**TÀI LIỆU ĐẶC TẢ YÊU CẦU PHẦN MỀM**

**WEBSITE QUẢN LÝ TỔ CHỨC ĐÀO TẠO CUSC**

1. **GIỚI THIỆU**
   1. **Mục đích**

* Hệ thống website quản lý tố chức đào tạo được phát triển nhằm:
  + Quản lý thông tin *khoá học, học phần, lớp học phần.*
  + Tự động phân lớp cho sinh viên dữ trên dữ liệu đầu vào.
  + Lập thời khoá biểu (lịch học, lịch thi) dựa trên các dữ liệu đầu vào bao gồm học phần, giảng viên, sinh viên, phòng học tại cơ sở.
  + Phát triển khả năng liên kết, tích hợp với hệ thống khác để nhận dữ liệu từ các nghiệp vụ có liên quan
  1. **Đối tượng sử dụng**

|  |  |
| --- | --- |
| **Vai trò** | **Mô tả** |
| Quản trị viên | Quản lý hệ thống, cấu hình khóa học, phân quyền. |
| Cán bộ đào tạo | Xem lịch dạy, điểm danh, quản lý lớp học phần. |
| Giảng viên | Xem lịch học, lịch thi, thông báo. |
| Học viên | Nhập liệu, quản lý lớp học, lập thời khóa biểu. |

* 1. **Phạm vi**
* Hệ thống tập trung vào quản lý đào tạo, không bao gồm các chức năng thanh toán và đào tạo trực tuyến.
* Dữ liệu học viên và giảng viên được đồng bộ từ hệ thống khác qua các file Excel, JSON hoặc API.

1. **MÔ TẢ CHI TIẾT CHỨC NĂNG ĐẶC TRƯNG CỦA HỆ THỐNG**
   1. **Quản lý khoá học và học phần**

