ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM

ĐỀ TÀI:

XÂY DỰNG WEBSITE QUẢN LÝ VÀ SẮP XẾP THỜI KHOÁ BIỂU, PHÒNG HỌC TẠI TRƯỜNG ĐẠI HỌC

Người hướng dẫn: **TS. ĐẶNG THIÊN BÌNH**
Sinh viên thực hiện: **HÀ NGỌC HƯNG**
Số thẻ sinh viên: **102210098**
Lớp**:** **21TCLC_DT1**

Đà Nẵng, 06/2026

TÓM TẮT

Tên đề tài: Website quản lý và sắp xếp thời khoá biểu, phòng học tại trường đại học
Sinh viên thực hiện: Hà Ngọc Hưng
Số thẻ SV: 102210098 Lớp: 21TCLC_DT1
Hiện nay, công tác xây dựng thời khóa biểu tại các trường đại học là một nhiệm vụ phức tạp do phải đáp ứng đồng thời nhiều ràng buộc liên quan đến giảng viên, phòng học, lớp học phần, thời gian giảng dạy và cơ sở vật chất. Tại nhiều đơn vị, việc lập thời khóa biểu vẫn còn thực hiện thủ công hoặc bán tự động, dẫn đến mất nhiều thời gian, dễ phát sinh xung đột và khó tối ưu việc sử dụng nguồn lực.
Đề tài hướng đến việc xây dựng hệ thống DUT Smart Schedule nhằm hỗ trợ quản lý và tự động hóa quá trình lập thời khóa biểu cho Trường Đại học Bách khoa – Đại học Đà Nẵng. Hệ thống cho phép quản lý tập trung dữ liệu giảng viên, học phần, lớp học phần, phòng học và các học kỳ đào tạo. Đồng thời, hệ thống ứng dụng các thuật toán tối ưu để tự động phân công lịch học phù hợp, đảm bảo các ràng buộc về giảng viên, phòng học, sức chứa và thời gian học, từ đó giảm thiểu xung đột và nâng cao hiệu quả sử dụng tài nguyên.
Ngoài chức năng lập thời khóa biểu tự động, hệ thống còn cung cấp các công cụ quản trị, thống kê và tra cứu, giúp cán bộ quản lý dễ dàng theo dõi tình trạng sử dụng phòng học, khối lượng giảng dạy của giảng viên và kết quả tối ưu thời khóa biểu. Hệ thống cũng hỗ trợ giảng viên và sinh viên tra cứu lịch học một cách nhanh chóng và thuận tiện.
Như vậy, mục tiêu và nhiệm vụ chính của đề tài là xây dựng một hệ thống quản lý và tối ưu thời khóa biểu thông minh, góp phần đẩy mạnh chuyển đổi số trong công tác quản lý đào tạo, giảm khối lượng công việc thủ công và nâng cao chất lượng quản lý học vụ tại nhà trường.
Tóm tắt nội dung của báo cáo:

- Mở đầu: Giới thiệu đề tài, phân tích hiện trạng công tác xây dựng thời khóa biểu và phân bổ phòng học tại Trường Đại học Bách khoa – Đại học Đà Nẵng; trình bày lý do lựa chọn đề tài, mục tiêu nghiên cứu, đối tượng nghiên cứu, phạm vi nghiên cứu và các công nghệ được sử dụng trong quá trình thực hiện.
- Chương 1. Cơ sở lý thuyết: Trình bày các kiến thức nền tảng liên quan đến bài toán lập thời khóa biểu trong môi trường đại học; các khái niệm về lớp học phần, giảng viên, phòng học và thời khóa biểu; các loại ràng buộc trong bài toán xếp lịch; tổng quan về đề tài cấp Bộ CB2005 và định hướng kế thừa trong

đề tài. Chương này cũng trình bày phương pháp tối ưu hóa được sử dụng, bao gồm Google OR-Tools, bộ giải CP-SAT và các công nghệ phục vụ việc xây dựng hệ thống.

- Chương 2. Phân tích và thiết kế hệ thống: Trình bày kết quả khảo sát thực tế tại Trường Đại học Bách khoa – Đại học Đà Nẵng; phân tích nghiệp vụ của các nhóm người dùng; xác định yêu cầu chức năng và phi chức năng của hệ thống; xây dựng các biểu đồ Use Case, biểu đồ hoạt động, đặc tả các chức năng chính và thiết kế cơ sở dữ liệu phục vụ công tác quản lý và sắp xếp thời khóa biểu.
- Chương 3. Triển khai và đánh giá kết quả: Trình bày quá trình xây dựng hệ thống DUT Smart Schedule, các công nghệ được sử dụng, kiến trúc phần mềm, thiết kế cơ sở dữ liệu và các chức năng đã triển khai. Chương này cũng mô tả mô hình bài toán xếp thời khóa biểu, các ràng buộc được áp dụng, phương pháp tối ưu hóa bằng bộ giải CP-SAT, kết quả thực nghiệm trên dữ liệu thực tế của nhà trường và đánh giá hiệu quả hoạt động của hệ thống.
- Kết luận và hướng phát triển: Tổng kết các kết quả đạt được trong quá trình nghiên cứu và xây dựng hệ thống; đánh giá những hạn chế còn tồn tại; đồng thời đề xuất các hướng phát triển trong tương lai như nâng cao chất lượng tối ưu hóa thời khóa biểu, hỗ trợ xếp lịch thi, tích hợp với các hệ thống đào tạo của nhà trường và mở rộng khả năng ứng dụng thực tế.

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Hà Ngọc Hưng Số thẻ sinh viên: 102210098

Lớp: 21TCLC_DT1 Khoa: Công nghệ thông tin Ngành: Công nghệ phần mềm

1. *Tên đề tài đồ án:*

Website quản lý và sắp xếp thời khoá biểu, phòng học tại trường đại học

2. *Đề tài thuộc diện:* ☐ *Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện*

3. *Các số liệu và dữ liệu ban đầu:*

Dữ liệu về khoa, phòng học, kế hoạch học tập, lớp học phần, danh sách giảng viên do trường Đại học Bách Khoa – Đại học Đà Nẵng cung cấp trước khi nhập vào hệ thống

4. *Nội dung các phần thuyết minh và tính toán:*

Mở đầu: Giới thiệu về đề tài, hiện trạng thực tế và lý do thực hiện đề tài, đồng thời giới thiệu sơ lược về mục tiêu cần đạt được, các tính năng chính và đối tượng của đề tài.

Chương 1. Cơ sở lý thuyết: Trình bày các kiến thức nền tảng và các công nghệ được sử dụng liên quan đến bài toán lập thời khóa biểu trong môi trường đại học.

Chương 2. Phân tích và thiết kế hệ thống: Phân tích nghiệp vụ của các nhóm người dùng; trình bày các sơ đồ phân tích và thiết kế, xây dựng luồng hoạt động của website

Chương 3. Triển khai và đánh giá kết quả: Trình bày quá trình xây dựng website DUT SmartSchedule; mô tả các chức năng đã triển khai; đánh giá kết quả triển khai trên dữ liệu thực tế của nhà trường và khả năng đáp ứng các yêu cầu nghiệp vụ.

Kết luận và hướng phát triển: Đánh giá kết quả đạt được và định hướng phát triển.

5. *Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):*

Không

6. *Họ tên người hướng dẫn:* TS. Đặng Thiên Bình

7. *Ngày giao nhiệm vụ đồ án:* *09/03/2026*

8. *Ngày hoàn thành đồ án:* *14/06/2026*

Đà Nẵng, ngày 05 tháng 06 năm 2026

Trưởng khoa CNTT **Người hướng dẫn**

TS. Đặng Thiên Bình

LỜI NÓI ĐẦU

Công tác xây dựng thời khóa biểu giảng dạy là một trong những nhiệm vụ quan trọng trong hoạt động đào tạo của các trường đại học. Một thời khóa biểu hợp lý không chỉ giúp bảo đảm tiến độ giảng dạy, học tập mà còn tạo điều kiện thuận lợi cho giảng viên, sinh viên và các đơn vị quản lý trong quá trình tổ chức đào tạo. Tuy nhiên, với số lượng lớn lớp học phần được mở trong mỗi học kỳ cùng nhiều ràng buộc liên quan đến giảng viên, phòng học và thời gian giảng dạy, việc xây dựng thời khóa biểu luôn là một bài toán phức tạp.

Qua quá trình tìm hiểu thực tế tại Trường Đại học Bách khoa - Đại học Đà Nẵng, nhóm nhận thấy dữ liệu phục vụ công tác lập thời khóa biểu được lưu trữ từ nhiều nguồn khác nhau như danh sách giảng viên, lớp học phần, phòng học và kế hoạch đào tạo. Việc tổng hợp, cập nhật và xử lý các dữ liệu này thường tốn nhiều thời gian. Bên cạnh đó, khi thực hiện sắp xếp thủ công hoặc phụ thuộc nhiều vào kinh nghiệm của người quản lý, quá trình lập thời khóa biểu có thể phát sinh các trường hợp trùng lịch giảng dạy, chưa khai thác hiệu quả phòng học hoặc mất nhiều thời gian điều chỉnh khi dữ liệu thay đổi.

Xuất phát từ những vấn đề trên, em lựa chọn thực hiện đề tài "Xây dựng website quản lý và sắp xếp thời khoá biểu, phòng học tại trường đại học". Đề tài hướng đến việc xây dựng một hệ thống quản lý tập trung, hỗ trợ lưu trữ và quản lý dữ liệu giảng viên, phòng học, lớp học phần, đồng thời hỗ trợ tự động sắp xếp thời khóa biểu dựa trên các ràng buộc và tiêu chí tối ưu đã được xác định.

Trong quá trình thực hiện, em đã tiến hành khảo sát nhu cầu thực tế, nghiên cứu các phương pháp giải quyết bài toán lập thời khóa biểu và xây dựng hệ thống dưới dạng ứng dụng web. Hệ thống được phát triển với các chức năng như quản lý giảng viên, quản lý phòng học, quản lý lớp học phần, quản lý tài khoản người dùng, nhập dữ liệu từ tệp Excel, thống kê dữ liệu đào tạo và hỗ trợ tối ưu hóa thời khóa biểu. Ngoài ra, hệ thống còn tích hợp cơ chế đăng nhập bằng tài khoản Microsoft nhằm tạo thuận lợi cho người dùng trong quá trình sử dụng.

Đề tài là cơ hội để em vận dụng những kiến thức đã được học trong quá trình đào tạo như phân tích và thiết kế hệ thống, cơ sở dữ liệu, phát triển ứng dụng web, xây dựng API và tối ưu hóa bài toán. Đồng thời, quá trình thực hiện cũng giúp nhóm rèn luyện kỹ năng lập kế hoạch, nghiên cứu tài liệu, giải quyết vấn đề và triển khai một hệ thống phần mềm hoàn chỉnh.

Em xin gửi lời cảm ơn đến quý thầy cô Khoa Công nghệ thông tin, Trường Đại học Bách khoa - Đại học Đà Nẵng đã tận tình giảng dạy, truyền đạt cho em những kiến thức chuyên môn cùng các kỹ năng cần thiết trong suốt quá trình học tập và rèn luyện tại trường.

Đặc biệt, em xin bày tỏ lòng biết ơn sâu sắc đến giảng viên hướng dẫn TS. Đặng Thiên Bình đã tận tình định hướng và đồng hành cùng em từ giai đoạn lựa chọn đề tài đến quá trình xây dựng và hoàn thiện đồ án. Trong suốt thời gian thực hiện đề tài, thầy thường xuyên tổ chức các buổi trao đổi, báo cáo tiến độ, giải đáp những khó khăn và đưa ra nhiều góp ý chuyên môn quý báu. Những ý kiến đóng góp của thầy đã giúp em từng bước hoàn thiện hệ thống và đảm bảo tiến độ thực hiện đồ án.

Em xin chân thành cảm ơn thầy Trần Văn Huynh, chuyên viên Phòng Đào tạo, đã hỗ trợ cung cấp thông tin, dữ liệu thực tế và chia sẻ các quy trình nghiệp vụ liên quan đến công tác lập thời khóa biểu. Những góp ý và phản hồi từ thầy là cơ sở quan trọng giúp em xây dựng hệ thống phù hợp hơn với nhu cầu thực tế của nhà trường.

Bên cạnh đó, em xin gửi lời cảm ơn đến các anh, các bạn thuộc Tổ Công nghệ thông tin đã nhiệt tình chia sẻ kinh nghiệm thực tế, hỗ trợ giải đáp các vấn đề phát sinh trong quá trình phát triển hệ thống, giúp em có thêm kiến thức và kinh nghiệm để hoàn thiện sản phẩm.

Mặc dù đã nỗ lực hoàn thành đề tài trong phạm vi thời gian và điều kiện cho phép, đồ án khó tránh khỏi những thiếu sót nhất định. Em rất mong nhận được những ý kiến đóng góp và nhận xét từ quý thầy cô để tiếp tục hoàn thiện đề tài.

Một lần nữa, em xin chân thành cảm ơn.

CAM ĐOAN

Em xin cam đoan đồ án tốt nghiệp với tên đề tài: “Xây dựng website quản lý và sắp xếp thời khoá biểu, phòng học tại trường đại học” là công trình nghiên cứu của bản thân dưới sự hướng dẫn trực tiếp của giảng viên TS. Đặng Thiên Bình. Các tài liệu được tham khảo để sử dụng làm cơ sở lý thuyết trong báo cáo là có nguồn gốc và được trích dẫn rõ ràng trong mục tài liệu tham khảo. Em xin hoàn toàn chịu trách nhiệm về tính xác thực và nguyên bản của báo cáo.

Sinh viên thực hiện

Hà Ngọc Hưng

MỤC LỤC

TÓM TẮT .....
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP .....
LỜI NÓI ĐẦU VÀ CẢM ƠN .....
CAM ĐOAN .....
MỤC LỤC .....
DANH SÁCH CÁC BẢNG .....
DANH SÁCH CÁC HÌNH VẼ .....
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT .....
MỞ ĐẦU 1
1. Phân tích hiện trạng 1
2. Mục tiêu của đề tài 2
3. Đối tượng và phạm vi của đề tài 3
4. Công nghệ sử dụng 3
5. Cấu trúc của đồ án tốt nghiệp 4
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 5
1.1 Một số khái niệm chính 5
1.1.1 Thời khoá biểu đại học 5
1.1.2 Lớp học phần 5
1.1.3 Giảng viên 6
1.1.4 Phòng học 6
1.1.5 Bài toán lập thời khóa biểu 6
1.2 Bài toán lập thời khóa biểu trong môi trường đại học 7
1.2.1 Đặc điểm bài toán 7
1.2.2 Các ràng buộc trong bài toán 7
1.3 Phương pháp tối ưu hóa được sử dụng trong đề tài 8
1.3.1 Constraint Programming 8
1.3.2 Google OR-Tools 8
1.3.3 Bộ giải CP-SAT 9
1.3.4 Hàm mục tiêu tối ưu 11
1.4 Các công nghệ sử dụng 11
1.4.1 ASP.NET Core 11
1.4.2 ReactJS 11
1.4.3 Microsoft SQL Server 12
1.4.4 JWT Authentication 12

1.4.5 Microsoft Authentication 12
1.5 Tổng quan các nghiên cứu và hướng tiếp cận liên quan 13
1.6 Mô hình tổng quát của bài toán lập thời khóa biểu 14
1.7 Các tập hợp và thông số sử dụng trong mô hình 15
1.8 Biến quyết định 16
1.9 Ràng buộc cứng trong bài toán lập thời khóa biểu 16
1.10 Ràng buộc mềm trong bài toán lập thời khóa biểu 17
1.11 Hàm mục tiêu 19
CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 20
2.1 Khảo sát và phân tích nghiệp vụ 20
2.1.1 Khảo sát quy trình nghiệp vụ hiện tại 20
2.1.2 Nghiệp vụ của Quản trị viên (Admin) 21
2.1.3 Nghiệp vụ của Cán bộ quản lý đào tạo 21
2.1.4 Nghiệp vụ của Giảng viên 22
2.2 Yêu cầu hệ thống 22
2.2.1 Yêu cầu chức năng 22
2.2.2 Yêu cầu phi chức năng 24
2.3 Tổng quan nguyên lý hệ thống 25
2.4 Use case của hệ thống 27
2.4.1 Biểu đồ use case tổng quan hệ thống 27
2.4.2 Biểu đồ use case phân rã quản lý tài khoản (Quản trị viên) 28
2.4.3 Biểu đồ use case phân rã quản lý giảng viên 29
2.4.4 Biểu đồ use case phân rã quản lý phòng học 30
2.4.5 Biểu đồ use case phân rã quản lý lớp học phần 31
2.4.6 Biểu đồ use case phân rã xếp thời khoá biểu học kỳ 32
2.4.7 Biểu đồ use case phân rã giảng viên xem lịch dạy 33
2.5 Quy trình nghiệp vụ 34
2.5.1 Quy trình nghiệp vụ đăng nhập và chọn vai trò 34
2.5.2 Quy trình nghiệp vụ chạy thuật toán sắp xếp TKB theo học kỳ 36
2.5.3 Quy trình nghiệp vụ cập nhật lớp học phần và khoá lịch 38
2.5.4 Quy trình xử lý khi không tìm được lời giải 39
2.6 Đặc tả chức năng 40
2.6.1 Đặc tả use case đăng nhập hệ thống 40
2.6.2 Đặc tả use case quản lý tài khoản 41
2.6.3 Đặc tả use case Quản lý lớp học phần 42

2.6.4 Các trường thông tin lớp học phần (CourseSection) 44
2.6.5 Đặc tả use case sắp xếp TKB theo học kỳ 45
2.6.6 Đặc tả use case giảng viên xem thời khoá biểu 47
2.6.7 Đặc tả use case Cấu hình tham số xếp lịch 48
2.7 Thiết kế cơ sở dữ liệu 49
2.7.1 Mô hình quan hệ thực thể (ERD) 49
2.7.2 Thiết kế dữ liệu phục vụ thuật toán 50
2.7.3 Thiết kế bảng Faculty 51
2.7.4 Thiết kế bảng Lecturer 51
2.7.5 Thiết kế bảng Classroom 52
2.7.6 Thiết kế bảng CourseSection 52
2.7.7 Thiết kế bảng Timetable 53
2.7.8 Thiết kế bảng Account, Role và AccountRole 53
2.7.9 Thiết kế bảng SystemConfig 54
CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 55
3.1 Mô tả chức năng đã đạt được 55
3.1.1 Chức năng đăng nhập 56
3.1.2 Chức năng quản lý tài khoản 57
3.1.3 Chức năng quản lý giảng viên 58
3.1.4 Chức năng quản lý phòng học 59
3.1.5 Chức năng quản lý lớp học phần 60
3.1.6 Chức năng quản lý khoa 61
3.1.7 Dashboard thống kê 61
3.1.8 Chức năng cấu hình hệ thống 63
3.1.9 Chức năng xếp thời khóa biểu 64
3.1.10 Chức năng giảng viên xem lịch dạy 65
3.2 Thuật toán xếp thời khóa biểu 66
3.2.1 Quy trình triển khai thuật toán xếp thời khóa biểu 66
3.2.2 Kiến trúc phân rã bài toán 68
3.2.3 Chiến lược nới lỏng và thử lại 68
3.2.4 Mô hình bài toán 68
3.2.5 Biến quyết định 69
3.2.6 Mô hình hóa ràng buộc cứng trong hệ thống 69
3.2.7 Mô hình hoá các ràng buộc mềm trong hệ thống 73
3.2.8 Phân tích độ phức tạp 76

3.2.9 Quy trình giải bằng CP-SAT 77
3.3 Kết quả thực nghiệm 78
3.3.1 Dữ liệu thực nghiệm 78
3.3.2 Kết quả xếp thời khóa biểu 79
3.3.3 Đánh giá chất lượng thời khóa biểu 80
3.3.4 Đánh giá hiệu năng hệ thống 81
3.4 Đánh giá kết quả đạt được 82
3.4.1 Kết quả đạt được 82
3.4.2 Hạn chế 82
3.4.3 Hướng phát triển 83
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 84
TÀI LIỆU THAM KHẢO 85
PHỤ LỤC 86

DANH SÁCH CÁC BẢNG

Bảng 1.1. Phân loại ràng buộc trong bài toán lập thời khóa biểu....8
Bảng 1.2. Các công nghệ sử dụng trong hệ thống....13
Bảng 1.3. Các tập hợp và thông số sử dụng trong mô hình....15
Bảng 2.1. Yêu cầu chức năng của hệ thống....22
Bảng 2.2. Yêu cầu phi chức năng của hệ thống....24
Bảng 2.3. Đặc tả use case đăng nhập hệ thống....40
Bảng 2.4. Đặc tả use case quản lý tài khoản....41
Bảng 2.5. Đặc tả use case quản lý lớp học phần....42
Bảng 2.6. Các trường dữ liệu của thực thể CourseSection....44
Bảng 2.7. Đặc tả use case sắp xếp thời khóa biểu theo học kỳ....45
Bảng 2.8. Đặc tả use case giảng viên xem thời khóa biểu....47
Bảng 2.9. Đặc tả use case cấu hình tham số xếp lịch....48
Bảng 2.10. Các nhóm dữ liệu đầu vào phục vụ thuật toán xếp thời khóa biểu....50
Bảng 2.11. Thiết kế bảng Faculty....51
Bảng 2.12. Thiết kế bảng Lecturer....51
Bảng 2.13. Thiết kế bảng Classroom....52
Bảng 2.14. Thiết kế bảng CourseSection....52
Bảng 2.15. Thiết kế bảng Timetable....53
Bảng 2.16. Thiết kế các bảng Account, Role và AccountRole....53
Bảng 2.17. Thiết kế bảng SystemConfig....54
Bảng 3.1. Bảng tóm tắt các ràng buộc mềm....74
Bảng 3.2. Thông số dữ liệu thực nghiệm....76
Bảng 3.3. Kết quả xếp thời khóa biểu học kỳ....77
Bảng 3.4. Kết quả xếp thời khóa biểu theo từng học kỳ....77
Bảng 3.5. Kết quả đánh giá chất lượng thời khóa biểu....78
Bảng 3.6. Kết quả đánh giá hiệu năng hệ thống....79

DANH SÁCH CÁC HÌNH VẼ

Hình 1.1. Mô hình giải bài toán lập thời khóa biểu bằng bộ giải CP-SAT 10
Hình 2.1. Kiến trúc tổng quan của hệ thống DUT Smart Schedule 25
Hình 2.2. Biểu đồ use case tổng quan hệ thống 27
Hình 2.3. Biểu đồ use case phân rã quản lý tài khoản 28
Hình 2.4. Biểu đồ use case phân rã quản lý giảng viên 29
Hình 2.5. Biểu đồ use case phân rã quản lý phòng học 30
Hình 2.6. Biểu đồ use case phân rã quản lý lớp học phần 31
Hình 2.7. Biểu đồ use case phân rã xếp thời khoá biểu học kỳ 32
Hình 2.8. Biểu đồ use case phân rã giảng viên xem lịch dạy 33
Hình 2.9. Quy trình nghiệp vụ đăng nhập và chọn vai trò 34
Hình 2.10. Quy trình nghiệp vụ sắp xếp TKB theo học kỳ 36
Hình 2.11. Quy trình nghiệp vụ cập nhật lớp học phần và khoá lịch 38
Hình 2.12. Mô hình thực thể liên kết (ERD) của hệ thống DUT Smart Schedule 49
Hình 3.1. Giao diện đăng nhập hệ thống 56
Hình 3.2. Giao diện quản lý tài khoản 57
Hình 3.3. Giao diện quản lý giảng viên 58
Hình 3.4. Giao diện quản lý phòng học 59
Hình 3.5. Giao diện quản lý lớp học phần 60
Hình 3.6. Giao diện quản lý khoa 61
Hình 3.7. Dashboard thống kê dữ liệu 62
Hình 3.8. Giao diện cấu hình hệ thống 63
Hình 3.9. Giao diện xếp thời khóa biểu học kỳ 64
Hình 3.10. Giao diện giảng viên xem lịch dạy 65
Hình 3.11. Quy trình sắp xếp TKB theo học kỳ 67
Hình 3.12. Quy trình xây dựng thời khóa biểu bằng bộ giải CP-SAT 77

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Ký hiệu:

Ký hiệu	Diễn giải
HC	Hard Constraint – Ràng buộc cứng
SC	Soft Constraint – Ràng buộc mềm
C	Tập các lớp học phần cần xếp lịch
R	Tập các phòng học có thể sử dụng
T	Tập các khung thời gian trong tuần
L	Tập các giảng viên
D	Tập các ngày học trong tuần
B	Tập các buổi học trong ngày
c	Một lớp học phần thuộc tập C
r	Một phòng học thuộc tập R
t	Một khung thời gian thuộc tập T
l	Một giảng viên thuộc tập L
x[c, r, t]	Biến quyết định, nhận giá trị 1 nếu lớp học phần c được xếp vào phòng r tại khung thời gian t, ngược lại nhận giá trị 0
Capacity(r)	Sức chứa của phòng học r
StudentCount(c)	Số lượng sinh viên của lớp học phần c
lecturer(c)	Giảng viên phụ trách lớp học phần c
type(c)	Loại lớp học phần c
type(r)	Loại phòng học r
day(t)	Ngày tương ứng với khung thời gian t
session(t)	Buổi học tương ứng với khung thời gian t
P1	Chi phí phạt do lớp học được xếp vào thứ bảy
P2	Chi phí phạt do giảng viên phải lên trường nhiều ngày

P3	Chi phí phạt do có khoảng trống trong lịch dạy của giảng viên
P4	Chi phí phạt do giảng viên dạy kín cả ngày
P5	Chi phí phạt do phòng học có sức chứa chênh lệch lớn so với sĩ số lớp
P6	Chi phí phạt do lớp PFIEV không được xếp vào khu P
P7	Chi phí phạt do lớp Ngoại ngữ không được xếp vào khu B tầng 1 hoặc tầng 2
w1, w2, ..., w7	Trọng số tương ứng với từng thành phần chi phí phạt trong hàm mục tiêu
Z	Giá trị hàm mục tiêu cần tối thiểu hóa

Chữ viết tắt:

Chữ viết tắt		
Viết tắt	Nội dung đầy đủ	Diễn giải
TKB	Thời khóa biểu	Kế hoạch bố trí thời gian học tập và giảng dạy
API	Application Programming Interface	Giao diện lập trình ứng dụng cho phép các hệ thống giao tiếp với nhau
JSON	JavaScript Object Notation	Định dạng dữ liệu thường dùng khi trao đổi giữa Frontend và Backend
SSO	Single Sign-On	Cơ chế đăng nhập một lần cho nhiều hệ thống
MSAL	Microsoft Authentication Library	Thư viện hỗ trợ xác thực tài khoản Microsoft
CP	Constraint Programming	Phương pháp lập trình ràng buộc dùng trong tối ưu hóa
SAT	Boolean Satisfiability	Bài toán thỏa mãn logic Boolean

CP-SAT	Constraint Programming Satisfiability	Bộ giải tối ưu hóa ràng buộc của Google OR-Tools
OR-Tools	Operations Research Tools	Bộ công cụ tối ưu hóa và nghiên cứu tác nghiệp của Google
EF Core	Entity Framework Core	Công cụ ORM được sử dụng trong hệ sinh thái .NET
ERD	Entity Relationship Diagram	Biểu đồ mô tả các thực thể và mối quan hệ trong cơ sở dữ liệu
UC	Use Case	Trường hợp sử dụng mô tả chức năng của hệ thống
FR	Functional Requirement	Yêu cầu chức năng của hệ thống
NFR	Non-Functional Requirement	Yêu cầu phi chức năng của hệ thống
Admin	Administrator	Người quản trị hệ thống
ASP.NET Core	Active Server Pages .NET Core	Framework dùng để xây dựng Backend và Web API

MỞ ĐẦU

1. Phân tích hiện trạng

Công tác xây dựng thời khóa biểu và phân bổ phòng học là một trong những nghiệp vụ quan trọng trong hoạt động quản lý đào tạo tại các trường đại học. Một thời khóa biểu hợp lý không chỉ bảo đảm tiến độ giảng dạy và học tập mà còn góp phần sử dụng hiệu quả đội ngũ giảng viên, phòng học, phòng thí nghiệm, phòng thực hành và các tài nguyên cơ sở vật chất khác của nhà trường.

