; lưu an toàn (HTTPOnly cookie).
  5. Điều hướng tới Dashboard tương ứng (SV/GV/Admin).
* **Luồng thay thế/ngoại lệ:**
  1. A1: Sai thông tin → tăng bộ đếm thất bại; **BR:** Email hợp lệ; lockout sau N lần; token hết hạn sau T phút; buộc 2FA (nếu bật chính sách).
* **Dữ liệu & Ràng buộc:** email (RFC 5322), password (≥8, có chữ & số—tùy chính sách).
* **Quyền truy cập:** Public endpoint.
* **Thông báo/Lỗi:** Sai email hoặc mật khẩu; .
* **AC:**
  1. AC1: Nhập đúng thông tin → vào Dashboard đúng vai trò.
  2. AC2: Token lưu dưới cookie HTTPOnly.
* **Audit:** Ghi thời điểm đăng nhập, IP, user-agent, kết quả (success/fail).
* **NFR:** Thời gian phản hồi < 2s trong 95th percentile.
* **UI/API:** Login.js; (Auth service nội bộ).

**UC-02: Đăng ký tài khoản**

* **Mục tiêu:** Cho phép tạo tài khoản mới.
* **Phạm vi:** Tự đăng ký người dùng (mặc định role student; teacher qua admin duyệt nếu chính sách yêu cầu).
* **Tác nhân:** Khách.
* **Tiền điều kiện:** Email chưa tồn tại.
* **Kích hoạt:** Nhấn **Create account** trên Register.
* **Hậu điều kiện (Success):** Tài khoản ở trạng thái *pending* hoặc *active* (theo cấu hình xác minh email).  
  **(Failure):** Không tạo tài khoản, hiển thị lỗi cụ thể.
* **Luồng chính:**
  1. Nhập họ tên, email, mật khẩu, (mã SV, khoa, niên khóa...).
  2. Kiểm tra client-side định dạng & bắt buộc trường.
  3. Gửi đăng ký; server kiểm tra *uniqueness* email, studentCode.
  4. Tạo tài khoản; gửi email xác minh (nếu bật).
  5. Hiển thị hướng dẫn đăng nhập/xác minh.
* **Ngoại lệ:** Email trùng; studentCode trùng; mật khẩu yếu; lỗi mailer.
* **BR:** Password policy; studentCode duy nhất; có thể yêu cầu email tổ chức (@...edu.vn).
* **Dữ liệu/Ràng buộc:** fullName (≤100), email (unique), password (policy), studentCode (regex mã SV), faculty, intakeYear.
* **Quyền:** Public endpoint.
* **Thông báo:** Email đã tồn tại; Vui lòng xác minh email.
* **AC:**
  1. AC1: Đăng ký với dữ liệu hợp lệ → tạo user.
  2. AC2: Email trùng → bị từ chối với mã lỗi 409.
  3. AC3: Nhận email xác minh khi bật policy.
* **Audit:** Log tạo tài khoản (by=system, ip).
* **NFR:** Chống spam (rate limit, CAPTCHA nếu cần).
* **UI/API:** Register.js.

**UC-03: Đăng xuất**

* **Mục tiêu:** Kết thúc phiên làm việc an toàn.
* **Tác nhân:** Mọi vai trò.
* **Tiền điều kiện:** Đang đăng nhập.
* **Kích hoạt:** Chọn **Logout** trên navbar.
* **Hậu điều kiện:** Xóa session/token; điều hướng Login.
* **Luồng chính:** Xóa cookie token (server/invalidate); redirect.
* **Ngoại lệ:** Lỗi mạng → vẫn xóa token client; hiển thị đã đăng xuất.
* **BR:** Invalidate phía server để vô hiệu hóa token bị rò rỉ.
* **AC:** Sau logout, truy cập trang bảo vệ → bị yêu cầu đăng nhập.
* **Audit:** Log logout (by user, time).