* Mã chức năng: FUNC-001
* Tên chức năng: Quản lý khoá học và học phần
* Mục tiêu:
  + Cho phép quản trị viên/cán bộ đào tạo, chỉnh sửa, xoá thông tin khoá học và học phần
  + Gán giảng viên phụ trách cho từng học phần
* Đầu vào (Input):
  + Thông tin khoá học:
    - Tên khoá học (*bắt buộc)*
    - Mã khoá học (*bắt buộc, duy nhất)*
    - Thời gian bắt đầu (*định dạng DD/MM/YYYY)*
    - Thời gian kết thúc (*định dạng DD/MM/YYYY)*
  + Thông tin học phần:
    - Mã học phần *(bắt buộc, duy nhất).*
    - Tên học phần *(bắt buộc).*
    - Số tín chỉ *(bắt buộc, kiểu số).*
    - Khóa học liên kết (*chọn từ danh sách khóa học hiện có*).
  + Phân công giảng viên:
    - Chọn giảng viên từ danh sách giảng viên có sẵn *(đồng bộ từ hệ thống khác).*
* Luồng xử lý chính:
  + Tạo khóa học:
    - Người dùng chọn "Thêm khóa học".
    - Nhập thông tin khóa học (tên, mã, thời gian).
    - Hệ thống kiểm tra:
      * Mã khóa học chưa tồn tại.
      * Ngày bắt đầu phải trước ngày kết thúc.
    - Lưu vào database → Thông báo "Tạo khóa học thành công".
  + Tạo học phần:
    - Người dùng chọn "Thêm học phần".
    - Nhập thông tin học phần (mã, tên, tín chỉ, chọn khóa học).
    - Hệ thống kiểm tra:
      * Mã học phần chưa tồn tại.
      * Khóa học đã được tạo trước đó.
    - Lưu vào database → Thông báo "Tạo học phần thành công".
  + Phân công giảng viên:
    - Người dùng chọn học phần → "Phân công giảng viên".
    - Chọn giảng viên từ danh sách (lọc theo bộ môn nếu có).
    - Hệ thống kiểm tra:
      * Giảng viên chưa bị trùng lịch (nếu đã có lịch dạy).
    - Lưu vào database → Thông báo "Phân công thành công".
* Luồng thay thế / Ngoại lệ (Alternative Flow):
  + Nếu mã khóa học/học phần đã tồn tại:
    - Hiển thị lỗi "Mã đã tồn tại, vui lòng nhập mã khác".
  + Nếu ngày kết thúc < ngày bắt đầu:
    - Hiển thị lỗi "Ngày kết thúc phải sau ngày bắt đầu".
  + Nếu giảng viên bị trùng lịch:
    - Hiển thị cảnh báo "Giảng viên đã có lịch dạy trong khoảng thời gian này" và gợi ý chọn giảng viên khác.
* Đầu ra (Output):
  + Thành công:
    - Khóa học/học phần được lưu vào database.
    - Hiển thị thông tin trên giao diện danh sách.
  + Thất bại:
    - Thông báo lỗi chi tiết (ví dụ: trùng mã, ngày không hợp lệ).
* Ràng buộc / Điều kiện tiên quyết:
  + Quyền truy cập: Chỉ quản trị viên/nhân viên đào tạo được thực hiện chức năng này.
  + Dữ liệu cần có trước:
    - Danh sách giảng viên (đồng bộ từ hệ thống khác).
    - Các khóa học phải được tạo trước khi thêm học phần.
* Thông báo / Phản hồi (UI Feedback):
  + Thành công:
    - Toast message: "Đã tạo khóa học [Tên khóa học] thành công!".
    - Popup xác nhận khi phân công giảng viên.
  + Lỗi:
    - Highlight ô input sai (màu đỏ) + tooltip ghi rõ lý do (ví dụ: "Mã học phần đã tồn tại").
* Rủi ro / Lỗi cần xử lý:
  + Mất kết nối database:
    - Hiển thị "Lỗi kết nối, vui lòng thử lại sau", tự động retry sau 5s.
  + File import giảng viên bị lỗi định dạng:
    - Thông báo "File không hợp lệ, vui lòng tải lại file mẫu" + cung cấp link download template.
  + Xóa nhầm khóa học đang có học phần:
    - Kiểm tra ràng buộc khóa ngoại → Thông báo "Không thể xóa vì khóa học đang chứa học phần".
  1. **Quản lý lớp học phần**
* Mã chức năng: FUNC-002
* Tên chức năng: Quản lý lớp học phần
* Mục tiêu:
  + Nhận dữ liệu sinh viên từ file Excel/JSON hoặc API.
  + Tự động phân lớp dựa trên số lượng sinh viên và học phần.
  + Quản lý thông tin lớp học phần (thêm, xóa, sửa, xem danh sách sinh viên).
* Đầu vào:
  + Nhập dữ liệu sinh viên:
    - File Excel/JSON chứa danh sách sinh viên (MSSV, họ tên, email, học phần đăng ký).
    - Hoặc dữ liệu từ API (nếu tích hợp với hệ thống khác).
  + Thông tin lớp học phần:
    - Mã lớp học phần (bắt buộc, duy nhất).
    - Học phần liên kết (chọn từ danh sách học phần hiện có).
    - Sĩ số tối đa (nếu có).
  + Thao tác thủ công
    - Thêm/xóa/sửa sinh viên trong lớp (nếu cần điều chỉnh).
* Luồng xử lý chính:
  + Nhập dữ liệu sinh viên:
    - Người dùng chọn "Nhập danh sách sinh viên" → Tải lên file Excel/JSON.
    - Hệ thống kiểm tra:
      * File hợp lệ (đúng định dạng, không bị lỗi).
      * Các trường bắt buộc (MSSV, họ tên) phải có.
    - Parse dữ liệu → Hiển thị preview → Xác nhận import.
  + Tự động phân lớp:
    - Hệ thống tự động:
      * Nhóm sinh viên cùng học phần thành các lớp.
      * Đảm bảo sĩ số không vượt quá giới hạn (nếu có).
      * Gán mã lớp tự động (ví dụ: LOP\_[Mã học phần]\_001).
  + Quản lý lớp học phần
    - Thêm lớp: Nhập thông tin lớp → Lưu vào database.
    - Sửa lớp: Chọn lớp → Chỉnh sửa thông tin (sĩ số, học phần) → Cập nhật.
    - Xóa lớp: Kiểm tra ràng buộc (nếu lớp đã có sinh viên → không cho xóa).
* Luồng xử lý thay thế:
  + Nếu file import không hợp lệ:
    - Hiển thị lỗi "File không đúng định dạng, vui lòng kiểm tra lại" + gợi ý file mẫu.
  + Nếu sinh viên đã có trong lớp khác cùng học phần:
    - Thông báo "Sinh viên [MSSV] đã được phân lớp trước đó" và bỏ qua.
  + Nếu sĩ số vượt quá giới hạn:
    - Tự động tạo lớp mới hoặc thông báo "Đã đủ sĩ số, không thể thêm sinh viên".
* Đầu ra:
  + Thành công:
    - Danh sách lớp học phần được tạo/cập nhật trong database.
    - File log ghi lại các sinh viên đã được phân lớp (hoặc bị bỏ qua do lỗi).
  + Thất bại:
    - Thông báo lỗi chi tiết (thiếu trường dữ liệu, trùng lặp, định dạng file sai).
* Ràng buộc / Điều kiện tiên quyết:
  + Quyền truy cập: Chỉ quản trị viên/nhân viên đào tạo được thực hiện.
  + Dữ liệu cần có trước:
    - Danh sách học phần đã được tạo (để liên kết với lớp).
    - Nếu dùng API, hệ thống phải kết nối được với nguồn dữ liệu sinh viên.
* Thông báo / Phản hồi:
  + Thành công:
    - Toast message: "Đã nhập danh sách sinh viên thành công!" hoặc "Đã tạo lớp [Mã lớp]!".
    - Hiển thị danh sách lớp và sinh viên đã phân bổ.
  + Lỗi:
    - Bảng thống kê lỗi sau khi import (ví dụ: "5/100 sinh viên không hợp lệ").
    - Popup cảnh báo khi xóa lớp có sinh viên.
* Rủi ro / Lỗi cần xử lý:
  + File Excel/JSON bị thiếu dữ liệu:
    - Tạm dừng import, highlight dòng lỗi, cho phép người dùng chỉnh sửa trước khi tiếp tục.
  + Mất kết nối API khi đồng bộ sinh viên:
    - Lưu tạm dữ liệu vào cache, thông báo "Đang đồng bộ, vui lòng chờ..." và tự động retry.
  + Xung đột dữ liệu (sinh viên thuộc nhiều lớp cùng học phần):
    - Ưu tiên giữ lớp cũ, thông báo "Sinh viên [MSSV] đã có trong lớp [Mã lớp khác]".
  1. **Lập thời khoá biểu**
* Mã chức năng: FUNC-003
* Tên chức năng: Lập thời khóa biểu
* Mục tiêu:
  + Tạo và quản lý lịch học, lịch thi cho các lớp học phần
  + Phân bổ phòng học, giảng viên, thời gian học hợp lý