Tại Trường Đại học Bách khoa – Đại học Đà Nẵng, mỗi học kỳ có số lượng lớn lớp học phần được mở cho nhiều khoa, nhiều chương trình đào tạo và nhiều nhóm sinh viên khác nhau. Quá trình xây dựng thời khóa biểu phải đồng thời xét đến nhiều yếu tố như giảng viên phụ trách, số lượng sinh viên đăng ký, sức chứa phòng học, loại phòng học, khu vực phòng học, số tiết học, số buổi học trong tuần, lịch bận của giảng viên và các ràng buộc đặc thù của từng học phần. Vì vậy, nếu công tác lập lịch được thực hiện thủ công hoặc bán tự động, người quản lý dễ gặp khó khăn trong việc kiểm tra xung đột và tối ưu phương án phân bổ.

Một số vấn đề thường gặp trong thực tế bao gồm: giảng viên có thể bị trùng lịch giảng dạy, phòng học có thể bị sử dụng trùng thời điểm, lớp học phần có sĩ số lớn nhưng được bố trí vào phòng không đủ sức chứa, hoặc phòng học có sức chứa quá lớn nhưng chỉ được sử dụng cho lớp có số lượng sinh viên ít, gây lãng phí tài nguyên. Ngoài ra, việc phân bổ các lớp học trong các khu giảng đường khác nhau cũng ảnh hưởng đến quãng đường di chuyển của sinh viên và giảng viên, đặc biệt khi trong cùng một buổi học có nhiều lớp liên tiếp.

Bên cạnh đó, dữ liệu phục vụ công tác lập thời khóa biểu thường được quản lý từ nhiều nguồn khác nhau như danh sách giảng viên, danh sách lớp học phần, thông tin phòng học, kế hoạch đào tạo, dữ liệu đăng ký học phần và các quy định tổ chức giảng dạy. Việc tổng hợp, kiểm tra và cập nhật dữ liệu từ nhiều nguồn làm tăng khối lượng xử lý thủ công, đồng thời gây khó khăn khi có thay đổi đột xuất về lớp học phần, phòng học hoặc giảng viên.

Xuất phát từ thực tế trên, việc xây dựng một website quản lý và sắp xếp thời khóa biểu, phòng học là cần thiết. Hệ thống cần hỗ trợ quản lý tập trung dữ liệu đào tạo, tự động kiểm tra các ràng buộc cơ bản và ứng dụng thuật toán tối ưu nhằm tạo ra phương án thời khóa biểu hợp lệ, giảm xung đột, nâng cao hiệu quả sử dụng phòng học và hỗ trợ người quản lý trong quá trình ra quyết định.

2. Mục tiêu của đề tài

Đề tài hướng đến việc xây dựng hệ thống DUT Smart Schedule dưới dạng ứng dụng web nhằm hỗ trợ công tác quản lý dữ liệu đào tạo, lập thời khóa biểu và phân bổ phòng học tại Trường Đại học Bách khoa – Đại học Đà Nẵng.

Các mục tiêu chính của đề tài bao gồm:

- Xây dựng hệ thống quản lý tập trung các dữ liệu phục vụ công tác lập thời khóa biểu, bao gồm giảng viên, khoa, phòng học, lớp học phần, học kỳ, tài khoản người dùng và các tham số cấu hình hệ thống.
- Hỗ trợ nhập, cập nhật, tìm kiếm và quản lý dữ liệu đào tạo thông qua giao diện web, giảm phụ thuộc vào việc xử lý thủ công trên các tệp dữ liệu rời rạc.
- Xây dựng cơ chế đăng nhập, xác thực và phân quyền người dùng theo vai trò, bảo đảm mỗi nhóm người dùng chỉ được truy cập các chức năng phù hợp với nhiệm vụ của mình.
- Nghiên cứu và áp dụng phương pháp tối ưu hóa ràng buộc để hỗ trợ tự động sắp xếp thời khóa biểu và phân bổ phòng học cho các lớp học phần.
- Mô hình hóa bài toán lập thời khóa biểu dưới dạng tập các lớp học phần, giảng viên, phòng học, khung thời gian và các ràng buộc liên quan.
- Phân loại và xử lý các ràng buộc cứng nhằm bảo đảm tính hợp lệ của thời khóa biểu, chẳng hạn như không trùng lịch giảng viên, không trùng phòng học, phòng học đủ sức chứa và lớp học phần được bố trí đúng loại phòng.
- Xây dựng các ràng buộc mềm và hàm mục tiêu nhằm nâng cao chất lượng thời khóa biểu, chẳng hạn như giảm số lớp học vào thứ bảy, giảm khoảng trống trong lịch giảng dạy của giảng viên, giảm số ngày giảng viên phải đến trường và ưu tiên sử dụng phòng học có sức chứa phù hợp.
- Hỗ trợ cán bộ quản lý theo dõi kết quả xếp lịch, thống kê dữ liệu đào tạo và điều chỉnh cấu hình thuật toán theo từng học kỳ.
- Hỗ trợ giảng viên tra cứu lịch giảng dạy một cách nhanh chóng, trực quan và thuận tiện.
- Đánh giá kết quả triển khai hệ thống trên dữ liệu thực tế hoặc dữ liệu mô phỏng gần với thực tế của nhà trường, từ đó nhận xét khả năng ứng dụng và đề xuất hướng phát triển tiếp theo.

3. Đối tượng và phạm vi của đề tài

Đối tượng nghiên cứu của đề tài là bài toán quản lý dữ liệu đào tạo và sắp xếp thời khóa biểu, phòng học trong môi trường đại học. Các thành phần chính của bài toán bao gồm lớp học phần, giảng viên, phòng học, học kỳ, thời gian giảng dạy, sĩ số sinh viên, loại phòng học và các ràng buộc liên quan đến quá trình tổ chức giảng dạy.

Phạm vi của đề tài tập trung vào việc xây dựng hệ thống website phục vụ công tác quản lý và sắp xếp thời khóa biểu tại Trường Đại học Bách khoa – Đại học Đà Nẵng. Hệ thống hỗ trợ các nghiệp vụ chính như quản lý tài khoản, quản lý giảng viên, quản lý phòng học, quản lý lớp học phần, cấu hình tham số xếp lịch, tự động sắp xếp thời khóa biểu, thống kê dữ liệu và tra cứu lịch giảng dạy.

Về mặt thuật toán, đề tài tập trung mô hình hóa và giải quyết bài toán xếp thời khóa biểu theo hướng tối ưu hóa ràng buộc. Hệ thống xem xét các ràng buộc cơ bản như không trùng lịch giảng viên, không trùng phòng học, phòng học đáp ứng sức chứa, lớp học phần được bố trí đúng số buổi học và đúng loại phòng. Bên cạnh đó, hệ thống cũng xét đến một số tiêu chí tối ưu như hạn chế học vào thứ bảy, giảm khoảng trống trong lịch giảng dạy của giảng viên, giảm số ngày giảng viên phải đến trường và nâng cao hiệu quả sử dụng phòng học.

Đề tài chưa đi sâu vào các nghiệp vụ ngoài phạm vi thời khóa biểu như quản lý học phí, quản lý điểm, xét tốt nghiệp hoặc quản lý toàn bộ chương trình đào tạo. Ngoài ra, hệ thống chưa mở rộng sang bài toán xếp lịch thi, xếp lịch bảo vệ đồ án, tự động xử lý toàn bộ thay đổi phát sinh sau khi sinh viên đăng ký học phần hoặc tích hợp hoàn chỉnh với tất cả các hệ thống đào tạo hiện có của nhà trường. Đây là các hướng có thể tiếp tục phát triển trong tương lai.

4. Công nghệ sử dụng

Các công nghệ chính được sử dụng trong quá trình xây dựng hệ thống gồm:

- ASP.NET Core Web API để xây dựng các dịch vụ Backend.
- ReactJS để phát triển giao diện người dùng.
- Microsoft SQL Server để lưu trữ và quản lý dữ liệu.
- Google OR-Tools với bộ giải CP-SAT để xây dựng thuật toán sắp xếp thời khóa biểu.
- JWT và Microsoft Authentication để xác thực và phân quyền người dùng.
- Git và GitHub phục vụ quá trình triển khai và quản lý mã nguồn.

5. Cấu trúc của đồ án tốt nghiệp

Nội dung của đồ án được tổ chức thành ba chương chính.

Chương 1. Cơ sở lý thuyết trình bày các kiến thức nền tảng liên quan đến bài toán lập thời khóa biểu trong môi trường đại học, các khái niệm về lớp học phần, giảng viên, phòng học và thời khóa biểu; giới thiệu các loại ràng buộc trong bài toán xếp lịch, tổng quan về đề tài cấp Bộ CB2005 và định hướng kế thừa trong đề tài. Bên cạnh đó, chương này cũng trình bày phương pháp tối ưu hóa được sử dụng, bao gồm Google OR-Tools, bộ giải CP-SAT và các công nghệ được lựa chọn để xây dựng hệ thống.

Chương 2. Phân tích và thiết kế hệ thống trình bày kết quả khảo sát thực tế tại Trường Đại học Bách khoa – Đại học Đà Nẵng; phân tích nghiệp vụ của các nhóm người dùng; xác định yêu cầu chức năng và phi chức năng của hệ thống; xây dựng các biểu đồ Use Case, biểu đồ hoạt động và đặc tả các chức năng chính. Ngoài ra, chương này còn trình bày quá trình thiết kế cơ sở dữ liệu và mô hình dữ liệu phục vụ công tác quản lý và sắp xếp thời khóa biểu.

Chương 3. Triển khai và đánh giá kết quả trình bày quá trình xây dựng hệ thống DUT Smart Schedule, kiến trúc phần mềm, cơ sở dữ liệu và các chức năng đã triển khai. Chương này cũng trình bày mô hình bài toán xếp thời khóa biểu, các ràng buộc được áp dụng, phương pháp tối ưu hóa sử dụng bộ giải CP-SAT, kết quả thực nghiệm trên dữ liệu thực tế của nhà trường và đánh giá hiệu quả hoạt động của hệ thống.

Thông qua việc nghiên cứu và xây dựng hệ thống, đồ án hướng đến việc hỗ trợ công tác quản lý dữ liệu đào tạo và xây dựng thời khóa biểu giảng dạy tại Trường Đại học Bách khoa – Đại học Đà Nẵng, góp phần giảm khối lượng công việc thủ công, nâng cao hiệu quả sử dụng cơ sở vật chất và tạo nền tảng cho việc tiếp tục nghiên cứu, phát triển các giải pháp tối ưu hóa trong quản lý đào tạo ở những giai đoạn tiếp theo.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

Trong chương mở đầu, đồ án đã trình bày hiện trạng công tác quản lý và xây dựng thời khóa biểu tại Trường Đại học Bách khoa – Đại học Đà Nẵng, đồng thời làm rõ mục tiêu và định hướng nghiên cứu của đề tài. Để xây dựng được một hệ thống hỗ trợ sắp xếp thời khóa biểu hiệu quả, trước hết cần nghiên cứu các cơ sở lý thuyết liên quan đến bài toán lập thời khóa biểu trong môi trường đại học, các phương pháp tối ưu hóa hiện đại và Nội dung chương này trình bày các khái niệm cơ bản về thời khóa biểu, lớp học phần, giảng viên và phòng học; phân tích đặc điểm của bài toán lập thời khóa biểu trong môi trường đại học; giới thiệu đề tài cấp Bộ CB2005 và định hướng kế thừa của đề tài; đồng thời trình bày các phương pháp tối ưu hóa và các công nghệ được sử dụng trong quá trình phát triển website DUT Smart Schedule.

1.1 Một số khái niệm chính

1.1.1 Thời khoá biểu đại học

Thời khóa biểu là kế hoạch tổ chức giảng dạy và học tập của nhà trường trong một khoảng thời gian nhất định, thường là một học kỳ hoặc năm học. Thời khóa biểu xác định thời gian học, phòng học và giảng viên phụ trách của từng lớp học phần.

Trong môi trường đại học, thời khóa biểu đóng vai trò quan trọng trong việc điều phối hoạt động giảng dạy, học tập và sử dụng cơ sở vật chất. Một thời khóa biểu hợp lý giúp bảo đảm tiến độ đào tạo, nâng cao hiệu quả sử dụng phòng học và giảm thiểu các xung đột về tài nguyên.

1.1.2 Lớp học phần

Lớp học phần là đơn vị tổ chức giảng dạy cụ thể của một học phần trong một học kỳ. Mỗi lớp học phần có các thông tin như:

- Mã lớp học phần.
- Tên học phần.
- Giảng viên phụ trách.
- Số lượng sinh viên đăng ký.
- Số tín chỉ.
- Học kỳ và năm học.
- Số tuần giảng dạy.

Trong bài toán lập thời khóa biểu, lớp học phần là đối tượng chính cần được gán vào các khung thời gian và phòng học phù hợp.

1.1.3 Giảng viên

Giảng viên là người trực tiếp thực hiện hoạt động giảng dạy đối với các lớp học phần được phân công.

Mỗi giảng viên có những đặc điểm riêng như:

- Chuyên môn giảng dạy.
- Danh sách học phần được phân công.
- Các khoảng thời gian không thể tham gia giảng dạy.
- Khối lượng giảng dạy tối đa trong học kỳ.

Trong quá trình xây dựng thời khóa biểu, hệ thống phải bảo đảm rằng một giảng viên không bị phân công giảng dạy nhiều lớp học phần tại cùng một thời điểm.

1.1.4 Phòng học

Phòng học là địa điểm diễn ra hoạt động giảng dạy của các lớp học phần.

Mỗi phòng học được mô tả bởi các thuộc tính:

- Mã phòng.
- Khu nhà.
- Sức chứa.
- Loại phòng.
- Trạng thái sử dụng.

Ngoài các phòng học lý thuyết thông thường, nhà trường còn có các phòng thực hành, phòng thí nghiệm và các phòng chuyên dụng phục vụ cho những học phần đặc thù.

1.1.5 Bài toán lập thời khóa biểu

Bài toán lập thời khóa biểu là bài toán tìm kiếm phương án phân công thời gian học và phòng học cho tất cả các lớp học phần sao cho thỏa mãn các yêu cầu và ràng buộc của nhà trường.

Đây là một bài toán tối ưu tổ hợp có độ phức tạp cao do số lượng phương án có thể sinh ra rất lớn và phải đồng thời thỏa mãn nhiều điều kiện khác nhau.

Từ góc độ toán học, bài toán lập thời khóa biểu có thể được xem như quá trình ánh xạ tập lớp học phần vào tập các khung thời gian và phòng học khả dụng sao cho tối đa hóa chất lượng thời khóa biểu đồng thời thỏa mãn các ràng buộc bắt buộc.

1.2 Bài toán lập thời khóa biểu trong môi trường đại học

1.2.1 Đặc điểm bài toán

Bài toán lập thời khóa biểu đại học là một trong những bài toán tối ưu hóa nổi tiếng trong lĩnh vực nghiên cứu tác nghiệp và trí tuệ nhân tạo.

Đặc điểm nổi bật của bài toán này bao gồm:

- Số lượng lớp học phần lớn.
- Nhiều giảng viên tham gia giảng dạy.
- Nhiều loại phòng học khác nhau.
- Nhiều ràng buộc nghiệp vụ.
- Dữ liệu thường xuyên thay đổi.

Tại Trường Đại học Bách khoa – Đại học Đà Nẵng, mỗi học kỳ có hàng nghìn lớp học phần cần được sắp xếp, làm cho không gian tìm kiếm của bài toán trở nên rất lớn.

1.2.2 Các ràng buộc trong bài toán

Các ràng buộc trong bài toán lập thời khóa biểu thường được chia thành hai nhóm chính: ràng buộc cứng và ràng buộc mềm.

a. Ràng buộc cứng

Ràng buộc cứng là các điều kiện bắt buộc phải được thỏa mãn. Nếu vi phạm một ràng buộc cứng thì thời khóa biểu được xem là không hợp lệ.

Các ràng buộc cứng tiêu biểu của hệ thống bao gồm:

- Một phòng học chỉ được sử dụng cho một lớp học phần tại một thời điểm.
- Một giảng viên chỉ được giảng dạy một lớp học phần tại một thời điểm.
- Sức chứa phòng học phải lớn hơn hoặc bằng số lượng sinh viên của lớp học phần.
- Lớp học phần phải được bố trí đủ số buổi học theo kế hoạch đào tạo.
- Các khoảng thời gian khóa lịch không được sử dụng để xếp lịch.
- Các lớp thực hành phải được bố trí tại các phòng thực hành tương ứng.
- Các lớp thí nghiệm phải được bố trí tại các phòng thí nghiệm tương ứng.

b. Ràng buộc mềm

Ràng buộc mềm là các tiêu chí tối ưu nhằm nâng cao chất lượng thời khóa biểu. Việc vi phạm ràng buộc mềm không làm cho thời khóa biểu mất tính hợp lệ nhưng sẽ làm giảm chất lượng của phương án xếp lịch.

Các ràng buộc mềm được sử dụng trong đề tài bao gồm:

- Giảm số lớp học vào thứ bảy.
- Giảm khoảng trống trong lịch giảng dạy của giảng viên.

- Giảm số ngày phải đến trường của giảng viên.
- Ưu tiên sử dụng phòng học có sức chứa phù hợp.
- Ưu tiên phân khu phòng học cho các chương trình đặc thù.
- Hạn chế việc thay đổi khu vực học tập của cùng một lớp học phần.

Bảng 1.1. Phân loại ràng buộc trong bài toán lập thời khoá biểu

Loại ràng buộc	Mục tiêu
Ràng buộc cứng	Bảo đảm tính hợp lệ của thời khóa biểu
Ràng buộc mềm	Nâng cao chất lượng và tính tối ưu của thời khóa biểu

1.3 Phương pháp tối ưu hóa được sử dụng trong đề tài

1.3.1 Constraint Programming

Constraint Programming (CP) là một phương pháp giải quyết bài toán dựa trên việc mô hình hóa các yêu cầu của bài toán dưới dạng các ràng buộc. Thay vì liệt kê toàn bộ các phương án có thể xảy ra, phương pháp này tập trung xác định các biến quyết định, miền giá trị của các biến và tập các ràng buộc cần được thỏa mãn.

Đối với bài toán lập thời khóa biểu, mỗi lớp học phần cần được gán vào một khung thời gian và một phòng học phù hợp. Quá trình gán này phải đồng thời thỏa mãn nhiều điều kiện liên quan đến giảng viên, phòng học, thời gian học và các yêu cầu đặc thù của nhà trường. Vì vậy, Constraint Programming là một hướng tiếp cận phù hợp để mô hình hóa bài toán và tìm kiếm lời giải.

Ưu điểm của phương pháp này là khả năng biểu diễn trực tiếp các ràng buộc thực tế của bài toán, giúp việc xây dựng mô hình trở nên linh hoạt và dễ mở rộng khi xuất hiện các yêu cầu mới.

1.3.2 Google OR-Tools

Google OR-Tools là thư viện mã nguồn mở được phát triển bởi Google nhằm hỗ trợ giải quyết các bài toán tối ưu hóa và nghiên cứu tác nghiệp. Thư viện cung cấp nhiều bộ giải khác nhau cho các bài toán tối ưu tuyến tính, tối ưu nguyên, bài toán định tuyến và bài toán lập lịch.

Trong đề tài này, Google OR-Tools được sử dụng để xây dựng mô hình sắp xếp thời khóa biểu dựa trên các ràng buộc của hệ thống. Thư viện hỗ trợ nhiều ngôn ngữ lập trình khác nhau như C++, Java, Python và C#. Nhờ khả năng tích hợp tốt với nền tảng .NET, OR-Tools được sử dụng trực tiếp trong Backend của hệ thống.

Một số ưu điểm nổi bật của Google OR-Tools gồm:

- Hiệu năng xử lý cao.
- Hỗ trợ nhiều loại bài toán tối ưu khác nhau.

- Có khả năng xử lý các bài toán với số lượng lớn biến quyết định.
- Cho phép xây dựng các mô hình ràng buộc phức tạp.
- Được cộng đồng sử dụng rộng rãi và liên tục cập nhật.

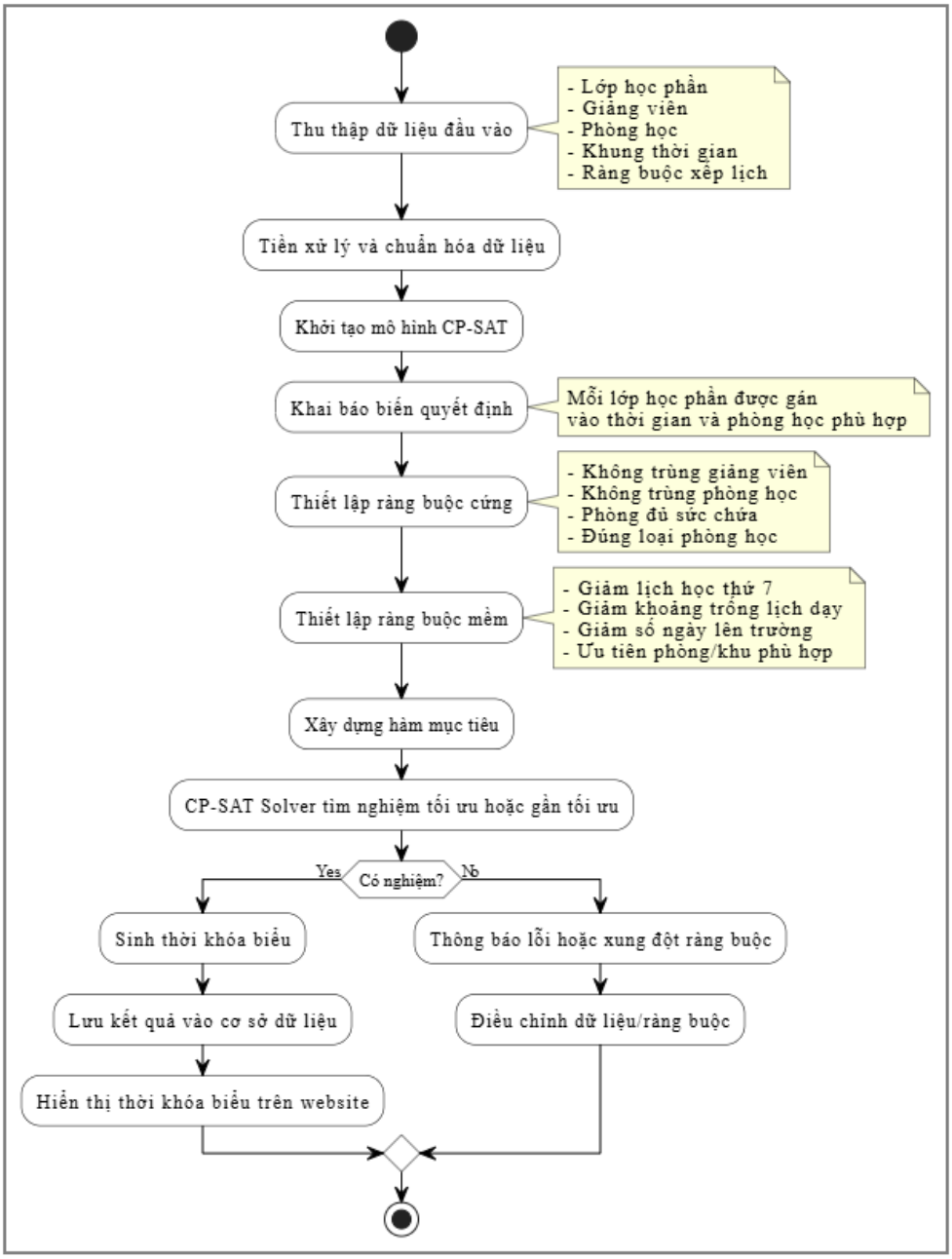
1.3.3 Bộ giải CP-SAT

CP-SAT (Constraint Programming Satisfiability Solver) là bộ giải tối ưu hóa ràng buộc hiện đại được tích hợp trong Google OR-Tools. Đây là sự kết hợp giữa kỹ thuật Constraint Programming và Boolean Satisfiability (SAT).

Bộ giải CP-SAT hoạt động bằng cách chuyển đổi bài toán thành tập các biến và ràng buộc logic, sau đó tìm kiếm phương án tối ưu thông qua các kỹ thuật suy diễn, cắt tỉa không gian tìm kiếm và tối ưu hóa tổ hợp.

Trong hệ thống DUT Smart Schedule, CP-SAT được sử dụng để:

- Phân công thời gian học cho các lớp học phần.
- Phân bổ phòng học phù hợp.
- Kiểm tra các ràng buộc cứng.
- Tối ưu các tiêu chí chất lượng thời khóa biểu.