**UC-04: Xem & cập nhật hồ sơ cá nhân**

* **Mục tiêu:** Cho phép người dùng tự quản hồ sơ, đổi mật khẩu.
* **Tác nhân:** SV/GV/Admin.
* **Tiền điều kiện:** Đăng nhập.
* **Kích hoạt:** Mở trang Profile.
* **Hậu điều kiện:** Hồ sơ được cập nhật; mật khẩu thay đổi (nếu thao tác).
* **Luồng chính:**
  1. Tải dữ liệu hồ sơ hiện tại.
  2. Người dùng sửa các trường cho phép; bấm **Lưu**.
  3. Server xác thực giá trị; cập nhật; trả về bản mới.
* **Luồng thay thế:** Đổi mật khẩu → yêu cầu oldPassword, newPassword, xác nhận khớp.
* **Ngoại lệ:** oldPassword sai; định dạng điện thoại sai; kích thước ảnh > 2MB.
* **BR:** Một số trường chỉ đọc (email, mã SV); đổi mật khẩu yêu cầu lại đăng nhập theo chính sách.
* **Dữ liệu/Ràng buộc:** phone (regex VN), avatar (≤2MB, jpg/png), displayName (≤100).
* **AC:** Cập nhật hợp lệ phản ánh ngay UI; sai định dạng hiển thị message cạnh trường.
* **Audit:** Log thay đổi trường nhạy cảm; che giá trị.

**UC-05: Tra cứu danh sách học phần (lọc/tìm kiếm)**

* **Mục tiêu:** Cung cấp danh mục học phần có lọc/tìm kiếm.
* **Tác nhân:** Khách/SV/GV/Admin.
* **Tiền điều kiện:** Có dữ liệu học phần.
* **Kích hoạt:** Mở Courses hoặc nhập từ khóa.
* **Hậu điều kiện:** Hiển thị danh sách theo tiêu chí, phân trang.
* **Luồng chính:**
  1. Client gọi GET /api/courses kèm tham số (q, major, yearLevel, semesterNumber, page, size).
  2. Server truy vấn, trả về total, items[].
  3. UI render danh sách; bộ lọc giữ trạng thái.
* **Ngoại lệ:** Không có kết quả → banner "Không tìm thấy học phần"; lỗi mạng → retry.
* **BR:** Từ khóa ≥2 ký tự; page size trong {10,20,50}; sắp xếp mặc định theo courseCode.
* **Dữ liệu/Ràng buộc:** Tham số hợp lệ (enum/number range), chống SQLi.
* **AC:** Gõ từ khóa trả về đúng tập; bộ lọc kết hợp hoạt động.
* **Audit:** Không yêu cầu.

**UC-06: Xem chi tiết học phần**

* **Mục tiêu:** Hiển thị đầy đủ thông tin 1 học phần và trạng thái của SV.
* **Tác nhân:** Tất cả.
* **Tiền điều kiện:** courseId hợp lệ.
* **Kích hoạt:** Nhấp vào 1 dòng trong Courses.
* **Hậu điều kiện:** Thông tin chi tiết + nút hành động tương ứng trạng thái/role.
* **Luồng chính:**
  1. GET /api/courses/:id tải chi tiết (mô tả, credits, lịch, phòng, capacity, enrolled, tiên quyết, GV…).
  2. GET /api/semesters/current để xác định cửa sổ đăng ký.
  3. Nếu là SV đã đăng nhập, GET /api/registrations?studentId=me&courseId=:id để lấy trạng thái.
* **Ngoại lệ:** 404 course; server lỗi.
* **BR:** Ngoài cửa sổ → disable **Đăng ký**; đã đăng ký → hiển thị **Drop** nếu cho phép.
* **AC:** Hiển thị chuẩn xác capacity/enrolled; trạng thái nút đúng bối cảnh.