  + Sắp xếp phòng thi, giám thị, thời gian thi
  + Tự động kiểm tra và cảnh báo xung đột lịch
* Đầu vào:
  + Thông tin lịch học:
    - Lớp học phần (chọn từ danh sách)
    - Giảng viên (chọn từ danh sách)
    - Phòng học (chọn từ danh sách phòng trống)
    - Thời gian (ngày, ca học, tuần áp dụng)
    - Học kỳ/năm học
  + Thông tin lịch thi:
    - Học phần (chọn từ danh sách)
    - Phòng thi (chọn từ danh sách)
    - Giám thị (chọn từ danh sách giảng viên)
    - Thời gian thi (ngày, giờ bắt đầu/kết thúc)
  + Ràng buộc:
    - Sức chứa phòng ≥ số lượng sinh viên
    - Giảng viên không bị trùng lịch
    - Phòng học không bị trùng lịch
* Luồng xử lý chính
  + Tạo lịch học:
    - Người dùng chọn "Tạo lịch học" → chọn lớp học phần
    - Hệ thống hiển thị danh sách giảng viên, phòng học khả dụng
    - Người dùng chọn giảng viên, phòng, thời gian
    - Hệ thống kiểm tra:
    - Phòng còn trống trong khung giờ được chọn
    - Giảng viên chưa có lịch dạy trùng
    - Phòng đủ sức chứa
    - Lưu lịch học → thông báo thành công
  + Tạo lịch thi:
    - Người dùng chọn "Tạo lịch thi" → chọn học phần
    - Hệ thống gợi ý phòng thi phù hợp (dựa trên số lượng sinh viên)
    - Người dùng chọn phòng, giám thị, thời gian thi
    - Hệ thống kiểm tra:
    - Phòng không bị trùng lịch thi khác
    - Giám thị không bị trùng lịch
    - Lưu lịch thi → thông báo thành công
  + Kiểm tra xung đột:
    - Hệ thống tự động quét xung đột khi:
    - Thêm mới lịch
    - Chỉnh sửa lịch hiện có
    - Hiển thị cảnh báo nếu phát hiện xung đột
* Luồng xử lý thay thế:
  + Nếu phòng đã được đặt:
    - Hiển thị thông báo "Phòng [tên phòng] đã có lịch trong khung giờ này"
    - Gợi ý các phòng khả dụng khác
  + Nếu giảng viên bị trùng lịch:
    - Hiển thị thông báo "Giảng viên [tên] đã có lịch dạy vào [thời gian]"
    - Gợi ý các khung giờ trống của giảng viên
  + Nếu sức chứa phòng không đủ:
    - Hiển thị thông báo "Phòng không đủ chỗ cho lớp học này"
    - Gợi ý các phòng có sức chứa lớn hơn
* Đầu ra
  + Thành công:
    - Lịch học/thi được lưu vào database
    - Xuất file PDF thời khóa biểu (nếu cần)
    - Gửi thông báo lịch đến giảng viên, sinh viên
  + Thất bại:
    - Báo lỗi chi tiết về nguyên nhân không thể tạo lịch
    - File log ghi lại các xung đột phát hiện được
* Ràng buộc / Điều kiện tiên quyết:
  + Quyền truy cập:
    - Chỉ quản trị viên/nhân viên đào tạo được tạo/chỉnh sửa lịch
    - Giảng viên chỉ được xem lịch của mình
  + Dữ liệu cần có trước:
    - Danh sách lớp học phần
    - Danh sách giảng viên
    - Danh sách phòng học với sức chứa
    - Học kỳ/năm học hiện tại
* Thông báo / Phản hồi:
  + Thành công:
    - Toast message: "Đã tạo lịch học thành công!"
    - Hiển thị màu sắc phân biệt các loại lịch trên giao diện
  + Cảnh báo:
    - Popup cảnh báo xung đột với nút "Xem chi tiết"
    - Highlight các ô lịch bị xung đột màu đỏ
  + Lỗi:
    - Tooltip giải thích lý do không thể lưu lịch
    - Bảng thống kê lỗi khi import hàng loạt
* Rủi ro / Lỗi cần xử lý:
  + Nhập liệu thủ công sai:
    - Validate ngay khi nhập: kiểm tra định dạng ngày giờ, khung giờ hợp lệ
  + Xung đột khi chỉnh sửa hàng loạt:
    - Cho phép preview các thay đổi trước khi áp dụng
    - Tạo bản backup trước khi thực hiện thay đổi lớn
  + Mất kết nối khi lưu:
    - Lưu tạm vào localStorage → đồng bộ khi có kết nối lại