Hình 1.1. Mô hình giải bài toán lập thời khóa biểu bằng bộ giải CP-SAT

Quá trình xử lý tổng quát gồm các bước:

1. Thu thập dữ liệu đầu vào.
2. Xây dựng tập biến quyết định.
3. Thiết lập các ràng buộc cứng.
4. Thiết lập các ràng buộc mềm.
5. Xây dựng hàm mục tiêu.
6. Tìm kiếm phương án tối ưu.
7. Xuất kết quả thời khóa biểu

1.3.4 Hàm mục tiêu tối ưu

Ngoài việc thỏa mãn các ràng buộc cứng, hệ thống còn hướng tới việc tối ưu chất lượng thời khóa biểu thông qua hàm mục tiêu.

Các tiêu chí tối ưu chính của hệ thống bao gồm:

- Giảm số lớp học vào ngày thứ bảy.
- Giảm khoảng trống trong lịch giảng dạy của giảng viên.
- Giảm số ngày phải đến trường của giảng viên.
- Giảm mức độ lãng phí sức chứa phòng học.
- Ưu tiên phân bổ phòng học đúng khu vực chuyên dụng.
- Hạn chế thay đổi khu vực học tập của cùng một lớp học phần.

1.4 Các công nghệ sử dụng

1.4.1 ASP.NET Core

ASP.NET Core là framework mã nguồn mở do Microsoft phát triển nhằm xây dựng các ứng dụng web và dịch vụ API hiện đại.

Trong hệ thống DUT Smart Schedule, ASP.NET Core được sử dụng để phát triển Backend, cung cấp các RESTful API phục vụ quản lý dữ liệu và thực hiện thuật toán sắp xếp thời khóa biểu.

Một số ưu điểm của ASP.NET Core:

- Hiệu năng cao.
- Hỗ trợ Dependency Injection.
- Dễ mở rộng và bảo trì.
- Tích hợp tốt với Entity Framework Core.

1.4.2 ReactJS

ReactJS là thư viện JavaScript được sử dụng để xây dựng giao diện người dùng. Hệ thống sử dụng ReactJS để phát triển các chức năng:

- Dashboard thống kê.
- Quản lý giảng viên.
- Quản lý phòng học.
- Quản lý lớp học phần.
- Cấu hình hệ thống.
- Xếp thời khóa biểu.
- Tra cứu lịch dạy.

ReactJS giúp giao diện có khả năng phản hồi nhanh và dễ dàng mở rộng trong tương lai.

1.4.3 Microsoft SQL Server

Microsoft SQL Server được sử dụng làm hệ quản trị cơ sở dữ liệu của hệ thống. Cơ sở dữ liệu lưu trữ toàn bộ thông tin về:

- Giảng viên.
- Phòng học.
- Lớp học phần.
- Tài khoản.
- Vai trò.
- Thời khóa biểu.
- Các tham số cấu hình.

SQL Server bảo đảm tính toàn vẹn dữ liệu và khả năng xử lý hiệu quả đối với lượng lớn dữ liệu đào tạo.

1.4.4 JWT Authentication

JWT (JSON Web Token) là cơ chế xác thực được sử dụng trong hệ thống nhằm quản lý phiên đăng nhập của người dùng.

Sau khi xác thực thành công, hệ thống cấp một JWT chứa các thông tin cần thiết như:

- AccountID.
- Username.
- Role.
- LecturerID (nếu có).

JWT giúp hệ thống giảm tải cho máy chủ và hỗ trợ xác thực hiệu quả trong các ứng dụng web hiện đại.

1.4.5 Microsoft Authentication

Bên cạnh hình thức đăng nhập bằng tài khoản nội bộ, hệ thống còn hỗ trợ đăng nhập thông qua tài khoản Microsoft của nhà trường.

Giải pháp này giúp:

- Tận dụng hạ tầng xác thực sẵn có.
- Giảm việc quản lý mật khẩu riêng.
- Tăng mức độ bảo mật.
- Nâng cao trải nghiệm người dùng.

Bảng 1.2. Các công nghệ sử dụng trong hệ thống

Công nghệ	Vai trò
ASP.NET Core	Phát triển Backend
ReactJS	Phát triển Frontend
SQL Server	Quản lý cơ sở dữ liệu
Entity Framework Core	ORM
Google OR-Tools	Tối ưu hóa thời khóa biểu
CP-SAT	Bộ giải tối ưu
JWT	Xác thực người dùng
Microsoft Authentication	Đăng nhập SSO

Chương này đã trình bày các cơ sở lý thuyết liên quan đến bài toán lập thời khóa biểu trong môi trường đại học, bao gồm các khái niệm nền tảng, đặc điểm bài toán, các loại ràng buộc, phương pháp tối ưu hóa bằng Constraint Programming và bộ giải CP-SAT. Bên cạnh đó, chương cũng giới thiệu các công nghệ được sử dụng để xây dựng hệ thống DUT Smart Schedule. Những nội dung này là cơ sở để thực hiện việc phân tích yêu cầu, thiết kế hệ thống và triển khai các chức năng trong các chương tiếp theo.

1.5 Tổng quan các nghiên cứu và hướng tiếp cận liên quan

Bài toán lập thời khóa biểu và phân bổ phòng học là một bài toán tối ưu tổ hợp có tính ứng dụng cao trong lĩnh vực quản lý đào tạo. Trong môi trường đại học, bài toán này thường có quy mô lớn do phải xử lý đồng thời nhiều lớp học phần, nhiều giảng viên, nhiều phòng học và nhiều khung thời gian khác nhau. Mục tiêu của bài toán không chỉ là tạo ra một thời khóa biểu hợp lệ mà còn phải nâng cao chất lượng lịch học, giảm xung đột, sử dụng hiệu quả cơ sở vật chất và tạo sự thuận tiện cho giảng viên, sinh viên.

Các hướng tiếp cận phổ biến đối với bài toán lập thời khóa biểu bao gồm thuật toán tô màu đồ thị, thuật toán di truyền, tối ưu bầy đàn, lập trình số nguyên, lập trình số nguyên hỗn hợp, lập trình ràng buộc và các phương pháp lai. Mỗi phương pháp có ưu điểm và hạn chế riêng. Các thuật toán metaheuristic như thuật toán di truyền hoặc tối ưu bầy đàn có khả năng tìm lời giải gần tối ưu trong không gian tìm kiếm lớn, tuy nhiên kết quả phụ thuộc nhiều vào cách thiết kế hàm thích nghi và tham số thuật toán. Trong khi đó, các phương pháp dựa trên mô hình toán học như lập trình số nguyên hỗn hợp hoặc lập trình ràng buộc cho phép mô tả bài toán một cách chặt chẽ, dễ kiểm soát ràng buộc và phù hợp với các bài toán cần bảo đảm tính hợp lệ của lời giải.

Trong đề tài "Tối ưu hóa việc bố trí phòng học và xây dựng thời khóa biểu cho Trường Đại học Bách khoa", nhóm tác giả đã tiếp cận bài toán bằng phương pháp lập

trình số nguyên hỗn hợp. Bài toán được mô hình hóa thông qua các tập hợp học phần, phòng học, buổi học, tiết học, nhóm sinh viên và giảng viên. Các ma trận quan hệ được sử dụng để biểu diễn mối liên hệ giữa học phần với nhóm sinh viên, học phần với giảng viên, học phần với thời gian học và khoảng cách tương đối giữa các phòng học. Hướng tiếp cận này cho thấy việc mô hình hóa toán học có thể hỗ trợ giải quyết bài toán lập thời khóa biểu và phân bổ phòng học theo nhiều mục tiêu khác nhau.

Kế thừa ý tưởng từ các nghiên cứu trước, đề tài này lựa chọn hướng tiếp cận lập trình ràng buộc với bộ giải CP-SAT của Google OR-Tools. Thay vì chỉ xem bài toán như một bài toán phân bổ thông thường, hệ thống DUT Smart Schedule mô hình hóa quá trình lập thời khóa biểu thành tập các biến quyết định và ràng buộc. Mỗi lớp học phần cần được gán vào một phòng học và một khung thời gian phù hợp. Bộ giải có nhiệm vụ tìm phương án thỏa mãn toàn bộ ràng buộc cứng và tối ưu hóa các ràng buộc mềm thông qua hàm mục tiêu.

1.6 Mô hình tổng quát của bài toán lập thời khóa biểu

Bài toán lập thời khóa biểu trong trường đại học có thể được mô tả như sau: Cho trước danh sách lớp học phần cần mở trong một học kỳ, danh sách giảng viên phụ trách, danh sách phòng học, danh sách khung thời gian có thể sử dụng và các ràng buộc nghiệp vụ liên quan. Cần xác định thời gian học và phòng học cho từng lớp học phần sao cho thời khóa biểu cuối cùng không phát sinh xung đột và đạt chất lượng tốt nhất theo các tiêu chí tối ưu đã định nghĩa.

Các thành phần chính của bài toán bao gồm:

Tập lớp học phần: là tập hợp các lớp cần được xếp lịch trong học kỳ. Mỗi lớp học phần có các thuộc tính như mã lớp học phần, tên học phần, số tín chỉ, số tiết học, số buổi học, giảng viên phụ trách, sĩ số dự kiến, loại lớp và chương trình đào tạo liên quan.

Tập giảng viên: là tập hợp các giảng viên tham gia giảng dạy trong học kỳ. Mỗi giảng viên có thể phụ trách một hoặc nhiều lớp học phần. Khi xếp lịch, hệ thống cần bảo đảm một giảng viên không được dạy nhiều lớp trong cùng một khung thời gian.

Tập phòng học: là tập hợp các phòng có thể sử dụng để tổ chức giảng dạy. Mỗi phòng học có các thông tin như mã phòng, tên phòng, khu giảng đường, tầng, sức chứa, loại phòng và trạng thái sử dụng.

Tập khung thời gian: là tập hợp các buổi học hoặc tiết học có thể sử dụng trong tuần. Tùy theo quy định đào tạo, mỗi ngày có thể được chia thành buổi sáng, buổi chiều và các tiết học cụ thể. Hệ thống cần gán lớp học phần vào các khung thời gian phù hợp với thời lượng học phần.

Tập ràng buộc: là tập các điều kiện cần được xét trong quá trình xếp lịch. Các ràng buộc được chia thành ràng buộc cứng và ràng buộc mềm. Ràng buộc cứng quyết định tính hợp lệ của thời khóa biểu, trong khi ràng buộc mềm ảnh hưởng đến chất lượng và mức độ tối ưu của lời giải.

1.7 Các tập hợp và thông số sử dụng trong mô hình

Để mô hình hóa bài toán, có thể sử dụng các tập hợp và thông số sau:

Bảng 1.3. Các tập hợp và thông số sử dụng trong mô hình

Ký hiệu	Ý nghĩa
C	Tập các lớp học phần cần xếp lịch
R	Tập các phòng học có thể sử dụng
T	Tập các khung thời gian trong tuần
L	Tập các giảng viên
D	Tập các ngày học trong tuần
B	Tập các buổi học trong ngày
c	Một lớp học phần thuộc tập C
r	Một phòng học thuộc tập R
t	Một khung thời gian thuộc tập T
l	Một giảng viên thuộc tập L
capacity(r)	Sức chứa của phòng học r
size(c)	Số lượng sinh viên của lớp học phần c
lecturer(c)	Giảng viên phụ trách lớp học phần c
type(c)	Loại lớp học phần, ví dụ lý thuyết, thực hành, thí nghiệm
type(r)	Loại phòng học, ví dụ phòng lý thuyết, phòng thực hành, phòng thí nghiệm
day(t)	Ngày tương ứng với khung thời gian t
session(t)	Buổi học tương ứng với khung thời gian t

Ngoài các tập hợp và thông số trên, hệ thống có thể sử dụng thêm các tham số cấu hình như trọng số phạt khi học vào thứ bảy, trọng số phạt khi giảng viên phải dạy nhiều ngày trong tuần, trọng số phạt khi phòng học có sức chứa chênh lệch quá lớn so với sĩ số lớp học phần và trọng số ưu tiên cho một số khu giảng đường đặc thù.

1.8 Biến quyết định

Biến quyết định chính của bài toán là biến nhị phân biểu diễn việc một lớp học phần có được gán vào một phòng học và một khung thời gian hay không.

Gọi x[c, r, t] là biến quyết định:

x[c, r, t] = 1 nếu lớp học phần c được xếp tại phòng học r vào khung thời gian t.

x[c, r, t] = 0 nếu ngược lại.

Dựa trên biến quyết định này, hệ thống có thể xây dựng các ràng buộc để kiểm tra một lớp học phần được xếp đủ lịch, một phòng học không bị trùng, một giảng viên không bị trùng lịch và phòng học đáp ứng yêu cầu của lớp học phần.

Ngoài biến quyết định chính, hệ thống có thể sử dụng các biến phụ để phục vụ tối ưu hóa ràng buộc mềm, ví dụ:

- Biến xác định một lớp học phần có được xếp vào ngày thứ bảy hay không.
- Biến xác định một giảng viên có phải đến trường trong một ngày cụ thể hay không.
- Biến xác định khoảng trống giữa các buổi dạy của giảng viên.
- Biến xác định mức chênh lệch giữa sức chứa phòng học và sĩ số lớp học phần.
- Biến xác định một lớp học phần có được xếp đúng khu vực ưu tiên hay không.

1.9 Ràng buộc cứng trong bài toán lập thời khóa biểu

Ràng buộc cứng là các điều kiện bắt buộc phải được thỏa mãn. Nếu một phương án vi phạm ràng buộc cứng thì phương án đó không được xem là thời khóa biểu hợp lệ.

Mỗi lớp học phần phải được xếp đủ lịch

Mỗi lớp học phần cần được xếp vào đúng số buổi hoặc số tiết theo kế hoạch đào tạo. Hệ thống không được bỏ sót lớp học phần, đồng thời không được xếp thừa số buổi học so với yêu cầu.

Ý nghĩa nghiệp vụ của ràng buộc này là bảo đảm tất cả lớp học phần được mở trong học kỳ đều có lịch học đầy đủ, phù hợp với thời lượng của học phần.

Giảng viên không được trùng lịch

Một giảng viên không thể giảng dạy hai lớp học phần khác nhau trong cùng một khung thời gian. Vì vậy, với mỗi giảng viên và mỗi khung thời gian, tổng số lớp học phần được phân công cho giảng viên đó không được vượt quá một.

Ràng buộc này giúp loại bỏ các trường hợp giảng viên bị xếp trùng lịch, bảo đảm tính khả thi trong quá trình tổ chức giảng dạy.

Phòng học không được trùng lịch

Một phòng học tại một khung thời gian chỉ có thể phục vụ tối đa một lớp học phần. Hệ thống không được xếp hai lớp học phần vào cùng một phòng trong cùng một thời điểm.

Ràng buộc này bảo đảm việc sử dụng cơ sở vật chất không bị xung đột và thời khóa biểu có thể triển khai trong thực tế.

Phòng học phải đáp ứng sức chứa

Sức chứa của phòng học phải lớn hơn hoặc bằng số lượng sinh viên của lớp học phần. Nếu lớp học phần có sĩ số lớn hơn sức chứa phòng học, phương án xếp lịch đó không hợp lệ.

Ràng buộc này giúp bảo đảm điều kiện học tập cho sinh viên và tránh tình trạng lớp học bị bố trí vào phòng không đủ chỗ.

Phòng học phải phù hợp loại lớp

Các lớp học lý thuyết, thực hành và thí nghiệm có yêu cầu khác nhau về phòng học. Lớp lý thuyết có thể được xếp vào phòng học thông thường; lớp thực hành cần được xếp vào phòng máy hoặc phòng thực hành; lớp thí nghiệm cần được xếp vào phòng thí nghiệm phù hợp.

Ràng buộc này giúp hệ thống bảo đảm lớp học phần được bố trí đúng loại phòng phục vụ yêu cầu giảng dạy.

Không sử dụng các khung thời gian bị khóa

Trong một số trường hợp, nhà trường có thể khóa một số khung thời gian do lịch họp, sự kiện, bảo trì cơ sở vật chất hoặc các lý do quản lý khác. Các khung thời gian bị khóa không được sử dụng để xếp lịch.

Ràng buộc này giúp người quản lý chủ động kiểm soát các khoảng thời gian không thể tổ chức giảng dạy.

1.10 Ràng buộc mềm trong bài toán lập thời khóa biểu

Ràng buộc mềm là các tiêu chí được đưa vào để cải thiện chất lượng thời khóa biểu. Việc vi phạm ràng buộc mềm không làm cho lời giải mất tính hợp lệ, nhưng sẽ làm tăng chi phí phạt trong hàm mục tiêu.

Hạn chế lớp học vào thứ bảy

Các lớp học được xếp vào thứ bảy có thể gây bất tiện cho giảng viên và sinh viên. Do đó, hệ thống nên hạn chế số lớp học vào thứ bảy, trừ trường hợp cần thiết do thiếu phòng học hoặc khung thời gian.

Giảm số ngày phải đến trường của giảng viên

Nếu lịch dạy của một giảng viên được phân tán quá nhiều ngày trong tuần, giảng viên sẽ mất nhiều thời gian di chuyển và khó sắp xếp công việc chuyên môn khác. Hệ thống cần ưu tiên gom lịch dạy của giảng viên vào số ngày hợp lý.

Giảm khoảng trống giữa các buổi dạy của giảng viên

Một thời khóa biểu tốt nên hạn chế tình trạng giảng viên có các buổi dạy cách xa nhau trong cùng một ngày. Khoảng trống quá lớn giữa các lớp học gây lãng phí thời gian chờ và làm giảm chất lượng lịch giảng dạy.

Ưu tiên phòng học có sức chứa phù hợp

Mặc dù phòng học chỉ cần có sức chứa lớn hơn hoặc bằng sĩ số lớp học phần, việc xếp lớp nhỏ vào phòng quá lớn có thể gây lãng phí tài nguyên. Vì vậy, hệ thống cần ưu tiên sử dụng phòng có sức chứa gần với sĩ số lớp học phần.

Ưu tiên khu vực phòng học đặc thù

Một số chương trình đào tạo hoặc nhóm lớp có thể được ưu tiên học tại một khu giảng đường nhất định. Ví dụ, các lớp thuộc chương trình đặc thù có thể được ưu tiên xếp vào khu phòng học riêng để thuận tiện cho công tác quản lý và học tập.

Hạn chế thay đổi khu vực học tập trong cùng một buổi

Nếu sinh viên phải di chuyển giữa các khu giảng đường xa nhau trong cùng một buổi học, việc học tập sẽ bị ảnh hưởng. Do đó, hệ thống có thể bổ sung tiêu chí hạn chế thay đổi khu vực học tập liên tục giữa các lớp thuộc cùng một nhóm sinh viên.

1.11 Hàm mục tiêu

Hàm mục tiêu được xây dựng nhằm tối thiểu hóa tổng chi phí phạt của các ràng buộc mềm. Mỗi loại ràng buộc mềm được gán một trọng số khác nhau tùy theo mức độ quan trọng trong thực tế.

Có thể mô tả hàm mục tiêu tổng quát như sau:

Minimize Z = w1.P1 + w2.P2 + w3.P3 + w4.P4 + w5.P5 + w6.P6

Trong đó:

Thành phần	Ý nghĩa
P1	Chi phí phạt do xếp lớp học vào thứ bảy
P2	Chi phí phạt do giảng viên phải đến trường nhiều ngày
P3	Chi phí phạt do tồn tại khoảng trống trong lịch giảng viên
P4	Chi phí phạt do sử dụng phòng học có sức chứa chưa phù hợp
P5	Chi phí phạt do không xếp đúng khu vực ưu tiên
P6	Chi phí phạt do thay đổi khu vực học tập quá nhiều
w1, w2, w3, w4, w5, w6	Trọng số tương ứng với từng tiêu chí

Việc sử dụng trọng số cho phép hệ thống linh hoạt điều chỉnh chiến lược tối ưu. Nếu nhà trường muốn ưu tiên giảm lịch học thứ bảy, có thể tăng trọng số w1. Nếu muốn ưu tiên sử dụng phòng học hiệu quả, có thể tăng trọng số w4. Cách tiếp cận này giúp hệ thống phù hợp hơn với yêu cầu thực tế của từng học kỳ.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Nội dung chương này trình bày quá trình khảo sát, phân tích nghiệp vụ và thiết kế hệ thống DUT Smart Schedule. Trước hết, chương xác định các nhóm người dùng chính và nghiệp vụ tương ứng trong quá trình quản lý dữ liệu đào tạo và xây dựng thời khóa biểu. Tiếp theo, chương trình bày các yêu cầu chức năng, yêu cầu phi chức năng, nguyên lý hoạt động tổng quát của hệ thống, các biểu đồ use case, quy trình nghiệp vụ và đặc tả các chức năng quan trọng. Cuối cùng, chương trình bày thiết kế cơ sở dữ liệu và các bảng dữ liệu chính phục vụ cho việc quản lý, sắp xếp và tra cứu thời khóa biểu.

2.1 Khảo sát và phân tích nghiệp vụ

Để xây dựng hệ thống phù hợp với thực tế sử dụng tại Trường Đại học Bách khoa – Đại học Đà Nẵng, trước hết cần khảo sát quy trình xây dựng thời khóa biểu hiện tại và xác định rõ các nhóm người dùng tham gia vào hệ thống. Việc phân tích nghiệp vụ giúp làm rõ vai trò, quyền hạn và nhu cầu sử dụng của từng nhóm người dùng, từ đó làm cơ sở cho việc thiết kế chức năng và cơ sở dữ liệu.

Qua quá trình tìm hiểu, hệ thống DUT Smart Schedule có ba nhóm người dùng chính gồm:

- Quản trị viên (Admin).
- Cán bộ quản lý đào tạo.
- Giảng viên.

Mỗi nhóm người dùng có những quyền hạn và trách nhiệm khác nhau trong quá trình vận hành hệ thống.

2.1.1 Khảo sát quy trình nghiệp vụ hiện tại

Quy trình xây dựng thời khóa biểu tại trường đại học thường bắt đầu từ việc tổng hợp kế hoạch đào tạo và danh sách các lớp học phần dự kiến mở trong học kỳ. Cán bộ quản lý cần thu thập các thông tin liên quan như mã lớp học phần, tên học phần, giảng viên phụ trách, số lượng sinh viên dự kiến, loại lớp học, số tín chỉ, thời lượng giảng dạy và các yêu cầu đặc thù của từng lớp học phần.

Sau khi có danh sách lớp học phần, cán bộ quản lý tiếp tục đối chiếu với dữ liệu phòng học hiện có, bao gồm mã phòng, sức chứa, loại phòng, khu giảng đường, tình trạng sử dụng và các khoảng thời gian có thể tổ chức giảng dạy. Trong quá trình xếp lịch, người quản lý cần kiểm tra nhiều ràng buộc như tránh trùng lịch giảng viên, tránh trùng phòng học, bảo đảm phòng học đủ sức chứa, bảo đảm lớp thực hành hoặc thí nghiệm được bố trí đúng phòng, đồng thời cân nhắc mức độ thuận tiện cho giảng viên và sinh viên.

Nếu thực hiện thủ công, quy trình này thường tốn nhiều thời gian do dữ liệu lớn và nhiều ràng buộc cần kiểm tra. Khi có thay đổi về giảng viên, lớp học phần hoặc phòng học, người quản lý phải điều chỉnh lại lịch, kiểm tra lại xung đột và cập nhật nhiều bảng dữ liệu liên quan. Điều này làm tăng nguy cơ sai sót và gây khó khăn trong việc bảo đảm chất lượng thời khóa biểu.

Do đó, hệ thống DUT Smart Schedule được xây dựng nhằm hỗ trợ quản lý dữ liệu tập trung, tự động kiểm tra ràng buộc và hỗ trợ tạo phương án thời khóa biểu dựa trên thuật toán tối ưu.

2.1.2 Nghiệp vụ của Quản trị viên (Admin)

Quản trị viên là người có quyền cao nhất trong hệ thống và chịu trách nhiệm quản lý toàn bộ dữ liệu cũng như vận hành hệ thống.

Các nghiệp vụ chính của Quản trị viên bao gồm:

- Quản lý tài khoản người dùng.
- Quản lý thông tin giảng viên.
- Quản lý phòng học.
- Quản lý lớp học phần.
- Quản lý dữ liệu thời khóa biểu.
- Cấu hình các tham số phục vụ thuật toán xếp lịch.
- Theo dõi tiến trình và kết quả xếp thời khóa biểu.
- Thực hiện khóa lịch hoặc mở khóa lịch khi cần thiết.
- Quản lý các khoảng thời gian khóa lịch của giảng viên và phòng học.
- Theo dõi các thống kê tổng hợp của hệ thống.

Ngoài ra, Quản trị viên có quyền khởi tạo hoặc chạy lại quá trình xếp thời khóa biểu cho từng học kỳ dựa trên các tham số và dữ liệu hiện có.

2.1.3 Nghiệp vụ của Cán bộ quản lý đào tạo

Cán bộ quản lý đào tạo là người trực tiếp tham gia công tác xây dựng kế hoạch giảng dạy và theo dõi thời khóa biểu.

Các nghiệp vụ chính gồm:

- Quản lý dữ liệu lớp học phần.
- Quản lý kế hoạch đào tạo theo từng học kỳ.
- Theo dõi thời khóa biểu đã được xây dựng.
- Kiểm tra tính hợp lệ của dữ liệu đầu vào.
- Theo dõi các lớp học phần chưa được xếp lịch.
- Theo dõi tình trạng sử dụng phòng học.
- Hỗ trợ điều chỉnh dữ liệu khi có thay đổi phát sinh.

2.1.4 Nghiệp vụ của Giảng viên

Giảng viên là đối tượng sử dụng hệ thống nhằm theo dõi lịch giảng dạy được phân công.

Các nghiệp vụ chính của giảng viên bao gồm:

- Tra cứu thời khóa biểu cá nhân.
- Xem lịch giảng dạy theo học kỳ.
- Xem thời khóa biểu theo tuần.
- Theo dõi thông tin lớp học phần được phân công.
- Theo dõi thông tin phòng học và thời gian giảng dạy.

2.2 Yêu cầu hệ thống

Yêu cầu hệ thống được chia thành hai nhóm chính: yêu cầu chức năng và yêu cầu phi chức năng. Yêu cầu chức năng mô tả các chức năng mà hệ thống cần cung cấp cho người dùng. Yêu cầu phi chức năng mô tả các tiêu chí về bảo mật, hiệu năng, khả năng mở rộng, tính dễ sử dụng và tính toàn vẹn dữ liệu.