**UC-07: Đăng ký học phần**

* **Mục tiêu:** Ghi nhận yêu cầu đăng ký môn của SV.
* **Tác nhân:** Sinh viên.
* **Tiền điều kiện:** Trong cửa sổ; không trùng lịch; chưa vượt tín chỉ; còn chỗ; đạt tiên quyết.
* **Kích hoạt:** Nút **Đăng ký** tại CourseDetail.
* **Hậu điều kiện:** Tạo bản ghi Registration (pending/approved).
* **Luồng chính:**
  1. Client gửi POST /api/registrations {courseId, semesterId(current)}.
  2. Server **re-validate** các ràng buộc (cửa sổ, capacity, trùng lịch, tín chỉ, tiên quyết).
  3. Nếu đạt: tạo record; nếu auto-approve và còn chỗ → approved & tăng enrolled.
* **Ngoại lệ:** Vi phạm bất kỳ ràng buộc → trả mã lỗi chuyên biệt (OUT\_OF\_WINDOW, CAPACITY\_FULL, CLASH, CREDIT\_LIMIT, PREREQ\_FAIL, DUPLICATE).
* **BR (chính):** BR-07.1…BR-07.6 như đã nêu ở mục 4.
* **Dữ liệu/Ràng buộc:** courseId bắt buộc; semesterId = current; id hợp lệ UUID/ULID.
* **AC:** Gặp CAPACITY\_FULL → hiển thị "Lớp đã đủ chỗ"; thành công → thẻ trạng thái chuyển approved/pending tức thì.
* **Audit:** Ghi by=student, courseId, kết quả.

**UC-08: Hủy/Drop đăng ký học phần**

* **Mục tiêu:** Cho phép SV tự hủy đăng ký còn hiệu lực.
* **Tác nhân:** Sinh viên.
* **Tiền điều kiện:** Có đăng ký ở trạng thái cho phép drop; trong hạn drop.
* **Kích hoạt:** Nút **Drop** tại MyRegistrations/CourseDetail.
* **Hậu điều kiện:** status → dropped; giảm enrolled nếu previously approved.
* **Luồng chính:** PUT /api/registrations/:id/drop → server kiểm tra quyền sở hữu + hạn drop → cập nhật.
* **Ngoại lệ:** Quá hạn; status không cho drop; không phải chủ sở hữu.
* **BR:** Giải phóng suất khi drop thành công; ghi lý do (tùy chọn).
* **AC:** Sau drop, bản ghi biến mất khỏi danh sách *đang học*; xuất hiện ở tab *đã hủy*.

**UC-09: Xem các đăng ký của tôi**

* **Mục tiêu:** Cung cấp danh sách đăng ký và tổng tín chỉ của SV.
* **Tác nhân:** Sinh viên.
* **Tiền điều kiện:** Đăng nhập.
* **Kích hoạt:** Mở MyRegistrations.
* **Hậu điều kiện:** Hiển thị danh sách + tổng tín chỉ theo trạng thái được tính.
* **Luồng chính:** GET /api/registrations?studentId=me → render; bộ lọc client theo status.
* **Ngoại lệ:** Không dữ liệu; lỗi mạng.
* **BR:** Tổng tín chỉ chỉ tính approved (hoặc cấu hình khác).
* **AC:** Khi đổi bộ lọc, danh sách cập nhật trong <300ms (client).

**UC-10: Tải/xuất danh mục đăng ký của tôi**

* **Mục tiêu:** Xuất báo cáo cá nhân (CSV/PDF).
* **Tác nhân:** Sinh viên.
* **Tiền điều kiện:** Có dữ liệu đăng ký.
* **Kích hoạt:** Nút **Download**.
* **Hậu điều kiện:** File được tạo và tải về.
* **Luồng chính:** Client yêu cầu /api/registrations/export?studentId=me&format=csv|pdf (nếu có) hoặc export client-side.
* **Ngoại lệ:** Lỗi tạo file; kích thước vượt giới hạn.
* **BR:** Định dạng cột chuẩn; timezone học kỳ; dấu phẩy thập phân theo locale.
* **AC:** Mở file thấy đủ dòng, số tín chỉ khớp tổng trên UI.

**UC-11: Tạo học phần**

* **Mục tiêu:** Thêm lớp/học phần mới vào kỳ phù hợp.
* **Tác nhân:** Admin/GV (được phân quyền).
* **Tiền điều kiện:** Có học kỳ *active*; dữ liệu tối thiểu sẵn có.
* **Kích hoạt:** Nút **Thêm học phần** tại CourseManagement.
* **Hậu điều kiện:** Học phần mới sẵn sàng để SV đăng ký (khi cửa sổ mở).
* **Luồng chính:**
  1. Nhập form: courseCode, courseName, credits, department, yearLevel, schedule[], room, capacity, teachers[], semesterId.
  2. Server kiểm tra: mã duy nhất, capacity≥0, lịch hợp lệ.
  3. Lưu DB; trả về đối tượng mới.
* **Ngoại lệ:** Trùng mã; capacity<0; lịch sai định dạng.
* **BR:** Không được tạo lớp ngoài kỳ cho phép (hoặc phải gắn rõ semesterId).
* **AC:** Tạo xong hiển thị trong danh sách quản trị và trang *Courses* (nếu publish=true).