    - Thông báo "Đang lưu dữ liệu cục bộ..."
  + Thay đổi dữ liệu nền:
    - Kiểm tra phiên bản dữ liệu khi lưu
    - Thông báo "Dữ liệu đã thay đổi, vui lòng làm mới trang"
  1. **Gửi thông báo**
* Mã chức năng: FUNC-004
* Tên chức năng: Quản lý gửi thông báo
* Mục tiêu:
  + Gửi thông báo tự động/nhân viên về lịch học, lịch thi đến giảng viên và sinh viên
  + Xử lý thông báo khẩn cấp (hủy lớp, thay đổi lịch)
  + Hỗ trợ đa kênh gửi (email, hệ thống nội bộ, SMS nếu có)
* Đầu vào:
  + Thông báo thường:
    - Loại thông báo (lịch học/lịch thi)
    - Đối tượng nhận (lớp học phần/giảng viên/cá nhân)
    - Nội dung (tự động sinh hoặc nhập tùy chỉnh)
  + Thông báo khẩn cấp:
    - Tiêu đề (vd: "HỦY LỚP NGÀY 20/10")
    - Nội dung chi tiết (lý do, thông tin thay đổi)
    - Mức độ ưu tiên (cao/trung bình/thấp)
  + Thiết lập gửi:
    - Kênh gửi (email/tin nội bộ/SMS)
    - Thời gian gửi (ngay lập tức/theo lịch)
* Luồng xử lý chính:
  + Gửi thông báo tự động (lịch học/thi):
    - Hệ thống tự động tạo thông báo khi:
      * Lịch học mới được tạo/cập nhật
      * Còn 24h trước khi buổi học/thi diễn ra
    - Nội dung tự động bao gồm:
      * Tên học phần, phòng, thời gian
      * Liên kết xem chi tiết (nếu có)
  + Gửi thông báo thủ công:
    - Người dùng chọn "Tạo thông báo" → Nhập tiêu đề/nội dung
    - Chọn đối tượng nhận (theo lớp/giảng viên/cá nhân)
    - Hệ thống kiểm tra:
      * Nội dung không trống
      * Đã chọn ít nhất 1 người nhận
    - Gửi → Lưu lịch sử thông báo
  + Thông báo khẩn cấp:
    - Bỏ qua hàng đợi ưu tiên → Gửi ngay lập tức
    - Đánh dấu là "QUAN TRỌNG" trong email/tin nhắn
* Luồng xử lý thay thế:
  + Nếu email không gửi được:
    - Thử lại sau 5 phút (tối đa 3 lần) → Chuyển sang gửi tin nội bộ
  + Nếu người nhận không có email:
    - Chỉ gửi qua tin nội bộ + cảnh báo "Không có email người nhận"
  + Khi hệ thống quá tải:
    - Ưu tiên gửi thông báo khẩn trước → Xếp hàng đợi cho loại khác
* Đầu ra:
  + Thành công:
    - Thông báo được gửi đến các kênh đã chọn
    - Báo cáo gửi (số lượng thành công/thất bại)
    - Lưu lịch sử thông báo (phục vụ tra cứu)
  + Thất bại:
    - Log lỗi chi tiết (email lỗi, lý do từ SMTP...)
    - Danh sách người nhận chưa được gửi
* Ràng buộc / Điều kiện tiên quyết:
  + Quyền truy cập:
    - Quản trị viên: Gửi mọi loại thông báo
    - Giảng viên: Chỉ gửi cho lớp mình phụ trách
  + Dữ liệu cần có trước:
    - Danh sách email/account người nhận hợp lệ
    - Cấu hình SMTP/SMS (nếu gửi qua email/SMS)
* Thông báo / Phản hồi:
  + Thành công:
    - Progress bar hiển thị tiến độ gửi
    - Popup "Đã gửi thành công đến 95% người nhận"
  + Cảnh báo:
    - Màu sắc khác nhau cho từng loại thông báo (vd: đỏ = khẩn cấp)
    - Icon cảnh báo khi chọn gửi đến quá nhiều người
  + Lỗi:
    - Bảng chi tiết lỗi (vd: "15 email không gửi được do sai định dạng")
    - Nút "Xuất báo cáo" để kiểm tra sau
* Rủi ro / Lỗi cần xử lý:
  + Gửi nhầm đối tượng:
    - Thêm bước xác nhận khi gửi đến >100 người
    - Cho phép "Hoàn tác" trong 5 phút đầu
  + Spam thông báo:
    - Giới hạn số lượng gửi/ngày với từng user
    - Cảnh báo khi phát hiện nội dung trùng lặp
  + Lỗi kết nối dịch vụ email/SMS:
    - Tự động chuyển sang kênh dự phòng
    - Thông báo "Hệ thống đang bảo trì, vui lòng gửi lại sau"