2.2.1 Yêu cầu chức năng

Bảng 2.1. Yêu cầu chức năng của hệ thống

Mã yêu cầu	Tên yêu cầu	Mô tả
FR01	Đăng nhập hệ thống	Người dùng có thể đăng nhập bằng tài khoản nội bộ hoặc tài khoản Microsoft
FR02	Phân quyền người dùng	Hệ thống phân quyền theo vai trò Admin, Cán bộ quản lý và Giảng viên
FR03	Quản lý tài khoản	Admin có thể thêm, sửa, xóa, tìm kiếm và gán vai trò cho tài khoản
FR04	Quản lý giảng viên	Hệ thống cho phép quản lý thông tin giảng viên phục vụ công tác xếp lịch
FR05	Quản lý phòng học	Hệ thống cho phép quản lý mã phòng, khu nhà, sức chứa và loại phòng
FR06	Quản lý lớp học phần	Hệ thống cho phép quản lý danh sách lớp học phần cần xếp lịch trong học kỳ
FR07	Nhập dữ liệu từ Excel	Người dùng có thể nhập dữ liệu giảng viên, phòng học hoặc lớp học phần từ tệp Excel

FR08	Cấu hình tham số hệ thống	Admin có thể cấu hình các trọng số và tham số phục vụ thuật toán xếp lịch
FR09	Xếp thời khóa biểu tự động	Hệ thống sử dụng thuật toán tối ưu để tạo phương án thời khóa biểu
FR10	Kiểm tra xung đột lịch	Hệ thống kiểm tra các xung đột về giảng viên, phòng học và thời gian học
FR11	Quản lý kết quả xếp lịch	Người dùng có quyền có thể xem, lưu, cập nhật hoặc xuất kết quả xếp lịch
FR12	Dashboard thống kê	Hệ thống hiển thị thống kê về lớp học phần, phòng học, giảng viên và kết quả xếp lịch
FR13	Tra cứu lịch giảng viên	Giảng viên có thể xem thời khóa biểu cá nhân
FR14	Xuất thời khóa biểu	Hệ thống hỗ trợ xuất dữ liệu thời khóa biểu để phục vụ quản lý và báo cáo

Các yêu cầu chức năng trên phản ánh các nghiệp vụ chính trong quá trình vận hành hệ thống. Trong đó, các chức năng quản lý dữ liệu đóng vai trò chuẩn bị dữ liệu đầu vào; chức năng cấu hình và xếp lịch là trung tâm của hệ thống; chức năng tra cứu và thống kê giúp khai thác kết quả sau khi thời khóa biểu được tạo ra.

2.2.2 Yêu cầu phi chức năng

Ngoài các chức năng nghiệp vụ, hệ thống cần đáp ứng các yêu cầu phi chức năng để bảo đảm khả năng sử dụng trong thực tế.

Bảng 2.2. Yêu cầu phi chức năng của hệ thống

Mã yêu cầu	Tên yêu cầu	Mô tả
NFR01	Tính chính xác	Kết quả xếp lịch không được vi phạm các ràng buộc cứng
NFR02	Tính bảo mật	Hệ thống phải xác thực người dùng và kiểm soát quyền truy cập
NFR03	Tính dễ sử dụng	Giao diện cần trực quan, dễ thao tác với người dùng quản lý đào tạo
NFR04	Tính mở rộng	Hệ thống có thể bổ sung thêm ràng buộc, vai trò hoặc chức năng mới
NFR05	Tính linh hoạt	Các trọng số thuật toán có thể thay đổi thông qua cấu hình hệ thống
NFR06	Tính toàn vẹn dữ liệu	Dữ liệu giảng viên, phòng học, lớp học phần và thời khóa biểu phải nhất quán
NFR07	Hiệu năng	Hệ thống cần xử lý được dữ liệu học kỳ với số lượng lớp học phần và phòng học lớn
NFR08	Khả năng bảo trì	Mã nguồn và cơ sở dữ liệu cần được tổ chức rõ ràng, thuận tiện cho phát triển tiếp

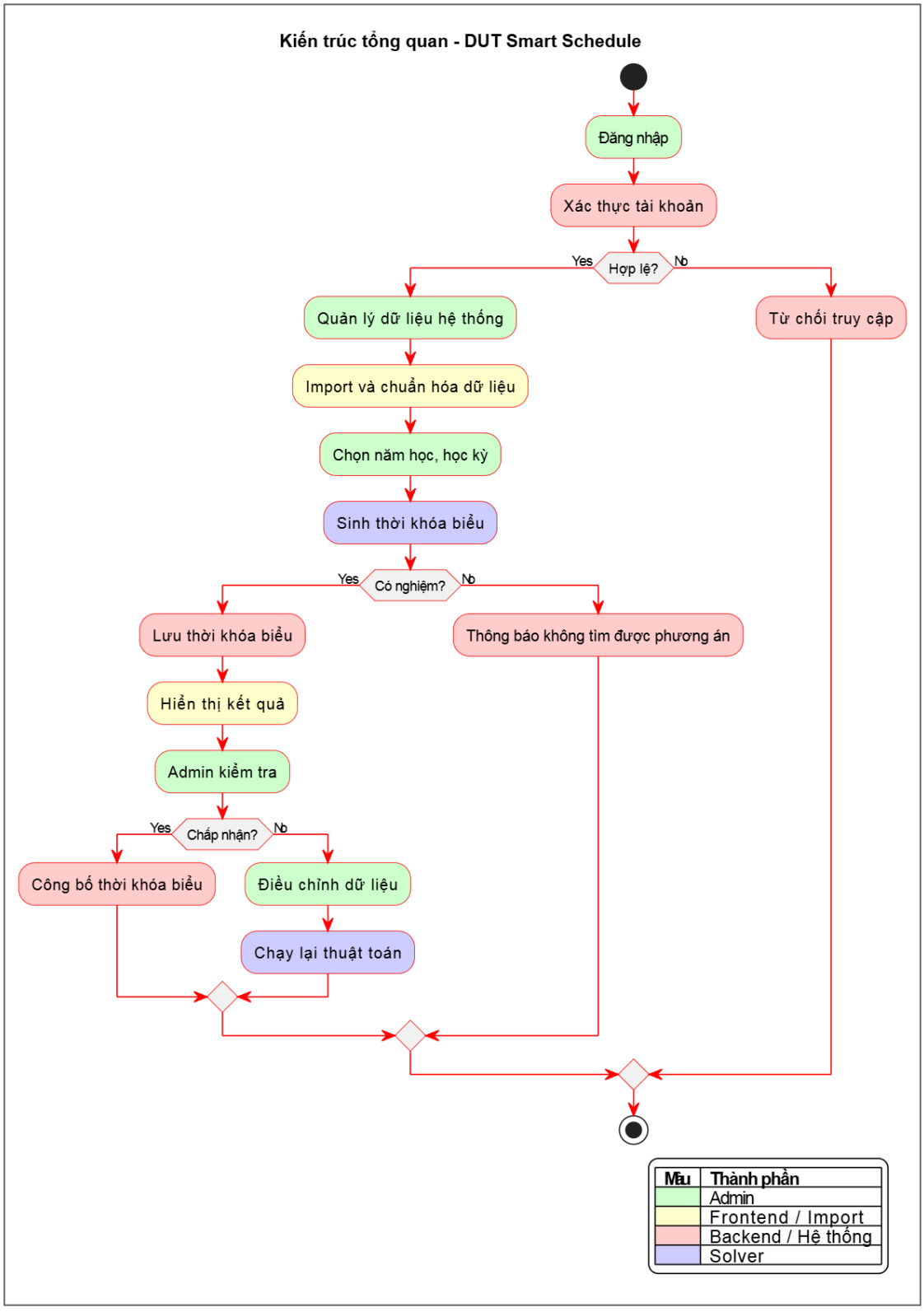
2.3 Tổng quan nguyên lý hệ thống

Hệ thống DUT Smart Schedule được xây dựng theo mô hình ứng dụng web, trong đó dữ liệu được lưu trữ tập trung trên cơ sở dữ liệu và được truy cập thông qua các dịch vụ API. Người dùng thao tác với hệ thống thông qua giao diện web, các yêu cầu được gửi đến Backend để xử lý nghiệp vụ, truy vấn dữ liệu hoặc thực thi thuật toán xếp thời khóa biểu.

Nguyên lý hoạt động tổng quát của hệ thống được mô tả như sau:

Người dùng đăng nhập vào hệ thống.

1. Hệ thống xác thực tài khoản và phân quyền theo vai trò.
2. Dữ liệu giảng viên, phòng học và lớp học phần được nhập hoặc cập nhật vào cơ sở dữ liệu.
3. Quản trị viên thiết lập các tham số phục vụ quá trình xếp lịch.
4. Thuật toán CP-SAT thực hiện quá trình phân công thời gian học và phòng học.
5. Kết quả thời khóa biểu được lưu vào cơ sở dữ liệu.
6. Người dùng tra cứu và khai thác kết quả thông qua giao diện web.



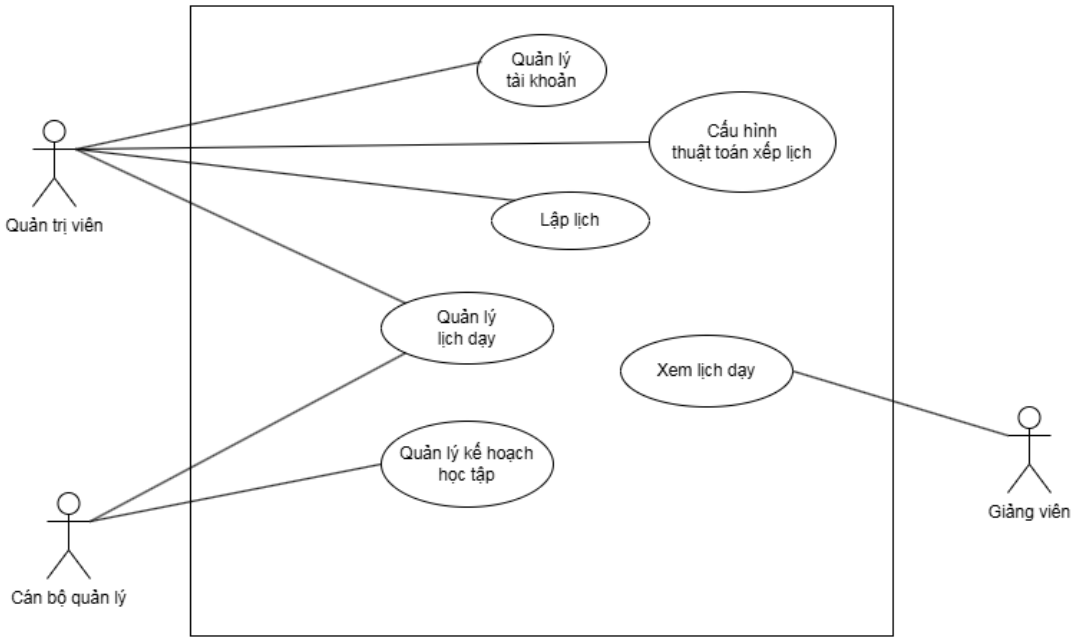
Hình 2.1. Kiến trúc tổng quan của hệ thống DUT Smart Schedule

Từ nguyên lý hoạt động tổng quát này, các chức năng cụ thể của hệ thống sẽ được mô hình hóa thông qua các biểu đồ Use Case và quy trình nghiệp vụ ở các mục tiếp theo.

2.4 Use case của hệ thống

2.4.1 Biểu đồ use case tổng quan hệ thống

Sau đây là sơ đồ use case tổng quan thể hiện toàn bộ các chức năng chính của website quản lý và sắp xếp thời khoá biểu, phòng học tại trường đại học.



Hình 2.2. Biểu đồ use case tổng quan hệ thống

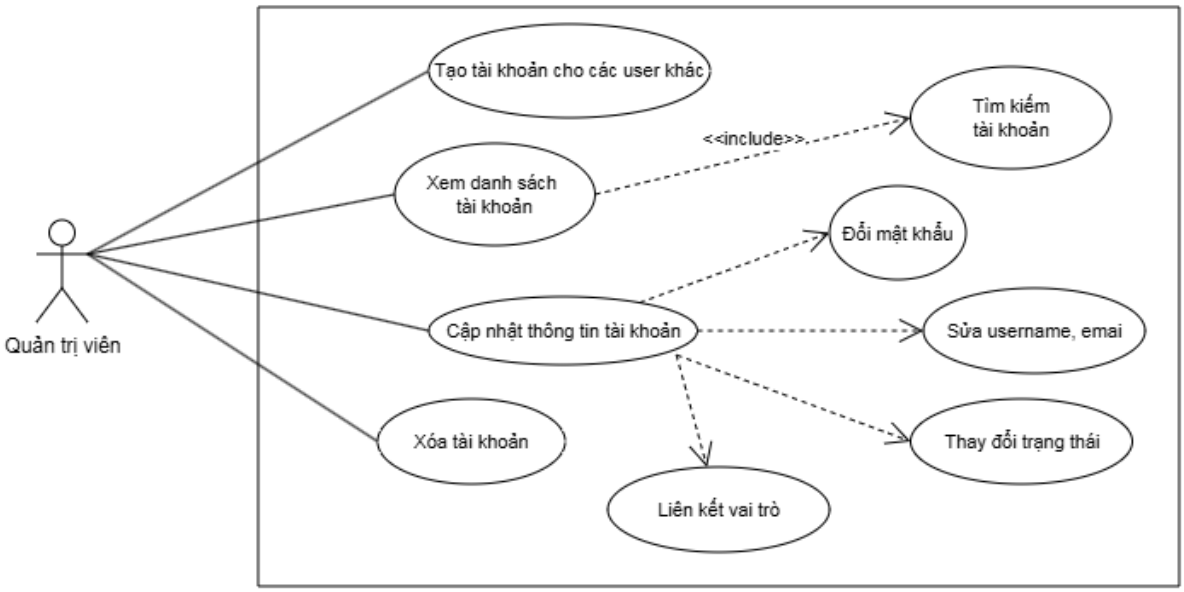
Biểu đồ use case tổng quan xác định ba nhóm tác nhân chính của hệ thống là Quản trị viên, Cán bộ quản lý và Giảng viên. Mỗi tác nhân đảm nhận những vai trò và nhiệm vụ khác nhau trong quá trình quản lý dữ liệu đào tạo và xây dựng thời khóa biểu của nhà trường.

Tác nhân Quản trị viên là người có quyền cao nhất trong hệ thống. Ngoài chức năng quản lý tài khoản người dùng nhằm bảo đảm việc phân quyền và kiểm soát truy cập hệ thống, Quản trị viên còn có thể thực hiện chức năng cấu hình thuật toán xếp lịch. Chức năng này cho phép thiết lập các tham số, ràng buộc và tiêu chí phục vụ quá trình xây dựng thời khóa biểu tự động. Bên cạnh đó, Quản trị viên cũng có quyền thực hiện chức năng lập lịch để tiến hành tạo thời khóa biểu cho từng học kỳ dựa trên dữ liệu đào tạo hiện có.

Tác nhân Cán bộ quản lý là người trực tiếp tham gia vào công tác quản lý đào tạo. Trong hệ thống, cán bộ quản lý thực hiện các nhóm chức năng như quản lý kế hoạch học tập, quản lý lớp học phần, theo dõi lịch dạy và khai thác kết quả thời khóa biểu. Các chức năng này hỗ trợ cập nhật và quản lý các thông tin liên quan đến học phần, lớp học phần, kế hoạch mở lớp và các dữ liệu cần thiết phục vụ quá trình xây dựng thời khóa biểu.

Tác nhân Giảng viên là đối tượng sử dụng hệ thống với mục đích chính là theo dõi lịch giảng dạy của bản thân. Thông qua chức năng xem lịch dạy, giảng viên có thể tra cứu nhanh thời gian, phòng học và các lớp học phần được phân công trong từng học kỳ.

2.4.2 Biểu đồ use case phân rã quản lý tài khoản (Quản trị viên)

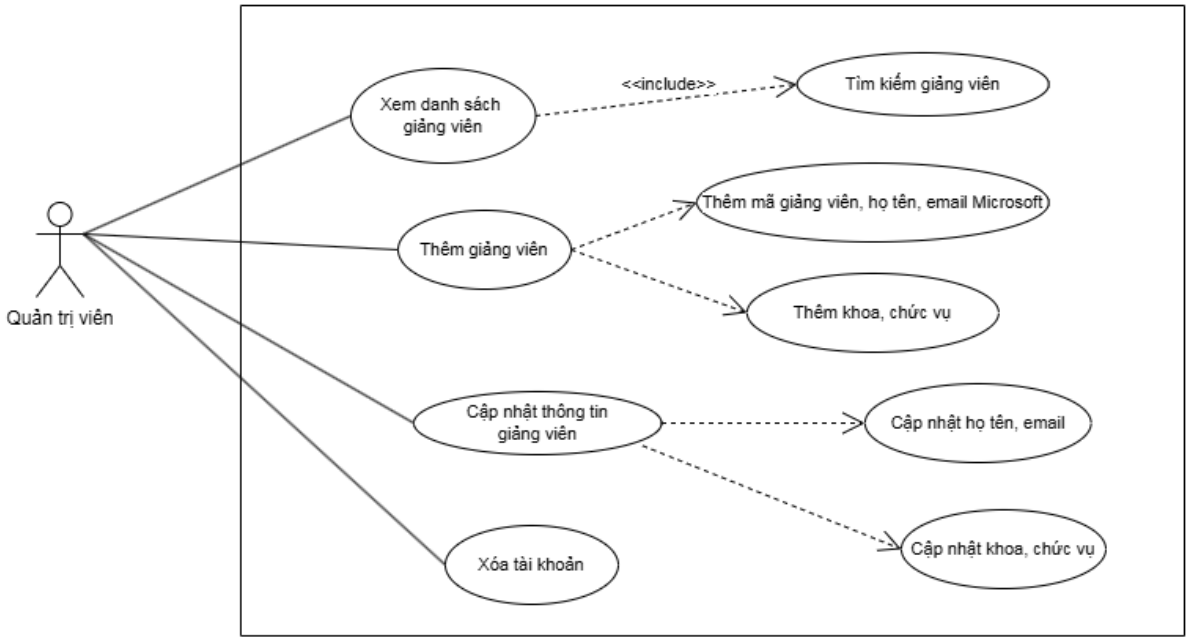


Hình 2.3. Biểu đồ use case phân rã quản lý tài khoản

Các chức năng được phân rã của use case quản lý tài khoản, thực hiện bởi tác nhân Quản trị viên, bao gồm tạo mới tài khoản, xem danh sách tài khoản, cập nhật thông tin tài khoản, xóa tài khoản, tìm kiếm tài khoản, đặt lại mật khẩu, thay đổi trạng thái hoạt động của tài khoản và liên kết vai trò cho từng người dùng.

Các chức năng này giúp Quản trị viên quản lý tập trung toàn bộ tài khoản trong hệ thống, đồng thời bảo đảm việc phân quyền và kiểm soát truy cập được thực hiện chính xác. Việc liên kết tài khoản với vai trò và thông tin giảng viên cũng giúp hệ thống xác định đúng phạm vi dữ liệu mà người dùng được phép truy cập.

2.4.3 Biểu đồ use case phân rã quản lý giảng viên

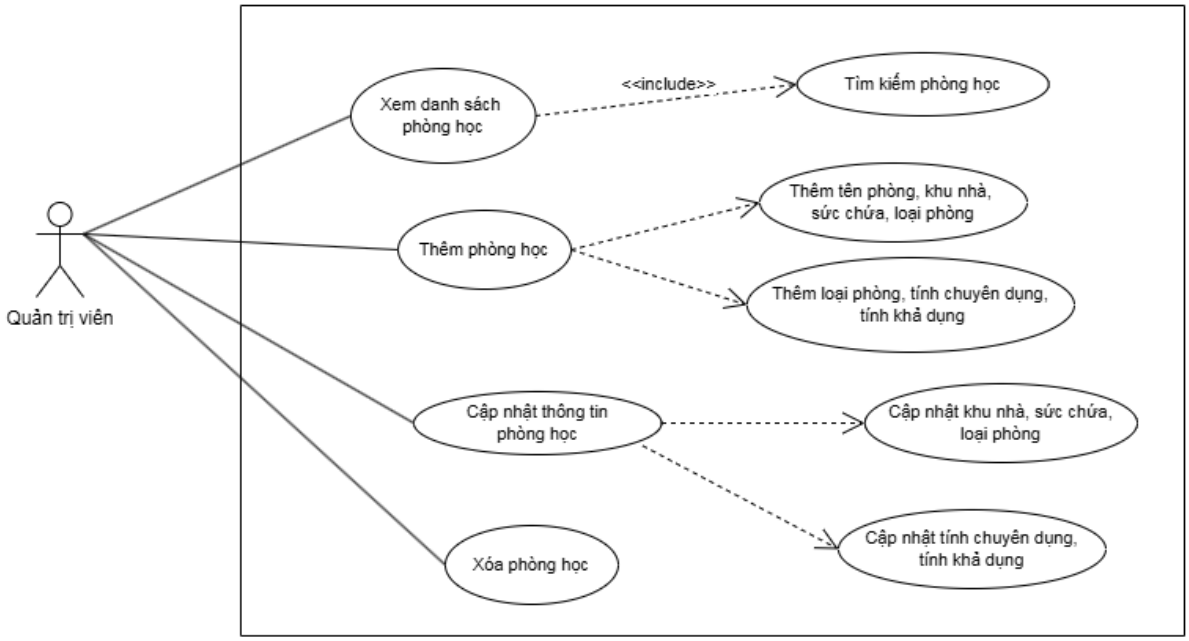


Hình 2.4. Biểu đồ use case phân rã quản lý giảng viên

Use case quản lý giảng viên cho phép Quản trị viên hoặc người dùng có quyền quản lý thực hiện các thao tác như xem danh sách giảng viên, thêm mới giảng viên, cập nhật thông tin giảng viên, xóa giảng viên và tìm kiếm giảng viên.

Khi thêm mới hoặc cập nhật giảng viên, hệ thống cho phép khai báo các thông tin như mã giảng viên, họ tên, khoa công tác và địa chỉ email Microsoft. Những dữ liệu này không chỉ phục vụ cho công tác quản lý mà còn là đầu vào quan trọng của thuật toán xếp thời khóa biểu, đặc biệt trong việc kiểm tra ràng buộc không trùng lịch giảng viên.

2.4.4 Biểu đồ use case phân rã quản lý phòng học

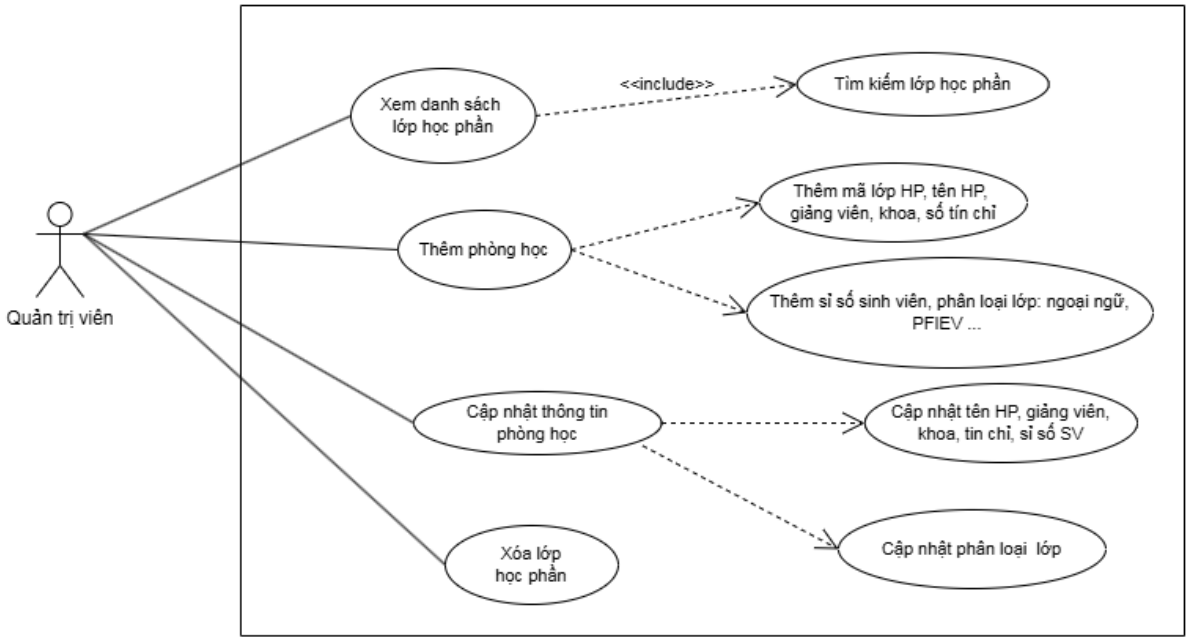


Hình 2.5. Biểu đồ use case phân rã quản lý phòng học

Use case quản lý phòng học cho phép Quản trị viên thực hiện các chức năng như xem danh sách phòng học, thêm mới phòng học, cập nhật thông tin phòng học, xóa phòng học và tìm kiếm phòng học.

Khi khai báo phòng học, hệ thống cần lưu trữ các thông tin như mã phòng, khu nhà, sức chứa và loại phòng. Ngoài ra, hệ thống có thể bổ sung các thuộc tính như phòng PFIEV, phòng Ngoại ngữ, phòng thực hành, phòng thí nghiệm hoặc phòng chuyên dụng khác. Các thông tin này được sử dụng trực tiếp trong quá trình lọc phòng hợp lệ và kiểm tra các ràng buộc của thuật toán xếp lịch.