**UC-12: Cập nhật học phần**

* **Mục tiêu:** Sửa thông tin học phần hiện hữu.
* **Tác nhân:** Admin/GV (được phân quyền).
* **Tiền điều kiện:** Học phần tồn tại.
* **Kích hoạt:** Hành động **Sửa**.
* **Hậu điều kiện:** Dữ liệu mới có hiệu lực.
* **Luồng chính:** PUT /api/courses/:id với các trường thay đổi.
* **Ngoại lệ:** Giảm capacity < approved; lịch gây xung đột phòng.
* **BR:** Không giảm capacity thấp hơn số đã *approved*; thay đổi lịch phải thông báo SV.
* **AC:** Sau cập nhật, SV/ GV thấy lịch mới; đăng ký không bị mất dữ liệu.

**UC-13: Xóa học phần**

* **Mục tiêu:** Gỡ bỏ học phần (xóa mềm khuyến nghị).
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Không có ràng buộc hoạt động (hoặc cho phép hủy lớp theo chính sách).
* **Kích hoạt:** **Xóa** tại CourseManagement.
* **Hậu điều kiện:** Học phần ẩn khỏi tìm kiếm; SV được thông báo.
* **Luồng chính:** Đặt status=archived (soft delete) qua PUT hoặc DELETE nếu chưa có đăng ký.
* **Ngoại lệ:** Đã có SV *approved* → buộc hủy lớp, hoàn tiền theo chính sách (nếu có).
* **BR:** Gửi thông báo đến SV đã đăng ký; lưu lý do hủy.
* **AC:** Học phần không còn hiển thị trong Courses.

**UC-14: Quản lý giảng viên phụ trách học phần**

* **Mục tiêu:** Gán/đổi giảng viên cho học phần.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Giảng viên tồn tại, role=teacher.
* **Kích hoạt:** Chọn GV trong CourseManagement.
* **Hậu điều kiện:** Quan hệ course–teacher cập nhật.
* **Luồng chính:** GET /api/users/teachers → chọn → PUT /api/courses/:id cập nhật teachers[].
* **Ngoại lệ:** GV không hợp lệ; gv trùng lặp.
* **BR:** Ít nhất 1 GV phụ trách.
* **AC:** CourseDetail hiển thị GV mới.

**UC-15: Quản lý lịch học, phòng học, sức chứa**

* **Mục tiêu:** Thiết lập lịch/room/capacity phù hợp.
* **Tác nhân:** Admin/GV (phân quyền).
* **Tiền điều kiện:** Phòng tồn tại; không bị trùng đặt.
* **Kích hoạt:** Sửa thuộc tính trên CourseManagement.
* **Hậu điều kiện:** Lịch/phòng/capacity cập nhật nhất quán.
* **Luồng chính:** PUT /api/courses/:id với schedule[], room, capacity.
* **Ngoại lệ:** Lịch chồng chéo; phòng không đủ chỗ.
* **BR:** Room capacity ≥ course capacity; chuẩn hóa format lịch (dayOfWeek, start, end, period).
* **AC:** Courses và CourseDetail phản ánh thay đổi.

**UC-16: Duyệt/Phê duyệt đăng ký**

* **Mục tiêu:** Xác nhận yêu cầu pending thành approved.
* **Tác nhân:** Admin/(GV nếu chính sách cho phép).
* **Tiền điều kiện:** Đăng ký ở trạng thái pending.
* **Kích hoạt:** Nút **Approve** tại RegistrationManagement.
* **Hậu điều kiện:** status=approved; cập nhật enrolled.
* **Luồng chính:**
  1. Lọc pending.
  2. Chọn một/multiple registrations.
  3. Server re-validate capacity/cửa sổ/trùng lịch/tín chỉ.
  4. Nếu đạt → cập nhật trạng thái; gửi thông báo cho SV.
* **Ngoại lệ:** Hết chỗ; phát hiện clash; quá cửa sổ → reject kèm mã lỗi tương ứng.
* **BR:** Duyệt hàng loạt phải đảm bảo **tính nguyên tử** từng bản ghi; ghi audit.
* **AC:** approved tăng; capacity giảm tương ứng.