  + Thông báo bị đánh dấu spam:
    - Kiểm tra nội dung (tránh từ khóa spam)
    - Thêm tùy chọn "Đánh dấu quan trọng" với email
  1. **Nhận dữ liệu từ hệ thống khác**
* Mã chức năng: FUNC-005
* Tên chức năng: Đồng bộ dữ liệu từ hệ thống ngoài
* Mục tiêu:
  + Nhận và xử lý dữ liệu sinh viên/giảng viên từ file Excel/JSON
  + Đồng bộ dữ liệu tự động qua API với hệ thống SIS
  + Đảm bảo tính toàn vẹn và nhất quán dữ liệu
* Đầu vào:
  + File upload:
    - Định dạng: Excel (.xlsx, .xls) hoặc JSON
    - Trường bắt buộc:
      * Sinh viên: MSSV, họ tên, lớp, email
      * Giảng viên: MSGV, họ tên, bộ môn, email
  + API Integration:
    - Endpoint API từ hệ thống SIS
    - Tham số: học kỳ, năm học, loại dữ liệu (sinh viên/giảng viên)
    - Authentication: Token hoặc API Key
  + Cấu hình đồng bộ:
    - Tần suất đồng bộ (hàng ngày/tuần)
    - Loại dữ liệu cần đồng bộ
    - Chế độ ghi đè/cập nhật
* Luồng xử lý chính:
  + Nhập dữ liệu từ file:
    - Người dùng chọn "Nhập dữ liệu" → Upload file
    - Hệ thống kiểm tra:
      * Định dạng file hợp lệ
      * Cấu trúc file đúng chuẩn
    - Parse dữ liệu → Hiển thị preview
    - Xác nhận import → Lưu vào database
  + Đồng bộ qua API:
    - Hệ thống tự động gọi API theo lịch cấu hình
    - Nhận response → Xử lý dữ liệu:
      * Thêm mới nếu chưa tồn tại
      * Cập nhật nếu có thay đổi
    - Ghi log quá trình đồng bộ
  + Xử lý xung đột dữ liệu:
    - Phát hiện bản ghi trùng (dựa trên MSSV/MSGV)
    - Áp dụng quy tắc cập nhật (ưu tiên dữ liệu mới/giữ nguyên bản cũ)
* Luồng xử lý thay thế:
  + Nếu file lỗi định dạng:
    - Dừng xử lý → Thông báo "File không đúng định dạng"
    - Gợi ý template mẫu
  + Nếu thiếu trường bắt buộc:
    - Đánh dấu bản ghi lỗi → Cho phép bỏ qua hoặc dừng import
  + Nếu API không phản hồi:
    - Thử lại 3 lần → Chuyển sang chế độ offline
    - Đánh dấu cần đồng bộ lại sau
  + Nếu dữ liệu xung đột:
    - Tạo báo cáo xung đột → Chờ xử lý thủ công
* Đầu ra:
  + Thành công:
    - Dữ liệu mới được cập nhật vào hệ thống
    - Báo cáo đồng bộ (số bản ghi thêm/cập nhật)
    - File log chi tiết
  + Thất bại:
    - Báo cáo lỗi chi tiết
    - File chứa các bản ghi lỗi (nếu import từ file)
* Ràng buộc / Điều kiện tiên quyết:
  + Quyền truy cập:
    - Chỉ Admin/Nhân viên đào tạo được thực hiện
    - API cần xác thực token/API Key
  + Yêu cầu hệ thống:
    - API SIS phải hoạt động và trả về đúng chuẩn JSON
    - File upload phải <10MB
* Thông báo / Phản hồi:
  + Thành công:
    - Progress bar hiển thị tiến độ xử lý
    - Popup "Đã nhập thành công 150/160 bản ghi"
  + Cảnh báo:
    - Màu vàng cho các bản ghi cần kiểm tra
    - Tooltip giải thích lỗi từng dòng
  + Lỗi:
    - Bảng chi tiết lỗi có thể export
    - Nút "Xem bản ghi lỗi" để chỉnh sửa
* Rủi ro / Lỗi cần xử lý:
  + Dữ liệu nhập bị mã hóa sai:
    - Tự động detect encoding → Chuyển đổi sang UTF-8
    - Cảnh báo "File có thể bị lỗi font"
  + API thay đổi cấu trúc:
    - Kiểm tra version API → Thông báo "Cần cập nhật cấu hình API"
    - Lưu cache dữ liệu cũ
  + Mất kết nối khi đang đồng bộ:
    - Lưu trạng thái tạm → Tự động tiếp tục khi có kết nối
    - Đánh dấu các bản ghi chưa đồng bộ
  + Dữ liệu trùng nhưng khác thông tin:
    - Tạo task xử lý thủ công → Thông báo cho admin
    - Gợi ý phương án hợp nhất dữ liệu