2.4.5 Biểu đồ use case phân rã quản lý lớp học phần

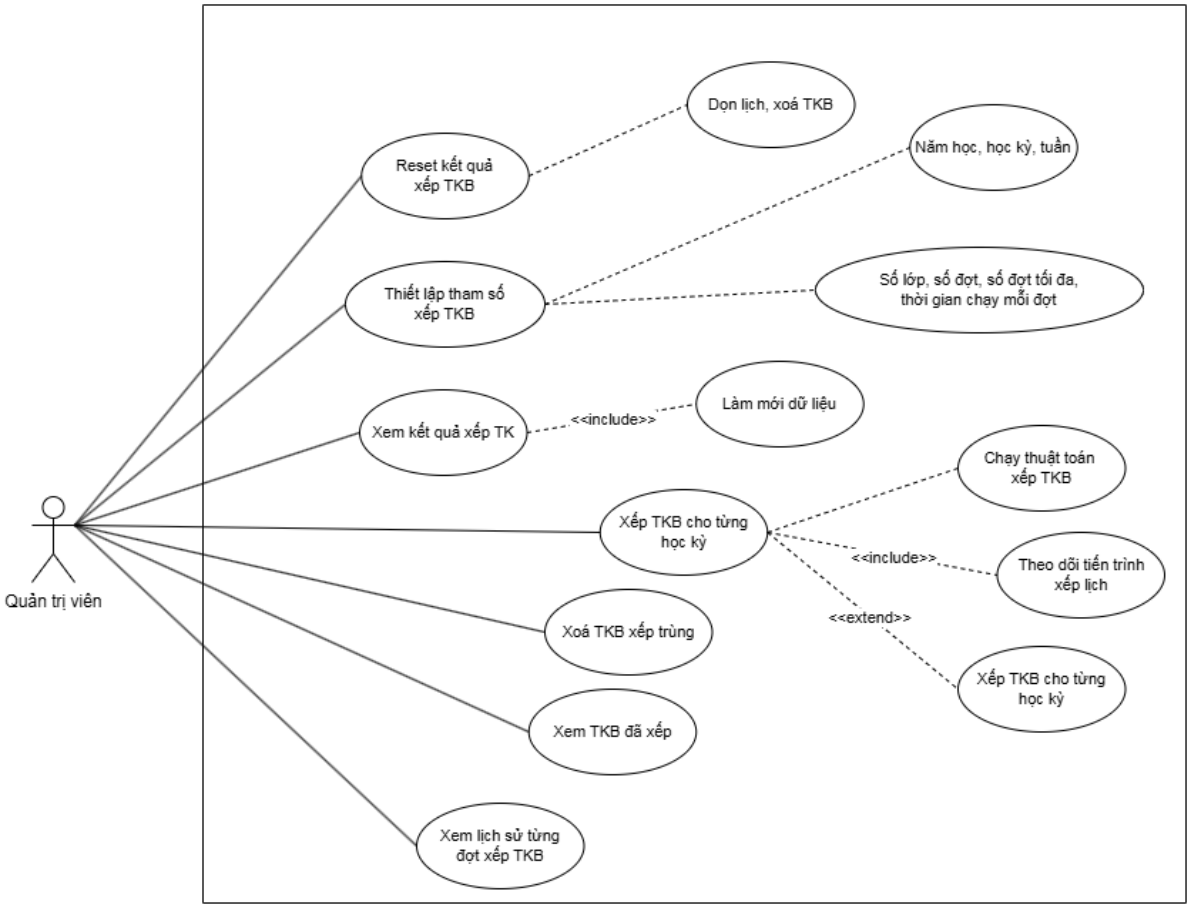


Hình 2.6. Biểu đồ use case phân rã quản lý lớp học phần

Use case quản lý lớp học phần cho phép người dùng có quyền thực hiện các thao tác như xem danh sách lớp học phần, thêm mới lớp học phần, cập nhật thông tin lớp học phần, xóa lớp học phần, tìm kiếm lớp học phần và nhập dữ liệu từ tệp Excel.

Khi thực hiện thêm mới hoặc cập nhật lớp học phần, người dùng có thể khai báo các thông tin như mã lớp học phần, tên học phần, giảng viên phụ trách, khoa quản lý, số tín chỉ, số tiết, sĩ số sinh viên và loại lớp học phần. Ngoài ra, hệ thống còn hỗ trợ nhận diện các lớp đặc thù như lớp PFIEV, lớp Ngoại ngữ, lớp thực hành hoặc lớp thí nghiệm để phục vụ quá trình xếp phòng học phù hợp.

2.4.6 Biểu đồ use case phân rã xếp thời khoá biểu học kỳ



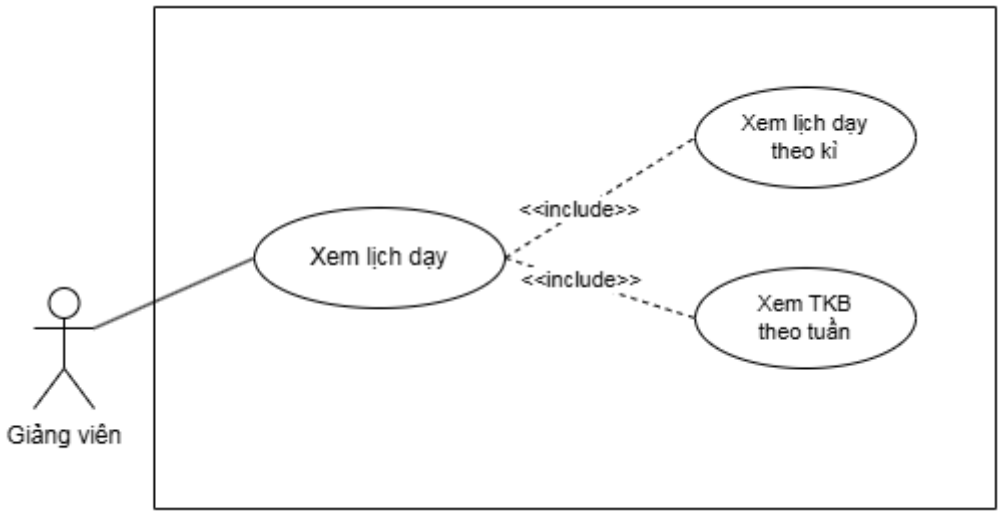
Hình 2.7. Biểu đồ use case phân rã xếp thời khoá biểu học kỳ

Use case xếp thời khóa biểu học kỳ là một trong những nhóm chức năng quan trọng nhất của hệ thống. Trước khi tiến hành xếp thời khóa biểu, Quản trị viên có thể thiết lập các tham số như năm học, học kỳ, số lớp học phần được xử lý trong mỗi đợt, số đợt thực hiện, thời gian chạy tối đa và các trọng số phạt của ràng buộc mềm.

Bên cạnh đó, hệ thống cũng cung cấp chức năng reset kết quả xếp thời khóa biểu nhằm dọn dẹp dữ liệu hoặc xóa các kết quả đã được tạo trước đó khi cần thực hiện lại quá trình xếp lịch. Sau khi thiết lập dữ liệu và tham số, Quản trị viên có thể khởi chạy thuật toán xếp lịch. Hệ thống sẽ tự động chạy bộ giải CP-SAT, theo dõi tiến trình xử lý và lưu kết quả vào cơ sở dữ liệu.

Sau khi quá trình xếp lịch hoàn thành, Quản trị viên có thể xem kết quả xếp thời khóa biểu, xem lịch sử các đợt xếp lịch, làm mới dữ liệu hoặc khóa các lịch đã được duyệt để sử dụng chính thức.

2.4.7 Biểu đồ use case phân rã giảng viên xem lịch dạy



Hình 2.8. Biểu đồ use case phân rã giảng viên xem lịch dạy

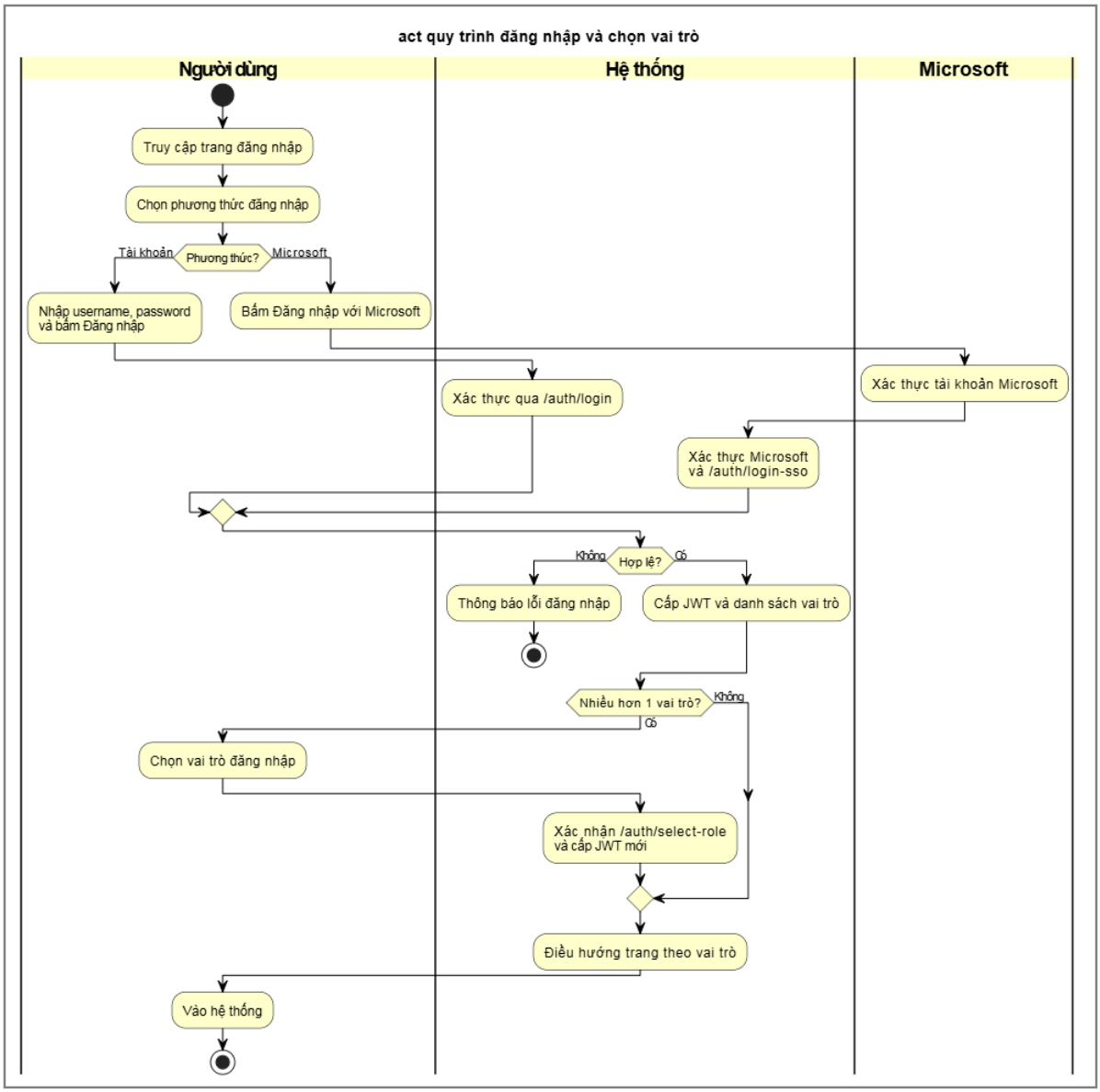
Use case giảng viên xem lịch dạy giúp giảng viên theo dõi và tra cứu lịch giảng dạy của bản thân sau khi thời khóa biểu học kỳ đã được xây dựng và phê duyệt.

Thông qua chức năng xem lịch dạy, giảng viên có thể theo dõi các lớp học phần được phân công giảng dạy, bao gồm thông tin về thời gian học, phòng học và lớp học phần tương ứng. Để thuận tiện cho việc tra cứu, hệ thống hỗ trợ hai hình thức hiển thị là xem lịch dạy theo học kỳ và xem thời khóa biểu theo tuần.

2.5 Quy trình nghiệp vụ

Quy trình nghiệp vụ mô tả trình tự thực hiện các thao tác chính trong hệ thống. Các quy trình được trình bày gồm đăng nhập và chọn vai trò, chạy thuật toán sắp xếp thời khóa biểu, cập nhật lớp học phần và khóa lịch, xử lý khi không tìm được lời giải.

2.5.1 Quy trình nghiệp vụ đăng nhập và chọn vai trò



Hình 2.9. Quy trình nghiệp vụ đăng nhập và chọn vai trò

Quy trình đăng nhập và chọn vai trò bắt đầu khi người dùng truy cập vào màn hình đăng nhập của hệ thống và lựa chọn phương thức đăng nhập mong muốn. Hệ thống hỗ trợ hai hình thức đăng nhập là sử dụng tài khoản nội bộ hoặc đăng nhập bằng tài khoản Microsoft của nhà trường thông qua cơ chế Single Sign-On.

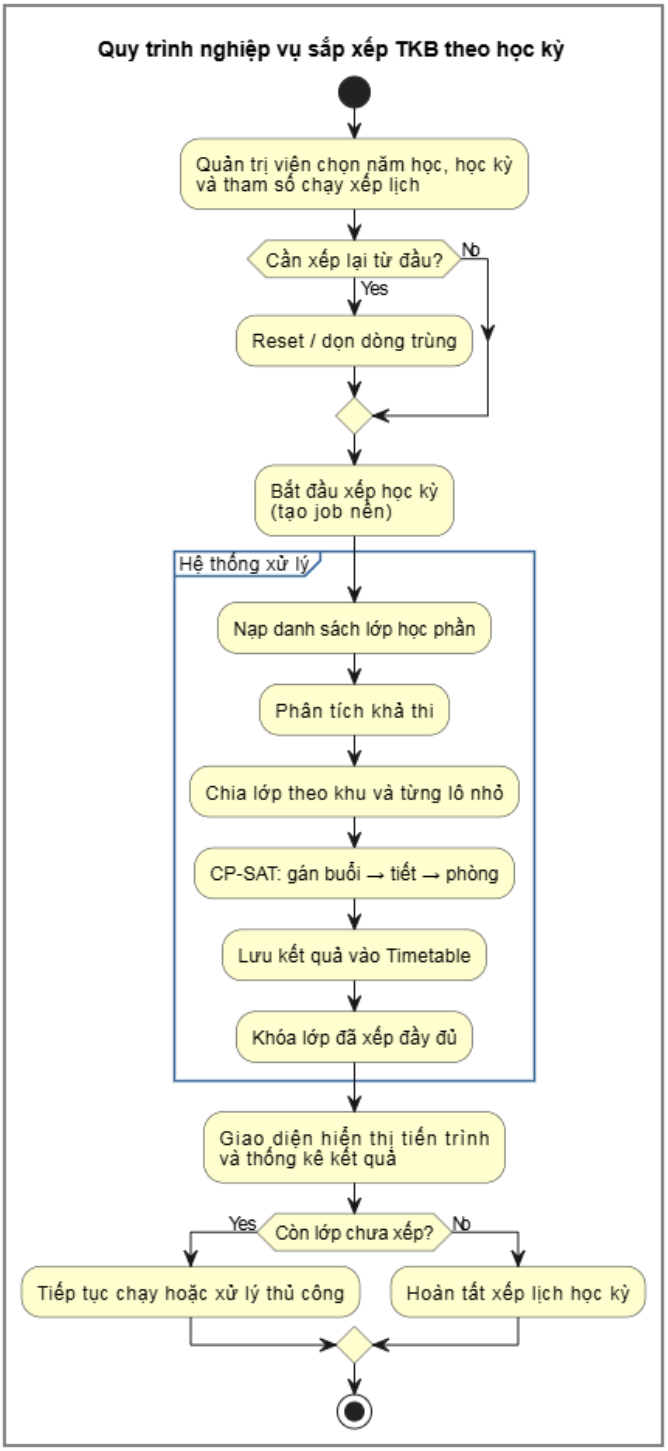
Đối với hình thức đăng nhập bằng tài khoản nội bộ, người dùng nhập tên đăng nhập và mật khẩu rồi gửi yêu cầu xác thực đến hệ thống. Đối với hình thức đăng nhập bằng tài khoản Microsoft, người dùng được chuyển hướng đến dịch vụ xác thực của

Microsoft để thực hiện đăng nhập và cấp quyền truy cập. Sau khi nhận được thông tin xác thực, hệ thống tiến hành kiểm tra tính hợp lệ của tài khoản thông qua các API xác thực tương ứng.

Nếu thông tin đăng nhập không hợp lệ, hệ thống sẽ thông báo lỗi và yêu cầu người dùng thực hiện đăng nhập lại. Ngược lại, nếu quá trình xác thực thành công, hệ thống sẽ cấp JWT và lấy danh sách các vai trò được gán cho tài khoản đăng nhập.

Trong trường hợp tài khoản chỉ được gán một vai trò duy nhất, hệ thống tự động xác định vai trò và chuyển người dùng đến giao diện chức năng tương ứng. Nếu tài khoản được gán nhiều vai trò khác nhau, hệ thống hiển thị danh sách vai trò để người dùng lựa chọn. Sau khi người dùng xác nhận vai trò muốn sử dụng, hệ thống cấp lại JWT tương ứng với vai trò đã chọn và điều hướng đến trang làm việc phù hợp.

2.5.2 Quy trình nghiệp vụ chạy thuật toán sắp xếp TKB theo học kỳ



Hình 2.10. Quy trình nghiệp vụ sắp xếp TKB theo học kỳ

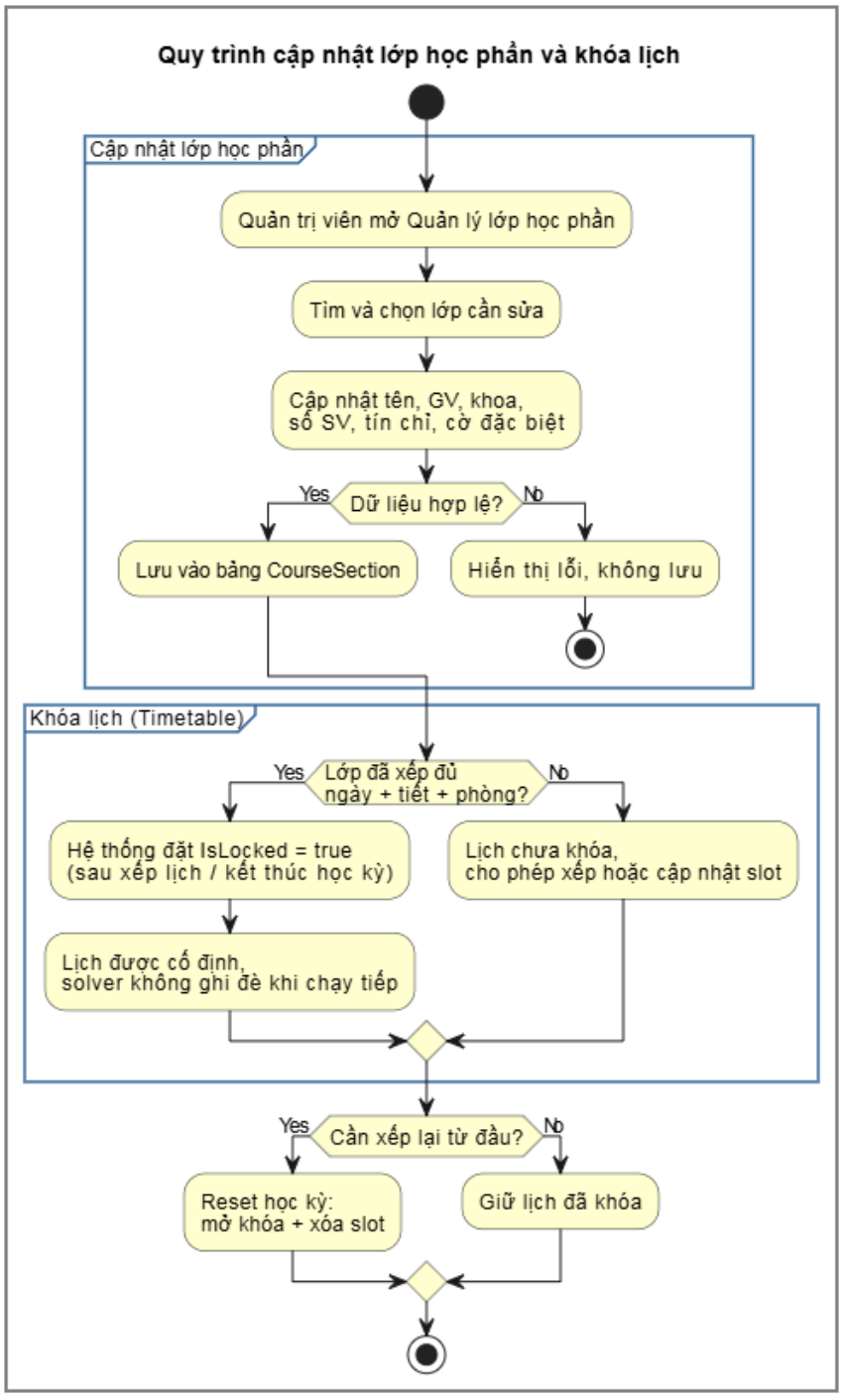
Quy trình xếp thời khóa biểu cho học kỳ bắt đầu khi Quản trị viên lựa chọn năm học, học kỳ và thiết lập các tham số cần thiết cho quá trình xếp lịch. Trong trường hợp học kỳ đã từng được xếp trước đó và cần thực hiện lại, hệ thống cho phép người dùng lựa chọn reset dữ liệu hoặc dọn dẹp các bản ghi thời khóa biểu trùng lặp trước khi bắt đầu quá trình xử lý mới.

Sau khi hoàn tất các bước chuẩn bị, Quản trị viên thực hiện lệnh xếp thời khóa biểu cho học kỳ. Hệ thống tạo một tiến trình xử lý nền và bắt đầu nạp dữ liệu các lớp học phần thuộc học kỳ được chọn. Tiếp theo, hệ thống phân tích dữ liệu đầu vào để kiểm tra tính khả thi của bài toán, đồng thời xác định các ràng buộc liên quan đến giảng viên, phòng học và thời gian học.

Để nâng cao hiệu quả xử lý, các lớp học phần có thể được chia thành nhiều nhóm nhỏ theo từng batch trước khi đưa vào thuật toán xếp lịch. Sau đó, hệ thống sử dụng mô hình tối ưu hóa CP-SAT để thực hiện quá trình phân công buổi học, tiết học và phòng học cho từng lớp học phần sao cho thỏa mãn các ràng buộc đã được thiết lập. Kết quả của mỗi đợt xử lý sẽ được lưu vào cơ sở dữ liệu thời khóa biểu để phục vụ việc tra cứu và quản lý.

Sau khi hoàn thành quá trình xếp lịch, hệ thống hiển thị tiến trình thực hiện cùng các thông tin thống kê kết quả cho người dùng theo dõi. Nếu vẫn còn các lớp học phần chưa được sắp xếp thành công do thiếu tài nguyên hoặc không thỏa mãn ràng buộc, Quản trị viên có thể tiếp tục thực hiện các đợt xử lý bổ sung hoặc can thiệp thủ công để hoàn thiện thời khóa biểu.

2.5.3 Quy trình nghiệp vụ cập nhật lớp học phần và khoá lịch



Hình 2.11. Quy trình nghiệp vụ cập nhật lớp học phần và khoá lịch

Quy trình cập nhật lớp học phần và khóa lịch được sử dụng trong trường hợp Quản trị viên cần điều chỉnh thông tin lớp học phần hoặc cố định kết quả thời khóa biểu sau khi quá trình xếp lịch hoàn thành.

Đầu tiên, Quản trị viên truy cập chức năng quản lý lớp học phần và lựa chọn lớp học phần cần chỉnh sửa. Tại đây, người dùng có thể cập nhật các thông tin liên quan như tên học phần, giảng viên phụ trách, khoa quản lý, số lượng sinh viên, số tín chỉ hoặc các

thuộc tính đặc biệt của lớp học phần. Sau khi hoàn tất việc chỉnh sửa, hệ thống tiến hành kiểm tra tính hợp lệ của dữ liệu đầu vào. Nếu dữ liệu không hợp lệ, hệ thống hiển thị thông báo lỗi và từ chối lưu thay đổi. Ngược lại, nếu dữ liệu hợp lệ, các thông tin mới được cập nhật vào cơ sở dữ liệu lớp học phần.

Sau khi dữ liệu lớp học phần được cập nhật, hệ thống kiểm tra trạng thái thời khóa biểu của lớp học phần đó. Nếu lớp học phần đã được phân công ngày học, tiết học và phòng học, Quản trị viên có thể thực hiện thao tác khóa lịch. Khi lịch được khóa, hệ thống đánh dấu trạng thái cố định cho lớp học phần, đồng nghĩa với việc các lần chạy thuật toán xếp lịch tiếp theo sẽ không thay đổi các buổi học đã được khóa trước đó.

Trong trường hợp lớp học phần chưa được xếp lịch hoặc chưa được khóa, hệ thống vẫn cho phép tiếp tục xếp lịch hoặc điều chỉnh các thông tin liên quan trong những lần xử lý tiếp theo. Ngoài ra, nếu Quản trị viên muốn thực hiện lại quá trình xếp lịch cho toàn bộ học kỳ, hệ thống cho phép lựa chọn xếp lại từ đầu.

2.5.4 Quy trình xử lý khi không tìm được lời giải

Trong một số trường hợp, bộ giải có thể không tìm được phương án thời khóa biểu hợp lệ. Nguyên nhân thường đến từ việc dữ liệu đầu vào thiếu tài nguyên hoặc các ràng buộc cứng quá chặt. Ví dụ, số lớp học phần cần xếp quá lớn so với số phòng học và khung thời gian hiện có; một số lớp có sĩ số lớn nhưng không có phòng học đủ sức chứa; hoặc nhiều lớp cùng giảng viên bị yêu cầu xếp trong các khoảng thời gian hạn chế.

Khi không tìm được lời giải, hệ thống cần thực hiện quy trình xử lý như sau:

1. Kiểm tra dữ liệu đầu vào, bao gồm lớp học phần, phòng học, giảng viên và khung thời gian.
2. Xác định nhóm ràng buộc có khả năng gây lỗi, chẳng hạn thiếu phòng học phù hợp hoặc giảng viên bị quá tải.
3. Hiển thị thông báo cho người dùng về nguyên nhân có thể gây không khả thi.
4. Gợi ý người dùng điều chỉnh dữ liệu hoặc cấu hình, ví dụ mở thêm khung thời gian, bổ sung phòng học, giảm số ràng buộc ưu tiên hoặc phân công lại giảng viên.
5. Cho phép người dùng chạy lại thuật toán sau khi dữ liệu được điều chỉnh.

Việc bổ sung quy trình này giúp hệ thống không chỉ tạo thời khóa biểu mà còn hỗ trợ người quản lý trong quá trình phân tích và xử lý các tình huống xếp lịch phức tạp.