**UC-17: Từ chối đăng ký**

* **Mục tiêu:** Đánh dấu thất bại kèm lý do.
* **Tác nhân:** Admin/(GV).
* **Tiền điều kiện:** Đăng ký pending.
* **Kích hoạt:** **Reject** tại RegistrationManagement.
* **Hậu điều kiện:** status=rejected; lưu reason.
* **Ngoại lệ:** —
* **BR:** Lý do bắt buộc (≥10 ký tự); thông báo SV.
* **AC:** Bản ghi hiển thị ở tab *rejected* với lý do.

**UC-18: Cập nhật trạng thái đăng ký (hoàn thành/hủy…)**

* **Mục tiêu:** Cập nhật completed, dropped, v.v. theo tiến trình học.
* **Tác nhân:** Admin/(GV).
* **Tiền điều kiện:** Đăng ký tồn tại.
* **Kích hoạt:** Hành động cập nhật trạng thái.
* **Hậu điều kiện:** Trạng thái phản ánh đúng thực tế; có log.
* **Luồng chính:** PUT/PATCH /api/registrations/:id với status mục tiêu; kiểm tra hợp lệ theo state machine.
* **Ngoại lệ:** Chuyển trái phép (ví dụ từ rejected → approved không qua quy trình lại).
* **BR:** Biểu đồ trạng thái cho phép: pending→approved|rejected; approved→completed|dropped.
* **AC:** Không thể cập nhật sang trạng thái không hợp lệ (HTTP 409).

**UC-19: Tìm kiếm/lọc đăng ký**

* **Mục tiêu:** Quản trị theo dõi tất cả đăng ký theo tiêu chí.
* **Tác nhân:** Admin/(GV).
* **Tiền điều kiện:** —
* **Kích hoạt:** Nhập từ khóa/bộ lọc.
* **Hậu điều kiện:** Danh sách lọc & phân trang.
* **Luồng chính:** GET /api/registrations?student=...&course=...&status=....
* **Ngoại lệ:** Không kết quả.
* **BR:** Chuẩn hóa bộ lọc; sắp xếp mặc định theo createdAt desc.
* **AC:** Tra cứu theo tên SV/ mã môn trả đúng tập.

**UC-20: Tạo học kỳ**

* **Mục tiêu:** Khởi tạo học kỳ mới với cửa sổ đăng ký.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Không chồng chéo với kỳ khác cùng năm học.
* **Kích hoạt:** **Thêm học kỳ** tại SemesterManagement.
* **Hậu điều kiện:** Kỳ mới có thể được đặt *current*.
* **Luồng chính:** POST /api/semesters {name, code, academicYear, semesterNumber, registrationStart, registrationEnd, isCurrent?}.
* **Ngoại lệ:** registrationStart≥registrationEnd; code trùng.
* **BR:** code duy nhất; ngày giờ theo timezone hệ thống.
* **AC:** Sau tạo, GET /api/semesters trả về bản ghi mới.

**UC-21: Cập nhật học kỳ**

* **Mục tiêu:** Điều chỉnh thông tin/cửa sổ đăng ký của kỳ.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Học kỳ tồn tại.
* **Kích hoạt:** **Sửa** tại SemesterManagement.
* **Hậu điều kiện:** Thông tin kỳ cập nhật.
* **Luồng chính:** PUT /api/semesters/:id.
* **Ngoại lệ:** Thu hẹp cửa sổ gây ảnh hưởng đăng ký đang diễn ra → yêu cầu xác nhận mạnh.
* **BR:** Khi thay đổi isCurrent, đảm bảo chỉ một kỳ current=true.
* **AC:** GET /api/semesters/current phản ánh đúng sau cập nhật.