1. **YÊU CẦU PHI CHỨC NĂNG**
   1. **Hiệu năng**

* Hệ thống hỗ trợ tối thiểu 500 người dùng đồng thời.
* Thời gian phản hồi < 2 giây cho các thao tác thông thường.
  1. **Bảo mật**
* Xác thực người dùng: Đăng nhập bằng tài khoản/mật khẩu, hỗ trợ OAuth2.
* Phân quyền: RBAC (Role-Based Access Control) để kiểm soát truy cập.
* Mã hóa dữ liệu nhạy cảm: SSL/TLS cho truyền tải, mã hóa database.
  1. **Khả năng mở rộng**
* Kiến trúc microservice: Dễ dàng mở rộng từng module.
* Hỗ trợ RESTful API: Để tích hợp với hệ thống khác.

1. **GIAO DIỆN CHO CÁC CHỨC NĂNG DỰ KIẾN**
   1. **Nhóm giao diện Quản trị viên**

* Dashboard tổng quan
  + Thống kê số lượng khóa học, lớp học, sinh viên
  + Biểu đồ hoạt động hệ thống
* Quản lý khóa học/học phần
  + Danh sách khóa học
  + Form thêm/sửa khóa học
  + Quản lý học phần thuộc khóa
* Quản lý lớp học phần
  + Danh sách lớp học
  + Form phân lớp tự động/thủ công
  + Giao diện import file Excel/JSON
* Quản lý thời khóa biểu
  + Lịch học dạng tuần/tháng
  + Form tạo lịch học/lịch thi
  + Kiểm tra xung đột lịch
* Gửi thông báo
  + Form soạn thông báo
  + Lịch sử gửi thông báo
  + Cài đặt template thông báo
  1. **Nhóm giao diện Giảng viên**
* Dashboard cá nhân
  + Lịch giảng dạy tuần
  + Thông báo mới
* Quản lý lớp phụ trách
  + Danh sách sinh viên
  + Điểm danh/bảng điểm
* Xem thời khóa biểu
  + Lịch dạy theo tuần
  + Lịch thi
  1. **Nhóm giao diện Học viên**
* Dashboard cá nhân
  + Lịch học/tin nhắn mới
  + Thông báo quan trọng
* Xem thời khóa biểu
  + Lịch học theo tuần
  + Lịch thi
* Xem điểm/hồ sơ
  + Kết quả học tập
  + Thông tin cá nhân
  1. **Nhóm giao diện hệ thống**
* Đăng nhập/Phân quyền
  + Form đăng nhập
  + Quên mật khẩu
* Cấu hình hệ thống
  + Kết nối API
  + Cài đặt đồng bộ dữ liệu
  + Quản lý template
* Báo cáo
  + Xuất báo cáo PDF/Excel
  + Thống kê dữ liệu

1. **TÀI LIỆU LIÊN QUAN & TÍCH HỢP**

* Đồng bộ dữ liệu học viên thông qua API của hệ thống quản lý sinh viên.
* Hệ thống email/SMS.
* Tài liệu API: Swagger/OpenAPI cho tích hợp bên thứ ba.

***Phiên bản tài liệu: 1.0***

***Ngày phát hành: 22/05/2025***

***Người phê duyệt: Lê Thị Diễm Kiều***