2.6 Đặc tả chức năng

2.6.1 Đặc tả use case đăng nhập hệ thống

Bảng 2.3. Đặc tả use case đăng nhập hệ thống

<table>
<tr><td>Mã use case</td><td>UC-01</td><td>Tên use case</td><td>Đăng nhập hệ thống</td></tr>
<tr><td>Tác nhân</td><td colspan="3">Người dùng (Admin / Cán bộ quản lý / Giảng viên)</td></tr>
<tr><td>Mục đích</td><td colspan="3">Xác thực danh tính và cấp quyền truy cập theo vai trò</td></tr>
<tr><td>Tiền điều kiện</td><td colspan="3">Tài khoản đã được Admin tạo; trạng thái hoạt động là đang hoạt động (IsActive = true); đã gán ít nhất một Role</td></tr>
<tr><td>Luồng sự kiện chính
(Thành công)</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1</td><td>Người dùng</td><td>Truy cập trang đăng nhập, nhập Username và Password</td></tr>
<tr><td>2</td><td>Hệ thống</td><td>Kiểm tra dữ liệu đầu vào (Username, Password không rỗng)</td></tr>
<tr><td>3</td><td>Hệ thống</td><td>Xác thực tài khoản, kiểm tra trạng thái kích hoạt và danh sách Role</td></tr>
<tr><td>4</td><td>Hệ thống</td><td>Tra cứu LecturerID liên kết với tài khoản (nếu có)</td></tr>
<tr><td>5</td><td>Hệ thống</td><td>Phát hành JWT (8 giờ), trả thông tin đăng nhập thành công</td></tr>
<tr><td>6</td><td>Hệ thống</td><td>Lưu token vào phiên, chuyển hướng trang mặc định theo role</td></tr>
</table>
</td></tr>
<tr><td>Luồng sự kiện thay thế</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1a</td><td>Người dùng</td><td>Chọn đăng nhập bằng tài khoản Microsoft</td></tr>
<tr><td>1b</td><td>Hệ thống</td><td>MSAL xác thực, trả Microsoft ID Token; khớp email với MicrosoftEmail của Account hoặc Lecturer; tiếp tục bước 4–6 luồng chính</td></tr>
<tr><td>2a</td><td>Hệ thống</td><td>Thông báo lỗi: Vui lòng nhập đầy đủ mã đăng nhập và mật khẩu</td></tr>
<tr><td>3a</td><td>Hệ thống</td><td>Thông báo lỗi: Mã đăng nhập hoặc mật khẩu không đúng</td></tr>
<tr><td>3b</td><td>Hệ thống</td><td>Thông báo lỗi: Tài khoản hiện đang bị khóa</td></tr>
<tr><td>3c</td><td>Hệ thống</td><td>Thông báo lỗi: Tài khoản chưa được gán quyền đăng nhập</td></tr>
<tr><td>3d</td><td>Hệ thống</td><td>Thông báo lỗi: Không tìm thấy tài khoản liên kết với email Microsoft</td></tr>
<tr><td>5a</td><td>Hệ thống</td><td>Phát hiện tài khoản có nhiều Role; chuyển trang chọn vai trò</td></tr>
<tr><td>5b</td><td>Người dùng</td><td>Chọn Role tại trang chọn vai trò</td></tr>
</table>
</td></tr>
</table>

	5c	Hệ thống	Cấp JWT mới với Role đã chọn, tiếp tục bước 6 luồng chính
	1a	Người dùng	Chọn đăng nhập bằng tài khoản Microsoft
	1b	Hệ thống	MSAL xác thực, trả Microsoft ID Token; khớp email với MicrosoftEmail của Account hoặc Lecturer; tiếp tục bước 4–6 luồng chính
Hậu điều kiện	JWT lưu phiên; chuyển đến trang mặc định theo role (hoặc trang chọn role nếu có nhiều Role)		

Bảng 2.3 đặc tả chi tiết quy trình nghiệp vụ của chức năng đăng nhập hệ thống. Đây là chức năng đầu tiên mà người dùng thực hiện khi truy cập vào hệ thống DUT Smart Schedule. Thông qua quá trình xác thực, hệ thống kiểm tra tính hợp lệ của tài khoản, quyền truy cập và vai trò được gán cho người dùng trước khi cho phép sử dụng các chức năng tương ứng.

2.6.2 Đặc tả use case quản lý tài khoản

Bảng 2.4. Đặc tả use case quản lý tài khoản

Mã use case	UC-02	Tên use case	Quản lý tài khoản
Tác nhân	Admin		
Mục đích	Tạo, sửa, xóa tài khoản; gán quyền; liên kết giảng viên; hỗ trợ đăng nhập SSO		
Tiền điều kiện	Admin đã đăng nhập hệ thống		

Luồng sự kiện chính (Thành công)

STT	Thực hiện bởi	Hành động
1	Admin	Truy cập trang Quản lý tài khoản
2	Hệ thống	Hiển thị danh sách tài khoản (có tìm kiếm, phân trang)
3	Admin	Thực hiện thao tác: thêm mới, sửa hoặc xóa tài khoản
4	Hệ thống	Kiểm tra ràng buộc dữ liệu (Username duy nhất, có ít nhất một Role…)
5	Hệ thống	Lưu thay đổi vào Account, Account_Role; cập nhật liên kết Lecturer (nếu có)
6	Hệ thống	Thông báo thành công, cập nhật danh sách

<table>
<tr><td>Luồng sự kiện thay thế</td><td>
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>3a</td><td>Admin</td><td>Chọn sửa tài khoản, thay đổi Password, MicrosoftEmail, IsActive, Role hoặc LecturerID</td></tr>
<tr><td>3b</td><td>Admin</td><td>Chọn xóa tài khoản và xác nhận</td></tr>
<tr><td>4a</td><td>Hệ thống</td><td>Thông báo lỗi: Username đã tồn tại</td></tr>
<tr><td>4b</td><td>Hệ thống</td><td>Thông báo lỗi: Phải chọn ít nhất một Role</td></tr>
<tr><td>4c</td><td>Hệ thống</td><td>Thông báo lỗi: Giảng viên đã liên kết với tài khoản khác</td></tr>
<tr><td>5a</td><td>Hệ thống</td><td>Gỡ liên kết Lecturer, xóa Account_Role, xóa Account (khi thao tác xóa)</td></tr>
</table>
</td></tr>
<tr><td>Hậu điều kiện</td><td>Dữ liệu Account và Account_Role được cập nhật đúng theo thao tác</td></tr>
</table>

Bảng 2.4 trình bày chi tiết quy trình nghiệp vụ của chức năng quản lý tài khoản do Quản trị viên thực hiện. Chức năng này cho phép quản lý toàn bộ tài khoản người dùng trong hệ thống, bao gồm tạo mới, cập nhật, phân quyền và vô hiệu hóa tài khoản khi cần thiết.

2.6.3 Đặc tả use case Quản lý lớp học phần

Bảng 2.5. Đặc tả use case quản lý lớp học phần

<table>
<tr><td>Mã use case</td><td>UC-03</td><td>Tên use case</td><td>Quản lý lớp học phần</td></tr>
<tr><td>Tác nhân</td><td colspan="3">Admin, Cán bộ quản lý</td></tr>
<tr><td>Mục đích</td><td colspan="3">Duy trì dữ liệu lớp học phần phục vụ sắp xếp thời khoá biểu</td></tr>
<tr><td>Tiền điều kiện</td><td colspan="3">Người dùng đã đăng nhập; đã có dữ liệu Giảng viên và Khoa</td></tr>
<tr><td>Luồng sự kiện chính
(Thành công)</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1</td><td>Người dùng</td><td>Truy cập trang Quản lý lớp học phần</td></tr>
<tr><td>2</td><td>Hệ thống</td><td>Tải danh sách Giảng viên, Khoa; hiển thị danh sách lớp học phần</td></tr>
<tr><td>3</td><td>Người dùng</td><td>Nhập hoặc sửa thông tin lớp học phần (CourseID, CourseName, Lecturer, Faculty, Credits, số sinh viên, các cờ đặc biệt)</td></tr>
<tr><td>4</td><td>Hệ thống</td><td>Kiểm tra ràng buộc; tự gán IsPFIEV nếu CourseID bắt đầu bằng 122 hoặc 123</td></tr>
<tr><td>5</td><td>Hệ thống</td><td>Lưu bản ghi CourseSection</td></tr>
<tr><td>6</td><td>Hệ thống</td><td>Thông báo thành công, cập nhật danh sách</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td>Luồng sự kiện thay thế</td><td>
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>3a</td><td>Người dùng</td><td>Chọn xóa lớp học phần và xác nhận</td></tr>
<tr><td>4a</td><td>Hệ thống</td><td>Thông báo lỗi: CourseID đã tồn tại</td></tr>
<tr><td>4b</td><td>Hệ thống</td><td>Thông báo lỗi: Tên lớp học phần không được để trống</td></tr>
<tr><td>5a</td><td>Hệ thống</td><td>Thông báo lỗi: Không thể xóa lớp đã có bản ghi Timetable</td></tr>
<tr><td>5b</td><td>Hệ thống</td><td>Xóa CourseSection và cập nhật danh sách (khi lớp chưa có Timetable)</td></tr>
<tr><td>3a</td><td>Người dùng</td><td>Chọn xóa lớp học phần và xác nhận</td></tr>
</table>
</td></tr>
<tr><td>Hậu điều kiện</td><td>Dữ liệu CourseSection được tạo, cập nhật hoặc xóa hợp lệ</td></tr>
</table>

Bảng 2.5 mô tả chi tiết các bước thực hiện chức năng quản lý lớp học phần. Đây là một trong những chức năng quan trọng nhất của hệ thống vì dữ liệu lớp học phần là đầu vào trực tiếp cho quá trình xây dựng thời khóa biểu.

2.6.4 Các trường thông tin lớp học phần (CourseSection)

Các trường thông tin cần thiết để sắp xếp thời khoá biểu cho một lớp học phần được liệt kê ở bảng dưới đây:

Bảng 2.6. Các trường dữ liệu của thực thể CourseSection

STT	Tên trường thông tin	Kiểu dữ liệu	Mô tả / Vai trò trong xếp lịch
1	TimetableID	int	Khóa chính giúp định danh thời khoá biểu của mỗi lớp học phần
2	CourseID	nvarchar(100)	Mã lớp học phần; liên kết dữ liệu lớp cần xếp TKB
3	LecturerID	int	Giảng viên phụ trách lớp học phần
4	CourseName	nvarchar(200)	Tên lớp học phần
5	AcademicYear	varchar(100)	Năm học diễn ra lớp học phần
6	Semester	int	Học kỳ diễn ra lớp học phần
7	Session	int	Số tiết mỗi học phần
8	TotalWeek	int	Tổng số tuần dạy trong học kỳ
9	Week	nvarchar(200)	Phạm vi tuần dạy cụ thể (nếu lớp không dạy suốt học kỳ)

Bảng 2.6 liệt kê các trường thông tin chính của một lớp học phần được sử dụng trong hệ thống. Các thông tin như mã lớp học phần, tên học phần, giảng viên phụ trách, số lượng sinh viên, số tín chỉ và loại lớp học phần đều đóng vai trò quan trọng trong quá trình quản lý đào tạo cũng như xây dựng thời khóa biểu.

2.6.5 Đặc tả use case sắp xếp TKB theo học kỳ

Bảng 2.7. Đặc tả use case sắp xếp TKB theo học kỳ

<table>
<tr><td>Mã use case</td><td>UC-05</td><td>Tên use case</td><td>Sắp xếp TKB theo học kỳ</td></tr>
<tr><td>Tác nhân</td><td colspan="3">Admin</td></tr>
<tr><td>Mục đích</td><td colspan="3">Tự động phân bổ ngày, tiết và phòng học cho các lớp học phần trong một học kỳ</td></tr>
<tr><td>Tiền điều kiện</td><td colspan="3">Admin đã đăng nhập; đã nhập CourseSection, Giảng viên, Phòng học; đã cấu hình tham số thuật toán</td></tr>
<tr><td>Luồng sự kiện chính (Thành công)</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1</td><td>Admin</td><td>Truy cập trang Sắp xếp TKB học kỳ, chọn năm học và học kỳ</td></tr>
<tr><td>2</td><td>Hệ thống</td><td>Hiển thị thống kê số lớp, số lớp đã xếp, số lớp chưa xếp</td></tr>
<tr><td>3</td><td>Admin</td><td>Khởi chạy sắp xếp TKB học kỳ</td></tr>
<tr><td>4</td><td>Hệ thống</td><td>Tạo job nền, chia lớp học phần thành các batch</td></tr>
<tr><td>5</td><td>Hệ thống</td><td>Thuật toán CP-SAT xử lý từng batch, ghi kết quả vào Timetable</td></tr>
<tr><td>6</td><td>Admin</td><td>Theo dõi tiến trình job đến khi hoàn tất</td></tr>
<tr><td>7</td><td>Admin</td><td>Xem kết quả, khóa lịch batch đã duyệt</td></tr>
<tr><td>8</td><td>Hệ thống</td><td>Đánh dấu lịch đã khóa</td></tr>
</table>
</td></tr>
<tr><td>Luồng sự kiện thay thế</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>2a</td><td>Admin</td><td>Chọn Dedupe hoặc Reset dữ liệu học kỳ (chỉ batch chưa khóa) trước khi chạy</td></tr>
<tr><td>2b</td><td>Hệ thống</td><td>Dọn dữ liệu trùng lặp hoặc xóa slot chưa khóa; quay lại bước 2 luồng chính</td></tr>
<tr><td>5a</td><td>Hệ thống</td><td>Một số lớp không xếp được do thiếu phòng/GV/slot; hoàn tất job với phần lớn lớp đã xếp, báo danh sách lớp chưa xếp</td></tr>
<tr><td>5b</td><td>Hệ thống</td><td>Job timeout theo MaxTimeSecondsPerBatch; dừng batch hiện tại, giữ kết quả batch đã hoàn thành</td></tr>
<tr><td>6a</td><td>Admin</td><td>Chọn hủy job đang xử lý</td></tr>
<tr><td>6b</td><td>Hệ thống</td><td>Dừng job, cập nhật trạng thái hủy; giữ kết quả batch đã hoàn thành trước thời điểm hủy</td></tr>
</table>
</td></tr>
<tr><td>Hậu điều kiện</td><td colspan="3">Thời khoá biểu có thêm các thông tin về thứ ngày học, tiết bắt đầu, phòng học; các đợt có thể được khóa để sử dụng chính thức</td></tr>
</table>

Bảng 2.7 đã đặc tả chi tiết quy trình thực hiện chức năng chạy xếp thời khóa biểu học kỳ. Chức năng này được thực hiện bởi tác nhân Quản trị viên và là chức năng trung tâm của hệ thống. Dựa trên dữ liệu lớp học phần, giảng viên, phòng học và các tham số cấu hình, hệ thống tiến hành xây dựng thời khóa biểu tự động bằng thuật toán tối ưu hóa. Kết quả thu được là các phương án sắp xếp bảo đảm các ràng buộc về thời gian, phòng học và nguồn lực giảng dạy của nhà trường.

2.6.6 Đặc tả use case giảng viên xem thời khoá biểu

Bảng 2.8. Đặc tả use case giảng viên xem thời khoá biểu

<table>
<tr><td>Mã use case</td><td>UC-06</td><td>Tên use case</td><td>Giảng viên xem thời khoá biểu</td></tr>
<tr><td>Tác nhân</td><td colspan="3">Giảng viên</td></tr>
<tr><td>Mục đích</td><td colspan="3">Tra cứu lịch dạy cá nhân theo học kỳ và tuần</td></tr>
<tr><td>Tiền điều kiện</td><td colspan="3">Giảng viên đã đăng nhập; tài khoản đã liên kết LecturerID</td></tr>
<tr><td>Luồng sự kiện chính
(Thành công)</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1</td><td>Giảng viên</td><td>Truy cập trang Thời khoá biểu</td></tr>
<tr><td>2</td><td>Hệ thống</td><td>Lấy LecturerID từ phiên đăng nhập</td></tr>
<tr><td>3</td><td>Giảng viên</td><td>(Tuỳ chọn) Chọn năm học, học kỳ, tuần</td></tr>
<tr><td>4</td><td>Hệ thống</td><td>Truy vấn Timetable (chỉ bản ghi đã xếp đủ ngày, tiết, phòng)</td></tr>
<tr><td>5</td><td>Hệ thống</td><td>Tải cấu hình khung giờ từng tiết (LessonTimeConfig)</td></tr>
<tr><td>6</td><td>Hệ thống</td><td>Hiển thị lưới Thứ 2–7 × 14 tiết kèm tên lớp, phòng, khung giờ</td></tr>
</table>
</td></tr>
<tr><td>Luồng sự kiện thay thế</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1a</td><td>Giảng viên</td><td>Chuyển sang trang Lịch dạy dạng danh sách</td></tr>
<tr><td>6a</td><td>Hệ thống</td><td>Hiển thị danh sách buổi dạy theo học kỳ dạng bảng (thay vì lưới tuần)</td></tr>
<tr><td>2a</td><td>Hệ thống</td><td>Thông báo lỗi: Tài khoản chưa liên kết LecturerID</td></tr>
<tr><td>4a</td><td>Hệ thống</td><td>Thông báo: Chưa có lịch dạy cho học kỳ/tuần đã chọn</td></tr>
</table>
</td></tr>
<tr><td>Hậu điều kiện</td><td colspan="3">Giảng viên xem được lịch dạy đã được sắp xếp chính thức</td></tr>
</table>

Bảng 2.8 mô tả chi tiết chức năng giảng viên xem thời khóa biểu. Sau khi quá trình xếp lịch hoàn tất, giảng viên có thể sử dụng chức năng này để tra cứu lịch giảng dạy của bản thân theo học kỳ hoặc theo từng tuần học cụ thể. Chức năng giúp giảng viên chủ động hơn trong công tác giảng dạy, đồng thời giảm thời gian tra cứu và hạn chế các sai sót trong quá trình theo dõi lịch học.

2.6.7 Đặc tả use case Cấu hình tham số xếp lịch

Bảng 2.9. Đặc tả use case cấu hình tham số xếp lịch

<table>
<tr><td>Mã use case</td><td>UC-07</td><td>Tên use case</td><td>Cấu hình tham số thuật toán sắp xếp thời khoá biểu</td></tr>
<tr><td>Tác nhân</td><td colspan="3">Cấu hình tham số thuật toán sắp xếp thời khoá biểu</td></tr>
<tr><td>Mục đích</td><td colspan="3">Điều chỉnh tham số học kỳ, lịch dạy và trọng số thuật toán</td></tr>
<tr><td>Tiền điều kiện</td><td colspan="3">Admin đã đăng nhập hệ thống</td></tr>
<tr><td>Luồng sự kiện chính
(Thành công)</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>1</td><td>Admin</td><td>Truy cập trang Cấu hình hệ thống</td></tr>
<tr><td>2</td><td>Hệ thống</td><td>Tải toàn bộ tham số từ SystemConfig, nhóm theo Academic / Schedule / Solver</td></tr>
<tr><td>3</td><td>Admin</td><td>Chỉnh sửa tham số (năm học, học kỳ, ngày dạy, tiết, khung giờ, thời gian solver, trọng số phạt…)</td></tr>
<tr><td>4</td><td>Hệ thống</td><td>Kiểm tra giá trị hợp lệ (kiểu dữ liệu, phạm vi cho phép)</td></tr>
<tr><td>5</td><td>Hệ thống</td><td>Lưu cập nhật vào SystemConfigs</td></tr>
<tr><td>6</td><td>Hệ thống</td><td>Thông báo cập nhật thành công</td></tr>
</table>
</td></tr>
<tr><td>Luồng sự kiện thay thế</td><td colspan="3">
<table>
<tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr>
<tr><td>3a</td><td>Admin</td><td>Chọn khôi phục giá trị mặc định cho một hoặc nhiều tham số</td></tr>
<tr><td>4a</td><td>Hệ thống</td><td>Thông báo lỗi: Giá trị không hợp lệ (số âm, tiết ngoài 1–14, định dạng sai); không lưu thay đổi</td></tr>
<tr><td>5a</td><td>Hệ thống</td><td>Ghi giá trị mặc định và lưu (khi khôi phục mặc định)</td></tr>
</table>
</td></tr>
<tr><td>Hậu điều kiện</td><td colspan="3">Các tham số cầu hình được cập nhật; lần sắp xếp TKB tiếp theo áp dụng tham số mới</td></tr>
</table>

Bảng 2.9 trình bày quy trình cấu hình các tham số phục vụ cho việc xây dựng thời khóa biểu. Các tham số này bao gồm năm học, học kỳ, giới hạn thời gian xử lý, kích thước lô dữ liệu và các ràng buộc liên quan đến thuật toán xếp lịch. Việc cho phép cấu hình linh hoạt các tham số giúp hệ thống thích nghi với quy mô dữ liệu khác nhau của từng học kỳ, đồng thời hỗ trợ người quản lý điều chỉnh quá trình tối ưu hóa phù hợp với nhu cầu thực tế của nhà trường.

2.7 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu đóng vai trò trung tâm trong hệ thống DUT Smart Schedule, chịu trách nhiệm lưu trữ toàn bộ dữ liệu phục vụ công tác quản lý đào tạo và xây dựng thời khóa biểu. Việc thiết kế cơ sở dữ liệu được thực hiện theo mô hình dữ liệu quan hệ nhằm bảo đảm tính nhất quán, khả năng mở rộng và thuận lợi trong quá trình khai thác dữ liệu.

Các nhóm dữ liệu chính của hệ thống bao gồm:

- Dữ liệu người dùng và phân quyền.
- Dữ liệu giảng viên.
- Dữ liệu phòng học.
- Dữ liệu lớp học phần.
- Dữ liệu thời khóa biểu.
- Dữ liệu cấu hình hệ thống.
- Dữ liệu khóa lịch giảng viên và phòng học.

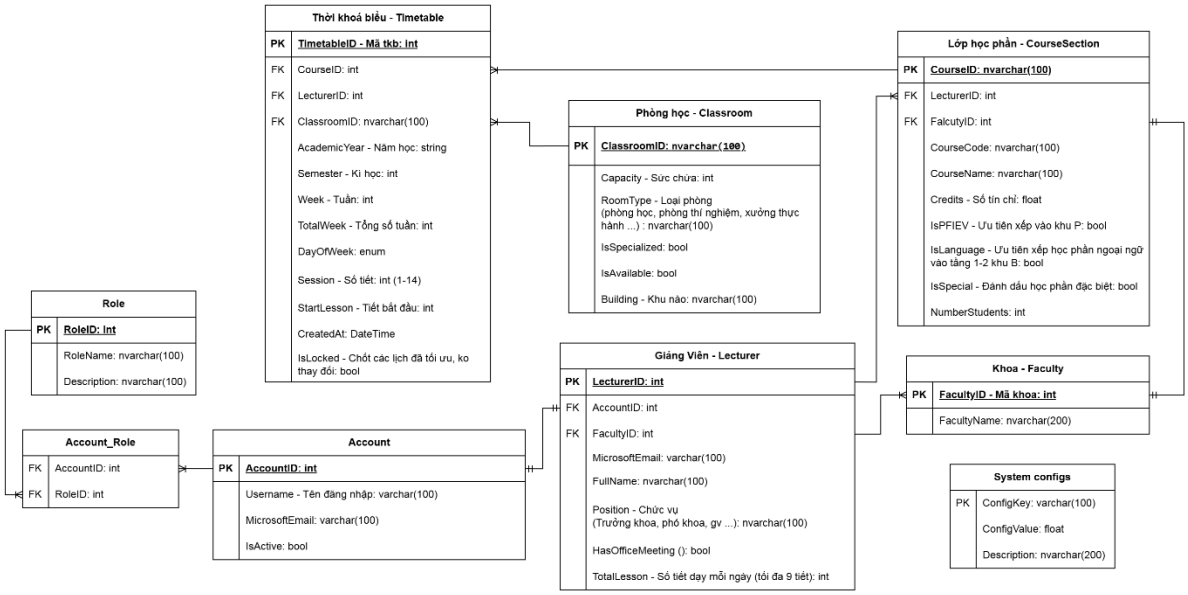
2.7.1 Mô hình quan hệ thực thể (ERD)

Mô hình ERD được xây dựng nhằm mô tả các thực thể chính trong hệ thống và mối quan hệ giữa các thực thể đó.

Các thực thể trung tâm bao gồm:

- Account
- Role
- Lecturer
- Faculty
- Classroom
- CourseSection
- Timetable
- SystemConfig

Thông qua các mối quan hệ giữa các thực thể, hệ thống có thể quản lý đồng bộ dữ liệu đào tạo và hỗ trợ quá trình xây dựng thời khóa biểu.



Hình 2.12. Mô hình thực thể liên kết (ERD) của hệ thống DUT Smart Schedule

2.7.2 Thiết kế dữ liệu phục vụ thuật toán

Để phục vụ bài toán xếp thời khóa biểu, cơ sở dữ liệu cần lưu trữ đầy đủ các nhóm dữ liệu sau:

Bảng 2.10. Các nhóm dữ liệu đầu vào phục vụ thuật toán xếp thời khóa biểu

Nhóm dữ liệu	Nội dung
Dữ liệu giảng viên	Mã giảng viên, họ tên, email, đơn vị, trạng thái, tài khoản liên kết
Dữ liệu phòng học	Mã phòng, khu, tầng, sức chứa, loại phòng, trạng thái
Dữ liệu lớp học phần	Mã lớp, tên học phần, số tín chỉ, số tiết, sĩ số, loại lớp, giảng viên phụ trách
Dữ liệu thời gian	Ngày học, buổi học, tiết học, trạng thái sử dụng hoặc khóa lịch
Dữ liệu thời khóa biểu	Lớp học phần, phòng học, giảng viên, ngày học, khung giờ, học kỳ
Dữ liệu cấu hình	Trọng số ràng buộc mềm, giới hạn thời gian chạy, tham số ưu tiên
Dữ liệu vai trò	Tài khoản, vai trò, quyền truy cập chức năng

2.7.3 Thiết kế bảng Faculty

Bảng 2.11. Thiết kế bảng Faculty

Trường	Kiểu dữ liệu	Mô tả
FacultyID	int	Mã khoa
FacultyName	nvarchar(200)	Tên khoa

Bảng Faculty được sử dụng để lưu trữ thông tin các khoa thuộc Trường Đại học Bách khoa – Đại học Đà Nẵng. Mỗi khoa được định danh bằng mã khoa (FacultyID) và tên khoa (FacultyName). Dữ liệu trong bảng này đóng vai trò làm danh mục dùng chung cho toàn hệ thống, phục vụ việc phân loại giảng viên, lớp học phần và các đối tượng đào tạo theo đơn vị quản lý. Thông qua bảng Faculty, hệ thống có thể tổ chức dữ liệu một cách khoa học và hỗ trợ các chức năng tra cứu, thống kê theo khoa.

2.7.4 Thiết kế bảng Lecturer

Bảng 2.12. Thiết kế bảng Lecturer

Trường	Kiểu dữ liệu	Mô tả
LecturerID	int	Mã giảng viên
LecturerName	nvarchar	Tên giảng viên
FacultyID	int	Mã khoa quản lý
MicrosoftEmail	nvarchar	Email Microsoft

Bảng Lecturer được sử dụng để lưu trữ thông tin đội ngũ giảng viên tham gia giảng dạy trong nhà trường. Các thông tin chính bao gồm mã giảng viên, họ tên, khoa quản lý và địa chỉ thư điện tử Microsoft do nhà trường cấp. Dữ liệu trong bảng này được sử dụng cho chức năng quản lý giảng viên, liên kết với tài khoản đăng nhập và phục vụ quá trình xây dựng thời khóa biểu. Thông qua mã giảng viên (LecturerID), hệ thống có thể xác định giảng viên phụ trách của từng lớp học phần và kiểm tra các ràng buộc liên quan đến lịch giảng dạy trong quá trình xếp lịch.