**UC-22: Đóng/Mở học kỳ, thiết lập cửa sổ đăng ký**

* **Mục tiêu:** Bật/tắt khả năng đăng ký cho kỳ hiện hành.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Có kỳ được đánh dấu current.
* **Kích hoạt:** Toggle trên SemesterManagement.
* **Hậu điều kiện:** Cửa sổ áp dụng ngay cho UC-07/16.
* **Luồng chính:** Cập nhật thuộc tính thời gian/ trạng thái mở-đóng.
* **Ngoại lệ:** —
* **BR:** current là duy nhất; khi mở kỳ mới phải đóng kỳ cũ.
* **AC:** Nút **Đăng ký** bị vô hiệu hóa khi ngoài cửa sổ.

**UC-23: Xem học kỳ hiện hành**

* **Mục tiêu:** Cho client và server biết kỳ đang áp dụng quy tắc đăng ký.
* **Tác nhân:** All/Hệ thống.
* **Tiền điều kiện:** —
* **Kích hoạt:** Tải CourseDetail/Dashboard.
* **Hậu điều kiện:** Nhận đối tượng semester hiện hành.
* **Luồng chính:** GET /api/semesters/current.
* **AC:** Trả về 200 với bản ghi duy nhất hoặc 204 nếu chưa cấu hình.

**UC-24: Tra cứu người dùng**

* **Mục tiêu:** Quản trị danh sách người dùng hệ thống.
* **Tác nhân:** Admin.
* **Kích hoạt:** Mở UserManagement.
* **Hậu điều kiện:** Danh sách phân trang, có bộ lọc.
* **Luồng chính:** GET /api/users?page,size,q,role.
* **Ngoại lệ:** —
* **BR:** Ẩn thông tin nhạy cảm (hash password, token).
* **AC:** Lọc theo từ khóa tên/email hoạt động.

**UC-25: Xem chi tiết người dùng**

* **Mục tiêu:** Xem hồ sơ chi tiết từng user.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Người dùng tồn tại.
* **Kích hoạt:** Nhấn vào hàng/biểu tượng xem chi tiết.
* **Hậu điều kiện:** Modal chi tiết hiển thị.
* **Luồng chính:** GET /api/users/:id.
* **BR:** Che dữ liệu riêng tư; chỉ hiển thị trường cần thiết.
* **AC:** Thông tin khớp với thực tế; audit hiển thị (nếu có).

**UC-26: Lọc người dùng theo vai trò**

* **Mục tiêu:** Xem danh sách người dùng theo role.
* **Tác nhân:** Admin.
* **Kích hoạt:** Chọn student/teacher/admin filter.
* **Luồng chính:** GET /api/users?role=student|teacher|admin.
* **AC:** Danh sách chỉ gồm user có role tương ứng.

**UC-27: Gán/đổi vai trò người dùng (tuỳ chính sách)**

* **Mục tiêu:** Thay đổi vai trò tài khoản.
* **Tác nhân:** Admin.
* **Tiền điều kiện:** Người dùng tồn tại; không phải admin cuối cùng.
* **Kích hoạt:** Sửa role và **Lưu**.
* **Hậu điều kiện:** Vai trò cập nhật, session có thể bị vô hiệu để user đăng nhập lại.
* **Luồng chính:** PUT /api/users/:id {role}.
* **Ngoại lệ:** Cố hạ quyền admin cuối cùng → chặn.
* **BR:** Yêu cầu 2FA/xác nhận lại mật khẩu admin; lưu audit.
* **AC:** Quyền trên UI thay đổi theo role mới sau khi login lại.

**UC-28: Dashboard Sinh viên**

* **Mục tiêu:** Cung cấp bức tranh tổng quan cho SV.
* **Tác nhân:** Sinh viên.
* **Nội dung:** Tổng tín chỉ đã duyệt, đăng ký gần đây, cảnh báo trùng lịch/chưa đủ tiên quyết, kỳ hiện hành.
* **Luồng dữ liệu:** GET /api/registrations?studentId=me, GET /api/semesters/current.
* **BR:** Chỉ hiển thị dữ liệu của chính SV; **không** lộ thông tin người khác.
* **AC:** Số liệu trùng khớp với MyRegistrations.