2.7.5 Thiết kế bảng Classroom

Bảng 2.13. Thiết kế bảng Classroom

Trường	Kiểu dữ liệu	Mô tả
ClassroomID	varchar	Mã phòng
Capacity	int	Sức chứa
RoomType	varchar	Loại phòng
Building	varchar	Khu nhà

Bảng Classroom lưu trữ thông tin các phòng học được sử dụng trong công tác đào tạo của nhà trường. Mỗi phòng học được mô tả thông qua các thuộc tính như mã phòng, sức chứa, loại phòng và khu nhà. Đây là nguồn dữ liệu quan trọng phục vụ quá trình phân bổ phòng học cho các lớp học phần. Trong quá trình xây dựng thời khóa biểu, hệ thống sử dụng thông tin từ bảng Classroom để kiểm tra sức chứa phòng học, phân loại phòng theo mục đích sử dụng và áp dụng các quy tắc phân khu phòng học theo yêu cầu thực tế của nhà trường.

2.7.6 Thiết kế bảng CourseSection

Bảng 2.14. Thiết kế bảng CourseSection

Trường	Kiểu dữ liệu	Mô tả
CourseID	varchar	Mã lớp học phần
CourseName	nvarchar	Tên lớp học phần
LecturerID	int	Giảng viên phụ trách
FacultyID	int	Mã khoa quản lý
NumberStudents	int	Sĩ số

Bảng CourseSection lưu trữ thông tin các lớp học phần được mở trong từng học kỳ. Đây là một trong những bảng dữ liệu quan trọng nhất của hệ thống vì toàn bộ quá trình xây dựng thời khóa biểu đều dựa trên dữ liệu của các lớp học phần. Các thông tin như mã lớp học phần, tên học phần, giảng viên phụ trách, khoa quản lý và số lượng sinh viên được sử dụng để xác định nhu cầu phòng học, kiểm tra các ràng buộc giảng dạy và làm dữ liệu đầu vào cho thuật toán xếp thời khóa biểu.

2.7.7 Thiết kế bảng Timetable

Bảng 2.15. Thiết kế bảng Timetable

Trường	Kiểu dữ liệu	Mô tả
TimetableID	int	Khóa chính
CourseID	varchar	Lớp học phần
ClassroomID	varchar	Phòng học
DayOfWeek	int	Thứ học
StartLesson	int	Tiết bắt đầu

Bảng Timetable được sử dụng để lưu trữ kết quả của quá trình sắp xếp thời khóa biểu. Mỗi bản ghi trong bảng tương ứng với một lớp học phần đã được phân công ngày học, tiết học và phòng học cụ thể. Dữ liệu trong bảng này được sử dụng để hiển thị thời khóa biểu cho cán bộ quản lý và giảng viên sau khi quá trình xếp lịch hoàn thành. Ngoài ra, đây cũng là cơ sở để thực hiện các chức năng thống kê, tra cứu và theo dõi lịch giảng dạy trong hệ thống.

2.7.8 Thiết kế bảng Account, Role và AccountRole

Bảng 2.16. Thiết kế các bảng Account, Role và AccountRole

Account		
Trường	Kiểu dữ liệu	Mô tả
AccountID	int	Mã tài khoản
Username	varchar(100)	Tên đăng nhập
MicrosoftEmail	varchar(100)	Email Microsoft
IsActive	bool	Trạng thái hoạt động

Role		
Trường	Kiểu dữ liệu	Mô tả
RoleID	int	Mã vai trò
RoleName	varchar(100)	Tên vai trò
Description	varchar(100)	Email Microsoft

Account_Role		
Trường	Kiểu dữ liệu	Mô tả
AccountID	int	Mã vai trò
RoleID	varchar(100)	Tên vai trò

Ba bảng Account, Role và AccountRole được sử dụng để xây dựng cơ chế xác thực và phân quyền người dùng trong hệ thống. Bảng Account lưu trữ thông tin tài khoản đăng nhập của người dùng; bảng Role lưu trữ các vai trò được định nghĩa trong hệ thống như Quản trị viên, Cán bộ quản lý và Giảng viên; trong khi đó bảng AccountRole đóng vai trò liên kết giữa tài khoản và vai trò. Mô hình này cho phép một tài khoản có thể được gán nhiều vai trò khác nhau, giúp hệ thống linh hoạt hơn trong việc quản lý quyền truy cập và sử dụng chức năng.

2.7.9 Thiết kế bảng SystemConfig

Bảng 2.17. Thiết kế bảng SystemConfig

Trường	Kiểu dữ liệu	Mô tả
ConfigKey	varchar(100)	Tên tham số cấu hình
ConfigValue	float	Giá trị tham số cấu hình
Description	nvarchar(200)	Mô tả tham số cấu hình

Bảng SystemConfig được sử dụng để lưu trữ các tham số cấu hình phục vụ quá trình vận hành hệ thống và thực hiện thuật toán sắp xếp thời khóa biểu. Các tham số được lưu dưới dạng cặp khóa – giá trị, cho phép người quản trị dễ dàng thay đổi cấu hình mà không cần chỉnh sửa mã nguồn chương trình. Việc tách riêng các tham số cấu hình thành một bảng độc lập giúp hệ thống dễ bảo trì, linh hoạt hơn trong quá trình sử dụng và thuận tiện khi điều chỉnh thuật toán cho các học kỳ khác nhau.

Chương này đã trình bày kết quả phân tích nghiệp vụ, xác định các chức năng chính của hệ thống thông qua các biểu đồ Use Case và quy trình nghiệp vụ. Bên cạnh đó, chương cũng mô tả chi tiết các chức năng quan trọng thông qua các bảng đặc tả Use Case và trình bày thiết kế cơ sở dữ liệu của hệ thống. Đây là cơ sở để triển khai các chức năng phần mềm và thuật toán sắp xếp thời khóa biểu được trình bày trong chương tiếp theo.

CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

Chương này trình bày quá trình triển khai hệ thống DUT Smart Schedule, các chức năng đã xây dựng, mô hình thuật toán sắp xếp thời khóa biểu và kết quả thực nghiệm trên dữ liệu thực tế của Trường Đại học Bách khoa - Đại học Đà Nẵng. Nội dung chương gồm bốn phần chính: mô tả các chức năng đã đạt được, trình bày thuật toán xếp thời khóa biểu, đánh giá kết quả thực nghiệm và tổng kết các kết quả sau triển khai.

3.1 Mô tả chức năng đã đạt được

Sau quá trình nghiên cứu và phát triển, hệ thống DUT Smart Schedule đã xây dựng được các nhóm chức năng chính bao gồm:

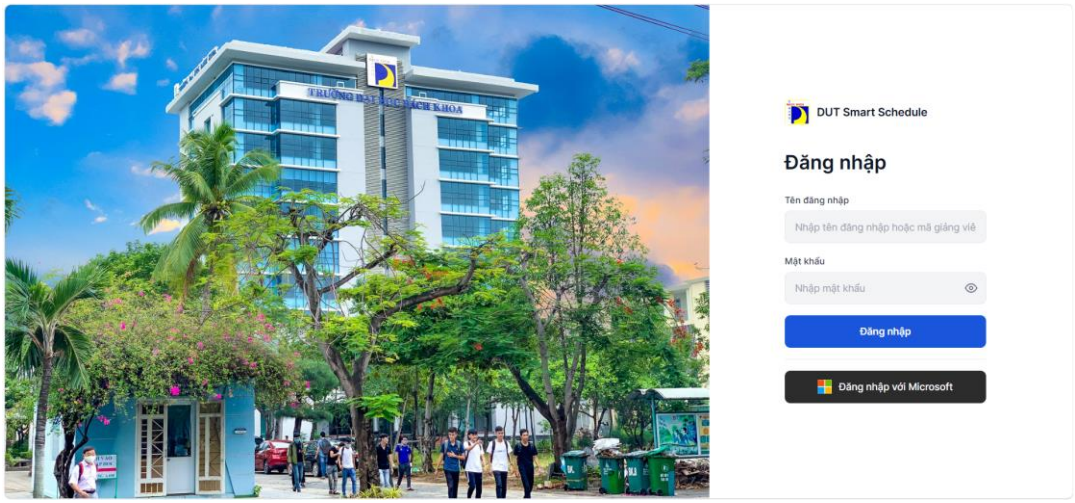
- Quản lý tài khoản và phân quyền.
- Quản lý giảng viên.
- Quản lý phòng học.
- Quản lý lớp học phần.
- Dashboard thống kê dữ liệu đào tạo.
- Cấu hình hệ thống.
- Xếp thời khóa biểu học kỳ.
- Quản lý kết quả xếp lịch.
- Giảng viên xem thời khóa biểu cá nhân.

3.1.1 Chức năng đăng nhập

Hệ thống hỗ trợ hai hình thức đăng nhập:

- Đăng nhập bằng tài khoản nội bộ.
- Đăng nhập bằng tài khoản Microsoft.

Sau khi xác thực thành công, hệ thống cấp JWT và phân quyền người dùng theo vai trò tương ứng. Đối với tài khoản có nhiều vai trò, hệ thống cho phép người dùng lựa chọn vai trò trước khi truy cập vào giao diện chức năng. Cơ chế này giúp bảo đảm mỗi người dùng chỉ được sử dụng các chức năng phù hợp với quyền hạn của mình.



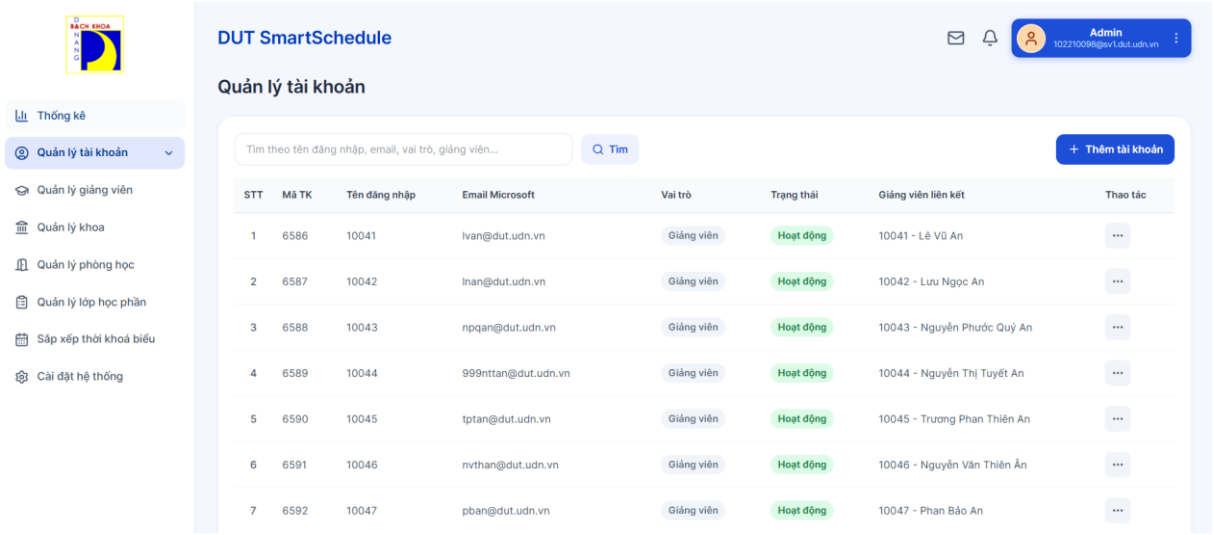
Hình 3.1. Giao diện đăng nhập hệ thống

3.1.2 Chức năng quản lý tài khoản

Chức năng quản lý tài khoản cho phép Quản trị viên thực hiện các thao tác sau:

- Xem danh sách tài khoản.
- Tạo tài khoản mới.
- Chỉnh sửa thông tin tài khoản.
- Khóa hoặc mở khóa tài khoản.
- Gán vai trò cho người dùng.
- Liên kết tài khoản với giảng viên.
- Hỗ trợ thông tin đăng nhập bằng tài khoản Microsoft.

Chức năng này giúp quản lý tập trung người dùng trong hệ thống, bảo đảm an toàn truy cập và hỗ trợ phân quyền theo từng nhóm người dùng.



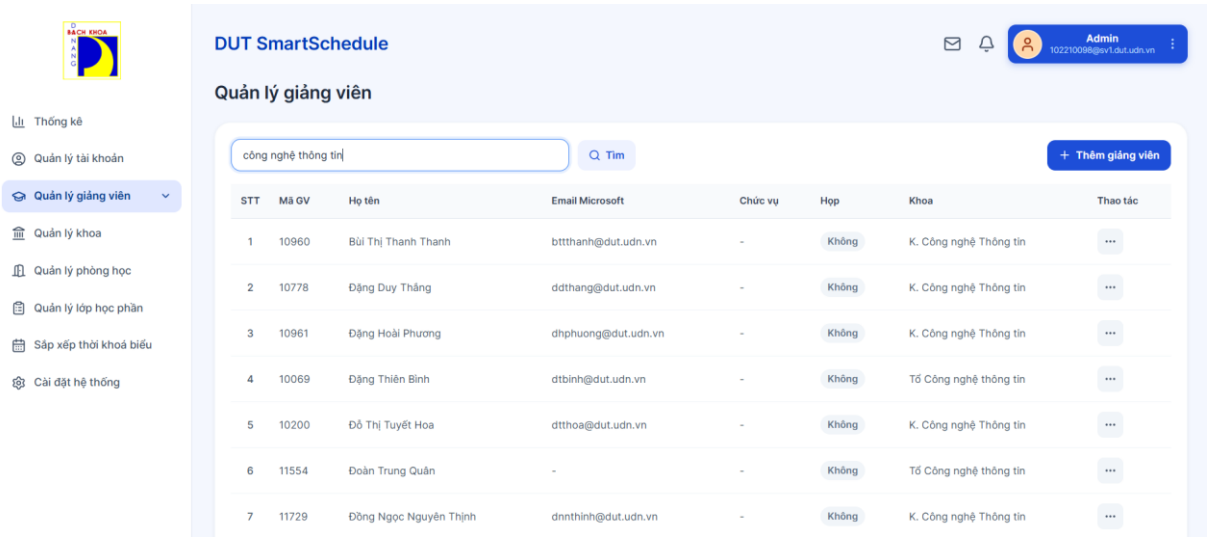
Hình 3.2. Giao diện quản lý tài khoản

3.1.3 Chức năng quản lý giảng viên

Hệ thống hỗ trợ quản lý tập trung dữ liệu giảng viên. Các chức năng chính gồm:

- Xem danh sách giảng viên.
- Thêm mới giảng viên.
- Cập nhật thông tin giảng viên.
- Xóa giảng viên.
- Tìm kiếm giảng viên.
- Liên kết tài khoản Microsoft.

Dữ liệu giảng viên là một trong những dữ liệu đầu vào quan trọng của thuật toán xếp thời khóa biểu. Thông qua thông tin giảng viên phụ trách từng lớp học phần, hệ thống có thể kiểm tra các ràng buộc như giảng viên không được dạy hai lớp cùng thời điểm và tổng số tiết dạy của giảng viên trong một buổi hoặc một ngày không vượt quá giới hạn cho phép.



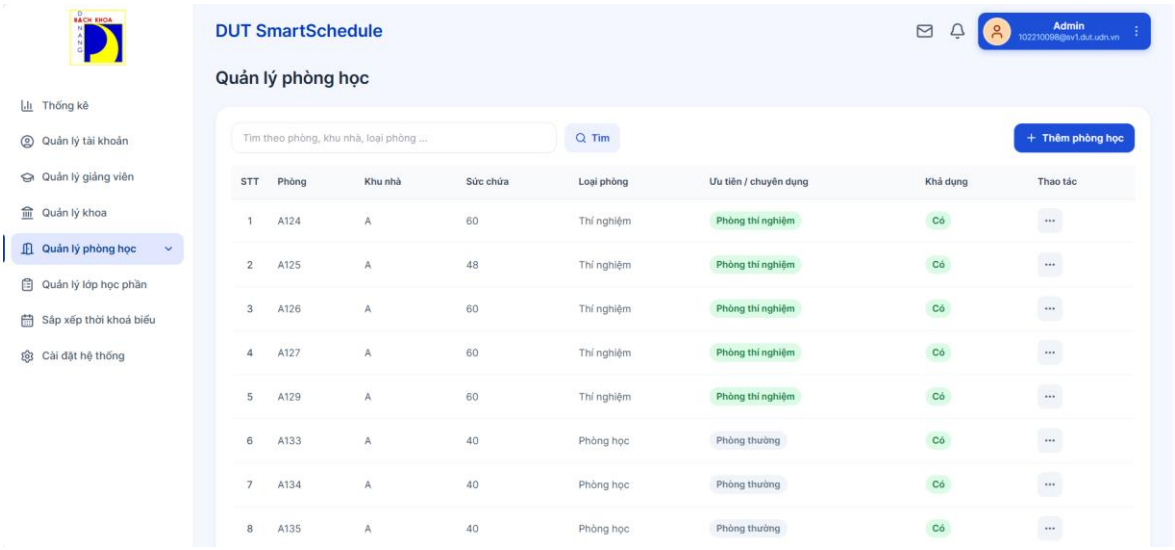
Hình 3.3. Giao diện quản lý giảng viên

3.1.4 Chức năng quản lý phòng học

Chức năng quản lý phòng học cho phép người quản lý:

- Quản lý danh sách phòng học.
- Quản lý sức chứa phòng.
- Quản lý loại phòng.
- Quản lý khu nhà, tầng.
- Cập nhật trạng thái phòng học.
- Phân loại phòng học thông thường, phòng PFIEV, phòng Ngoại ngữ, phòng thực hành, phòng thí nghiệm hoặc phòng chuyên dụng.

Thông tin phòng học được sử dụng trực tiếp trong thuật toán xếp lịch để kiểm tra sức chứa, tránh trùng phòng và bảo đảm lớp học phần được bố trí đúng loại phòng.



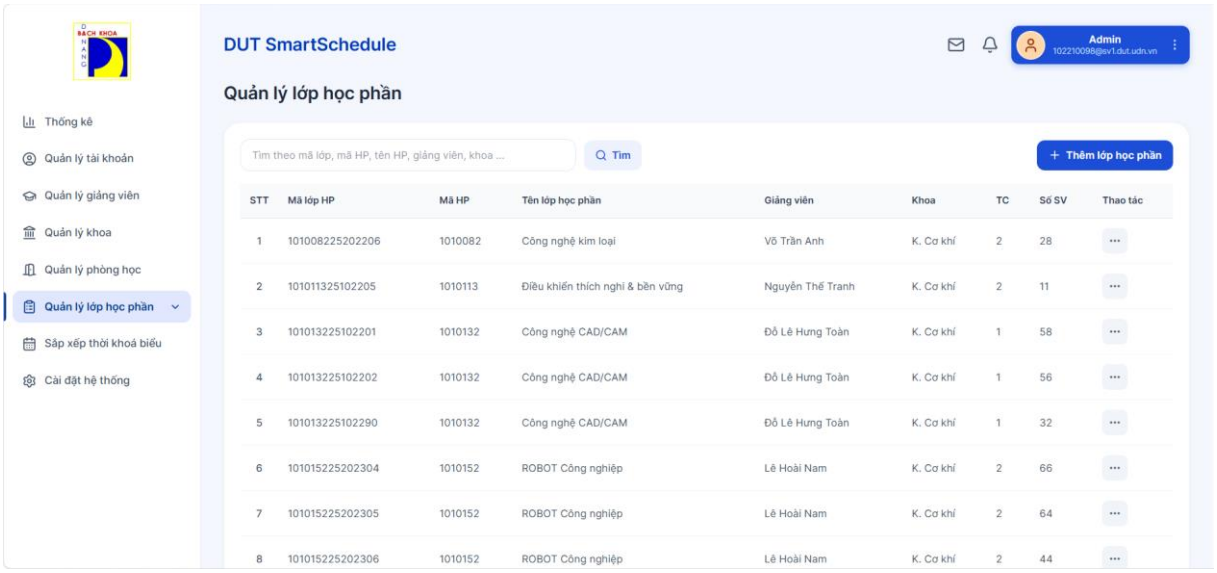
Hình 3.4. Giao diện quản lý phòng học

3.1.5 Chức năng quản lý lớp học phần

Lớp học phần là dữ liệu đầu vào của thuật toán xếp lịch. Hệ thống hỗ trợ:

- Thêm mới lớp học phần.
- Chỉnh sửa lớp học phần.
- Xóa lớp học phần.
- Tìm kiếm lớp học phần.
- Nhập dữ liệu từ Excel.
- Quản lý thông tin giảng viên phụ trách.
- Quản lý sĩ số sinh viên.
- Phân loại lớp học phần như lớp lý thuyết, lớp thực hành, lớp thí nghiệm, lớp Ngoại ngữ hoặc lớp PFIEV.

Việc quản lý đầy đủ thông tin lớp học phần giúp thuật toán có đủ dữ liệu để kiểm tra các ràng buộc như số tiết, loại phòng, sức chứa và khu vực phòng học ưu tiên.



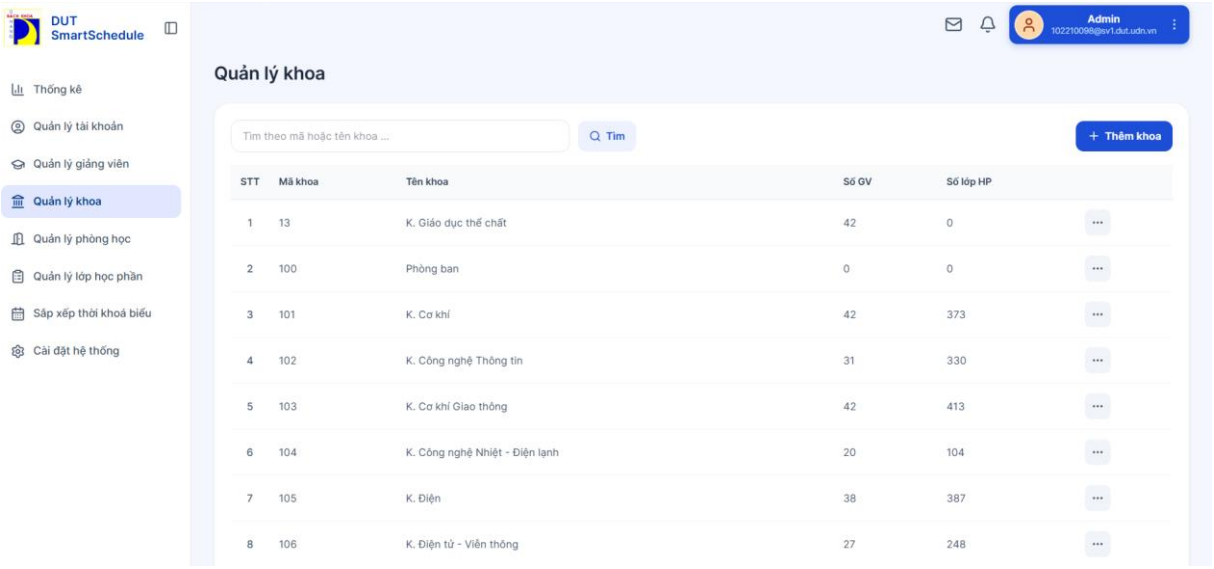
Hình 3.5. Giao diện quản lý lớp học phần

3.1.6 Chức năng quản lý khoa

Chức năng quản lý khoa cho phép người quản lý:

- Quản lý mã khoa.
- Quản lý tên khoa.
- Quản lý số giảng viên trong khoa.
- Quản lý số lớp học phần có trong khoa.

Thông tin khoa được sử dụng trực tiếp trong thuật toán xếp lịch để kiểm tra các ràng buộc về việc phân chia lớp học phần theo khoa



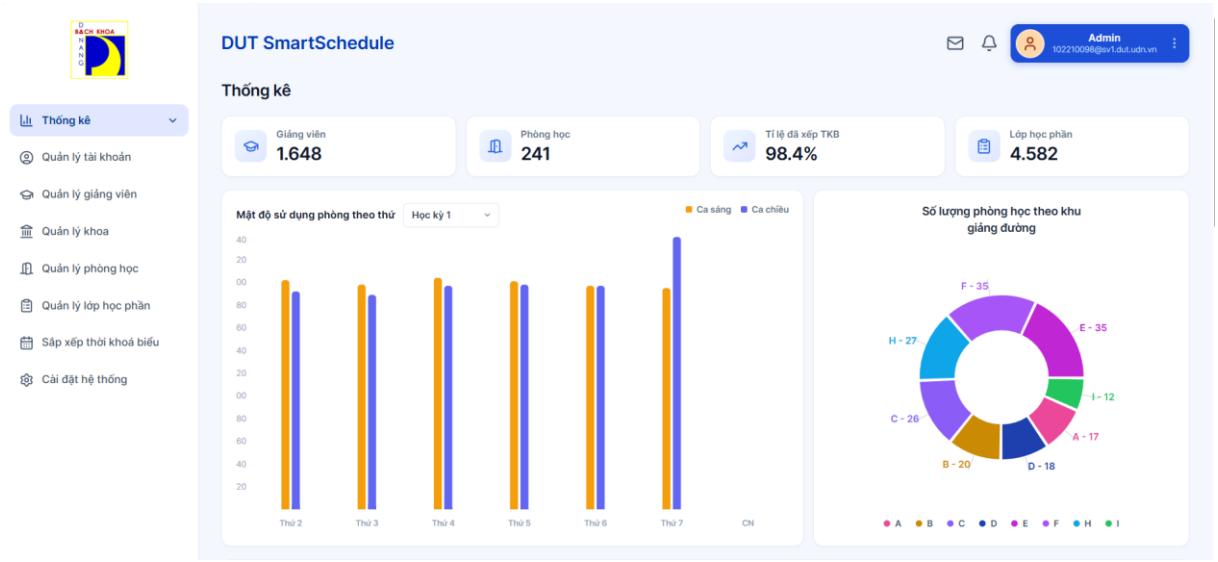
Hình 3.6. Giao diện quản lý khoa

3.1.7 Dashboard thống kê

Dashboard cung cấp các thống kê tổng quan như:

- Tổng số giảng viên.
- Tổng số phòng học.
- Tổng số lớp học phần.
- Thống kê tình trạng xếp lịch.
- Số lớp chưa xếp lịch.