**UC-29: Dashboard Giảng viên**

* **Mục tiêu:** Tổng quan lớp được phân công và tình trạng đăng ký.
* **Tác nhân:** Giảng viên.
* **Nội dung:** Danh sách lớp mình dạy, số SV đăng ký/approved, thông tin kỳ.
* **Luồng dữ liệu:** GET /api/courses/teacher/:id, GET /api/registrations?courseId=....
* **BR:** Chỉ thấy lớp mình phụ trách.
* **AC:** Số approved khớp với trang quản trị đăng ký.

**UC-30: Dashboard Quản trị**

* **Mục tiêu:** Thống kê hệ thống: người dùng, học phần, đăng ký.
* **Tác nhân:** Admin.
* **Nội dung:** Tổng số user theo vai trò, số học phần theo khoa/kỳ, số đăng ký theo trạng thái, top lớp đầy chỗ.
* **Luồng dữ liệu:** GET /api/users, GET /api/courses, GET /api/registrations.
* **BR:** Có thể cần cache/ETL đối với thống kê nặng.
* **AC:** Các số liệu khớp với từng module nguồn.

**2.3. Yêu cầu phi chức năng**

**1) Hiệu năng**

* **Tải dữ liệu song song**: Màn hình tổng quan gọi nhiều API đồng thời bằng Promise.all → giảm độ trễ khi hợp nhất số liệu.
* **Trạng thái tải rõ ràng**: Nhiều trang có cờ loading và hiển thị **spinner** trong lúc gọi API; các nút có trạng thái **disabled** khi đang xử lý.
* **Lọc/tìm kiếm phía client**: Danh sách (khóa học, đăng ký, người dùng) có tìm kiếm/lọc trực tiếp trên dữ liệu đã tải.
* **Chưa thấy**: cơ chế **phân trang (pagination)**, **debounce** tìm kiếm, **cache**/prefetch, hay bất kỳ ràng buộc mục tiêu **“< 2s”** trong mã giao diện.

**2) Bảo mật (phân quyền)**

* **Ngữ cảnh đăng nhập**: Có trang **đăng nhập**, dùng useAuth() để thực hiện login.
* **Phân quyền hiển thị theo vai trò**: UI điều kiện hóa theo role (**student / teacher / admin**).
  + Sinh viên mới thấy nút **đăng ký/hủy đăng ký** học phần.
  + Quản trị/giảng viên mới thấy các thao tác **phê duyệt/từ chối** đăng ký, quản trị học kỳ/khóa học/người dùng.
* **Đổi mật khẩu**: Trang hồ sơ cho phép **đổi mật khẩu**, kiểm tra tối thiểu độ dài và **xác nhận mật khẩu** khớp.
* **Chưa thấy**: chặn truy cập theo vai trò ở tầng route hay kiểm tra phân quyền phía server trong phần giao diện này (UI chỉ ẩn/hiện nút; không khẳng định kiểm soát truy cập phía backend).

**3) Khả năng sử dụng (UI/UX)**

* **Phản hồi thao tác**: Dùng **toast** báo lỗi/thành công khi tải dữ liệu, lưu thay đổi, phê duyệt/từ chối, đăng ký/hủy đăng ký.
* **Trạng thái rỗng & cảnh báo**: Có thông điệp khi **không có dữ liệu** hoặc khóa học không tồn tại/không hoạt động.
* **Biểu mẫu thân thiện**:
  + **Xác thực client**: yêu cầu bắt buộc, tối thiểu độ dài, mẫu năm học (YYYY-YYYY), thông báo lỗi cạnh trường.
  + **Hiển thị/ẩn mật khẩu** cho đăng nhập/đăng ký/đổi mật khẩu.
* **Khả năng thao tác**: có **tìm kiếm**, **lọc** (ví dụ lọc ngành, giảng viên, trạng thái), các **nút hành động** rõ ràng; bố cục **responsive** với lưới; sử dụng **icon** nhất quán.
* **Chưa thấy**: thuộc tính **ARIA**/phím tắt, hướng dẫn truy cập (accessibility) chuyên sâu.