Các dữ liệu này giúp người quản lý nhanh chóng nắm bắt tình hình vận hành của hệ thống, đồng thời theo dõi tiến độ xây dựng thời khóa biểu theo từng học kỳ.



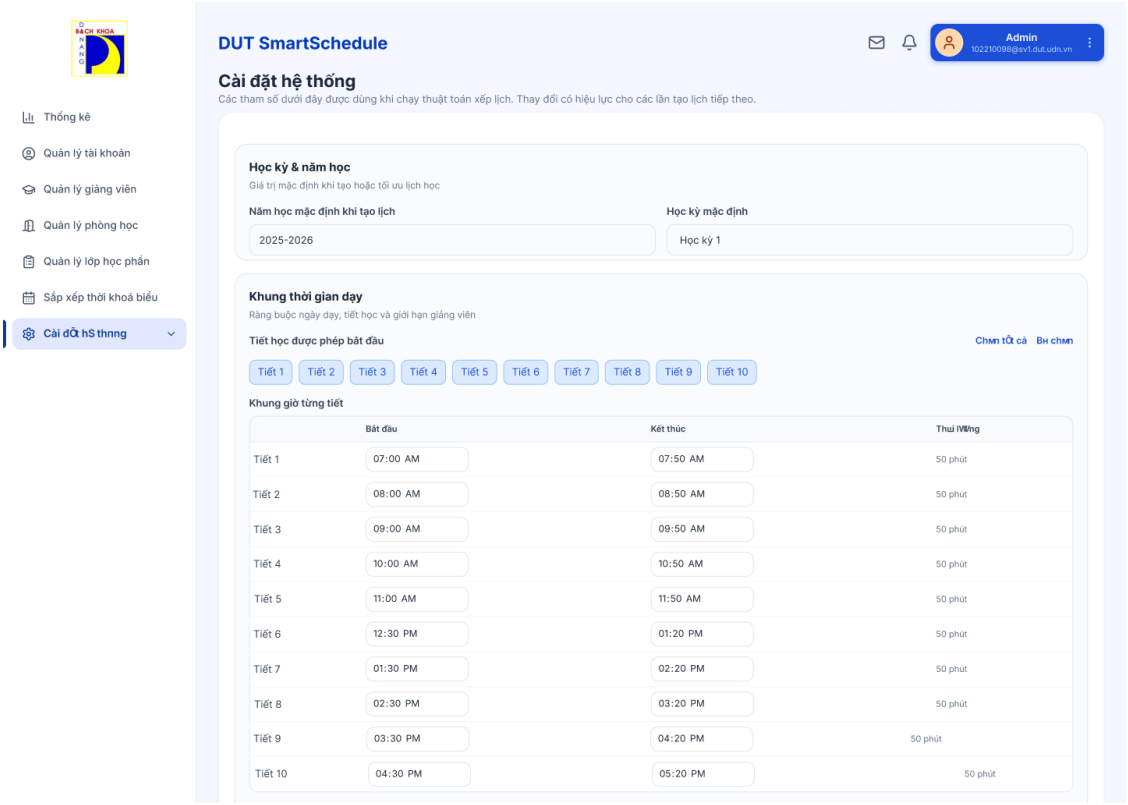
Hình 3.7. Dashboard thống kê dữ liệu

3.1.8 Chức năng cấu hình hệ thống

Hệ thống cho phép thay đổi các tham số phục vụ thuật toán xếp lịch như:

- Năm học.
- Học kỳ.
- Kích thước batch.
- Thời gian chạy tối đa.
- Các trọng số phạt.
- Cấu hình khung thời gian.
- Cấu hình giới hạn số tiết.

Việc cấu hình linh hoạt giúp hệ thống dễ dàng thích nghi với các học kỳ khác nhau và cho phép người quản lý điều chỉnh chiến lược tối ưu hóa theo nhu cầu thực tế.



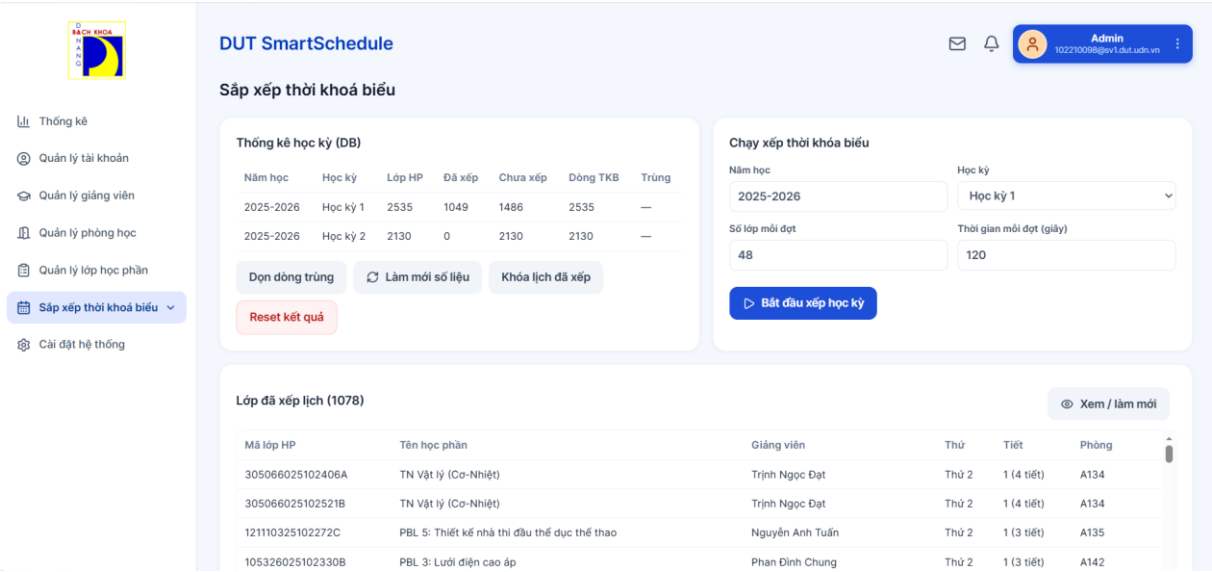
Hình 3.8. Giao diện cấu hình hệ thống

3.1.9 *Chức năng xếp thời khóa biểu*

Đây là chức năng quan trọng nhất của hệ thống. Người dùng có quyền có thể:

- Chọn học kỳ.
- Khởi chạy thuật toán.
- Theo dõi tiến trình xử lý.
- Xem kết quả xếp lịch.
- Xem danh sách lớp chưa xếp được.
- Reset kết quả xếp lịch nếu cần chạy lại.
- Khóa lịch đã duyệt.

Chức năng xếp thời khóa biểu sử dụng bộ giải CP-SAT để tìm phương án phân công ngày học, tiết học và phòng học cho các lớp học phần. Kết quả được lưu vào cơ sở dữ liệu để phục vụ tra cứu và thống kê.



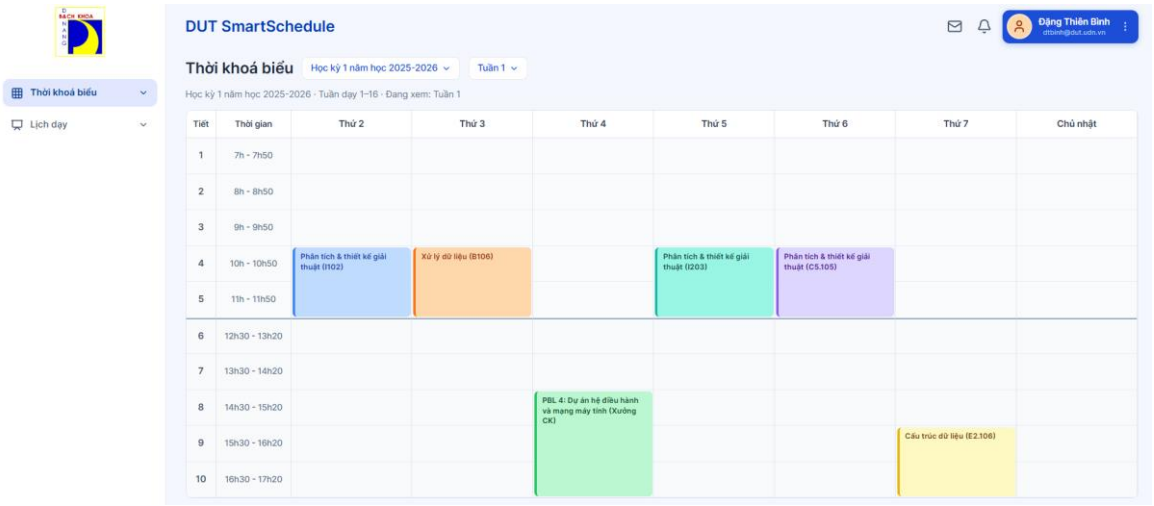
Hình 3.9. Giao diện xếp thời khóa biểu học kỳ

3.1.10 Chức năng giảng viên xem lịch dạy

Sau khi thời khóa biểu được sắp xếp hoàn tất, giảng viên có thể tra cứu lịch giảng dạy theo:

- Học kỳ.
- Tuần học.
- Danh sách buổi dạy.
- Lưới thời khóa biểu theo ngày và tiết học.

Hệ thống hiển thị trực quan thời gian học, phòng học và lớp học phần tương ứng. Chức năng này giúp giảng viên chủ động theo dõi lịch giảng dạy và giảm phụ thuộc vào các bảng lịch thủ công.



Hình 3.10. Giao diện giảng viên xem lịch dạy

3.2 Thuật toán xếp thời khóa biểu

Thuật toán xếp thời khóa biểu là thành phần trung tâm của hệ thống DUT Smart Schedule. Trong phạm vi đề tài, bài toán được mô hình hóa theo hướng tối ưu hóa ràng buộc, sử dụng Google OR-Tools với bộ giải CP-SAT. Mục tiêu của thuật toán là tìm phương án phân công thời gian học và phòng học cho các lớp học phần sao cho thỏa mãn các ràng buộc cứng và tối ưu các ràng buộc mềm.

3.2.1 Quy trình triển khai thuật toán xếp thời khóa biểu

Quá trình xếp thời khóa biểu trong hệ thống được triển khai theo các bước chính sau:

Bước 1. Thu thập dữ liệu đầu vào

Hệ thống lấy dữ liệu từ cơ sở dữ liệu, bao gồm danh sách lớp học phần cần xếp lịch, danh sách giảng viên, danh sách phòng học, danh sách khung thời gian, thông tin học kỳ và các tham số cấu hình thuật toán. Trước khi xây dựng mô hình tối ưu, hệ thống cần kiểm tra tính đầy đủ và hợp lệ của dữ liệu.

Bước 2. Tiền xử lý dữ liệu

Dữ liệu đầu vào được chuẩn hóa để phục vụ thuật toán. Các lớp học phần được chuyển thành tập các đối tượng cần xếp lịch; phòng học được phân loại theo sức chứa, khu vực và loại phòng; khung thời gian được xác định theo buổi sáng và buổi chiều. Những phương án gán không hợp lệ, ví dụ lớp thực hành vào phòng lý thuyết hoặc lớp có sĩ số vượt quá sức chứa phòng, có thể được loại bỏ trước để giảm không gian tìm kiếm.

Bước 3. Xây dựng biến quyết định

Hệ thống tạo biến quyết định cho từng khả năng gán lớp học phần vào phòng học và khung thời gian. Biến quyết định có giá trị nhị phân, thể hiện việc phương án gán đó có được lựa chọn hay không.

Bước 4. Xây dựng các ràng buộc cứng

Hệ thống thêm các ràng buộc bắt buộc như mỗi lớp học phần chỉ được xếp đúng một buổi học, giảng viên không bị trùng lịch, phòng học không bị trùng lịch, phòng học đủ sức chứa, lớp PFIEV và lớp Ngoại ngữ được xếp đúng nhóm phòng được phép, phòng chuyên dụng chỉ được dùng cho lớp có yêu cầu tương ứng.

Bước 5. Xây dựng các ràng buộc mềm và hàm mục tiêu

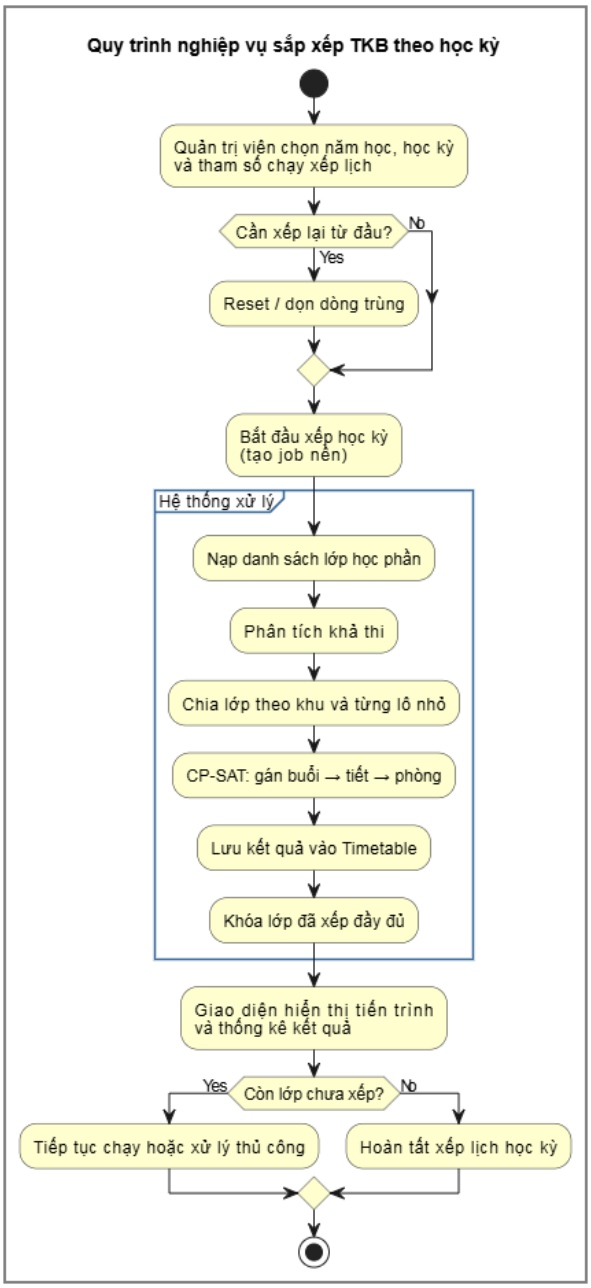
Các tiêu chí tối ưu được chuyển thành các thành phần chi phí phạt. Hệ thống tính chi phí phạt cho từng phương án dựa trên các tiêu chí như học vào thứ bảy, số ngày giảng viên phải đến trường, khoảng trống trong lịch giảng dạy, mức chênh lệch giữa sức chứa phòng và sĩ số lớp, ưu tiên lớp PFIEV vào khu P và ưu tiên lớp Ngoại ngữ vào khu B tầng 1, 2.

Bước 6. Gọi bộ giải CP-SAT

Sau khi mô hình được xây dựng, hệ thống gọi bộ giải CP-SAT để tìm lời giải. Bộ giải sẽ tìm phương án thỏa mãn toàn bộ ràng buộc cứng và tối thiểu hóa tổng chi phí phạt của các ràng buộc mềm. Người quản trị có thể cấu hình giới hạn thời gian chạy để kiểm soát hiệu năng hệ thống.

Bước 7. Lưu và hiển thị kết quả

Nếu tìm được lời giải, hệ thống chuyển kết quả từ mô hình thuật toán về dạng dữ liệu thời khóa biểu và lưu vào cơ sở dữ liệu. Người dùng có thể xem kết quả theo lớp học phần, giảng viên, phòng học hoặc học kỳ. Nếu không tìm được lời giải, hệ thống hiển thị thông báo và gợi ý kiểm tra lại dữ liệu hoặc ràng buộc.



Hình 3.11. Quy trình sắp xếp TKB theo học kỳ

3.2.2 Kiến trúc phân rã bài toán

Khi kích thước dữ liệu học kỳ tăng lên tới hàng ngàn lớp học phần, việc giải đồng thời (Combined Mode) phân bổ cả ca học và phòng học cùng lúc sẽ khiến không gian tìm kiếm bị bùng nổ tổ hợp, dẫn đến hiện tượng quá thời gian (Solver Timeout). Hệ thống đã áp dụng chiến lược giải hai giai đoạn tuần tự:

1. Giai đoạn 1: Bộ giải tập trung tìm kiếm phương án phân bổ khung thời gian (Thứ, Tiết học) tối ưu cho toàn bộ danh sách lớp học phần dựa trên việc phân chia quỹ thời gian tính toán thích hợp
2. Giai đoạn 2: Giữ cố định khung thời gian đã tìm được ở Giai đoạn 1, bộ giải tiến hành khớp nối dán nhãn phòng học phù hợp dựa trên sức chứa và khu vực ưu tiên.

3.2.3 Chiến lược nới lỏng và thử lại

Nếu xuất hiện các lớp học phần không thể tìm được lời giải hợp lệ do xung đột tài nguyên cục bộ (thiếu phòng sĩ số lớn, giảng viên quá tải lịch), hệ thống kích hoạt vòng lặp hồi quy tự động gồm 3 cấp độ (Relaxation Levels từ 1 đến 3):

- Cấp độ 1: Giảm bớt các trọng số phạt mềm, chia nhỏ tệp dữ liệu lỗi thành các lô nhỏ để giải độc lập.
- Cấp độ 2: Cho phép nới lỏng biên độ chênh lệch sức chứa phòng, chấp nhận phân bổ phòng có diện tích lãng phí hơn tiêu chuẩn ban đầu.
- Cấp độ 3: Tính toán hợp nhất các trọng số phạt mềm đã được nới lỏng, gia tăng giới hạn thời gian chạy nhằm quét triệt để không gian tìm kiếm còn sót lại.

3.2.4 Mô hình bài toán

Bài toán lập thời khóa biểu trong hệ thống DUT Smart Schedule được mô hình hóa dưới dạng bài toán tối ưu hóa ràng buộc (Constraint Optimization Problem). Mục tiêu của bài toán là xác định đồng thời thời gian học và phòng học cho toàn bộ các lớp học phần trong học kỳ sao cho:

- Không vi phạm các ràng buộc bắt buộc của nhà trường.
- Chất lượng thời khóa biểu đạt mức tối ưu theo các tiêu chí đã định nghĩa.

Cho trước các tập hợp và các chỉ số sau:

- C: Tập hợp các lớp học phần cần xếp lịch (chỉ số $c \in C$).
- R: Tập hợp các phòng học khả dụng (chỉ số $r \in R$).
- D: Tập hợp các ngày học trong tuần từ Thứ 2 đến Thứ 7 ($d \in D$).
- P: Tập hợp các tiết học trong một ngày ($p \in P$).
- L: Tập hợp giảng viên ($l \in L$).

- W: Tập hợp các tuần học trong học kỳ (w ∈ W).

Mỗi lớp học phần cần gán đúng một tổ hợp (R, T) phù hợp

Không gian tìm kiếm của bài toán là |R|×|T| đối với mỗi lớp học phần

Do có hàng nghìn lớp học phần nên tổng số khả năng phân công tăng theo cấp số mũ $(|R|\times|T|)^{|C|}$ đây là nguyên nhân khiến bài toán thuộc nhóm NP-Hard

3.2.5 Biến quyết định

Hệ thống sử dụng một biến logic nhị phân kiểm soát toàn bộ trạng thái phân bổ:

$$x_{c,r,d,p} \in \{0,1\} \quad \forall c \in C, r \in R, d \in D, p \in P$$

- $x_{c,r,d,p} = 1$: Lớp học phần c được phân bổ bắt đầu tại phòng r, vào ngày thứ d, bắt đầu từ tiết p.
- $x_{c,r,d,p} = 0$: Ngược lại.

Dựa trên biến quyết định này, hệ thống có thể xây dựng các ràng buộc để kiểm tra một lớp học phần được xếp đúng một buổi, một phòng học không bị trùng, một giảng viên không bị trùng lịch và phòng học đáp ứng yêu cầu của lớp học phần.

Ngoài biến quyết định chính, hệ thống có thể sử dụng các biến phụ để phục vụ tối ưu hóa ràng buộc mềm, ví dụ:

- Biến xác định lớp học phần có được xếp vào thứ bảy hay không.
- Biến xác định giảng viên có phải lên trường trong một ngày cụ thể hay không.
- Biến xác định khoảng trống trong lịch dạy của giảng viên.
- Biến xác định giảng viên có dạy kín cả ngày hay không.
- Biến xác định mức chênh lệch giữa sức chứa phòng học và sĩ số lớp học phần.
- Biến xác định lớp PFIEV hoặc Ngoại ngữ có được xếp đúng khu vực ưu tiên hay không.

Các biến phụ không làm thay đổi tính hợp lệ của thời khóa biểu mà chỉ được sử dụng trong hàm mục tiêu nhằm nâng cao chất lượng lời giải.

3.2.6 Mô hình hóa ràng buộc cứng trong hệ thống

Ràng buộc cứng là các điều kiện bắt buộc phải được thỏa mãn trong quá trình xây dựng thời khóa biểu. Nếu một phương án vi phạm ràng buộc cứng thì phương án đó không được xem là hợp lệ và sẽ bị loại khỏi không gian lời giải.

HC1: Mỗi lớp học phần được xếp tối đa một lần trong học kỳ

Một lớp học phần c sau khi phân bổ không thể xuất hiện ở hai thời điểm hoặc hai phòng khác nhau:

$$\sum_{r\in R}\sum_{d\in D}\sum_{p\in P_c^{allow}} x_{c,r,d,p} \le 1 \quad \forall c \in C$$

(Trong đó P_c^{allow} là tập hợp các tiết bắt đầu hợp lệ được cấu hình cho học phần).

HC2: Ràng buộc trùng lịch phòng học

Hai lớp học phần c_1 và c_2 có trùng nhau về tuần học ($W_{c_1} \cap W_{c_2} \neq \emptyset$) không được phép xếp chung một phòng r, chung một ngày d tại các tiết học chồng lấn nhau:

$$x_{c_1,r,d,p_1} + x_{c_2,r,d,p_2} \leq 1$$

HC3: Ràng buộc trùng lịch giảng viên (Lecturer Conflict)

Giảng viên l không thể dạy hai lớp học phần độc lập cùng một lúc trong cùng một tuần học, trừ trường hợp đặc biệt được định nghĩa là các nhóm lớp thực hành song song:

$$\sum_{r \in R} \sum_{c \in C_l} x_{c,r,d,p'} \leq 1 \quad \forall d \in D, \forall p \in P, \forall w \in W$$

HC4: Giới hạn năng suất giảng dạy tối đa của Giảng viên trong ngày

Để đảm bảo sức khỏe và quy định sư phạm, tổng số tiết dạy của giảng viên l trong ngày d ở bất kỳ tuần nào không được vượt quá giới hạn (mặc định là 9 tiết):

$$\sum_{c \in C_l} \sum_{r \in R} \sum_{p \in P} x_{c,r,d,p} \times len(c) \leq MaxLecturerLessonsPerDay$$

HC05. Các tiết của cùng một lớp học phần phải liên tục

Khi một lớp học phần có thời lượng nhiều tiết trong một buổi, các tiết học của lớp đó phải được bố trí liên tiếp nhau. Hệ thống chỉ tạo các khung thời gian hợp lệ tương ứng với các đoạn tiết liên tục.

Ví dụ, lớp học 3 tiết có thể được xếp vào tiết 1–3, 2–4, 3–5, 6–8, 7–9 hoặc 8–10. Mọi biến $x_{c,r,t}$ được sinh ra mặc định đã thỏa mãn tính liên tục của buổi học. Hệ thống không tạo các phương án rời rạc như tiết 1, tiết 3 và tiết 5.

HC06. Mỗi lớp học phần chỉ được gán đúng một phòng học và một khung thời gian

Mỗi lớp học phần bắt buộc phải được xếp lịch và chỉ được chọn duy nhất một phương án tổ hợp (phòng học, khung thời gian) trong tập phương án hợp lệ S_c.

Công thức toán học:

$$\sum_{(r,t)\in S_c} x_{c,r,t} = 1 \quad \forall c \in C$$

Ràng buộc này đảm bảo không có lớp nào bị bỏ sót, đồng thời không bị chia nhỏ sang nhiều phòng khác nhau hay nhiều khung giờ khác nhau trong cùng một buổi.

HC07. Sức chứa phòng học phải đủ cho lớp

Trước khi tạo biến quyết định, hệ thống kiểm tra sức chứa phòng học. Biến $x_{c,r,t}$ chỉ được khởi tạo nếu sức chứa của phòng r đáp ứng sĩ số của lớp c:

$$Capacity(r) \geq StudentCount(c)$$

HC08. Không được trùng phòng cùng thời điểm

Tại bất kỳ khung thời gian t nào, một phòng học r chỉ có thể phục vụ tối đa một lớp học phần.

Công thức toán học:

$$\sum_{c\in C} x_{c,r,t} \leq 1 \quad \forall r \in R, \forall t \in T$$

Ràng buộc này chống xung đột tài nguyên cốt lõi, bảo đảm không có hiện tượng "đụng phòng" trong thời khóa biểu.

HC09. Lớp PFIEV phải học trong các phòng PFIEV được phép

Đối với các lớp thuộc chương trình PFIEV, hệ thống chỉ cho phép tạo biến quyết định với các phòng thuộc danh sách phòng PFIEV được phép. Các phòng không thuộc danh sách này sẽ không được xét là phương án hợp lệ.

Hay nói cách khác nếu lớp $c \in C_{PFIEV}$ (tập các lớp chương trình PFIEV), thì biến chỉ được sinh ra nếu $r \in R_{PFIEV}$ (tập các phòng chuẩn PFIEV).

HC10. Lớp Ngoại ngữ phải học trong các phòng Ngoại ngữ được phép

Tương tự lớp PFIEV, các lớp Ngoại ngữ chỉ được xếp vào nhóm phòng Ngoại ngữ được phép. Hệ thống loại bỏ các phương án gán lớp Ngoại ngữ vào phòng không phù hợp.

Nếu $c \in C_{NgoaiNgu}$, thì r bắt buộc phải thuộc $R_{NgoaiNgu}$.

HC11. Phòng chuyên dụng chỉ được dùng cho lớp có yêu cầu chuyên dụng tương ứng

Hệ thống kiểm tra loại lớp học phần và loại phòng học trước khi tạo biến quyết định. Các lớp thí nghiệm chỉ được xếp vào phòng thí nghiệm phù hợp. Các lớp thực hành chỉ được xếp vào phòng thực hành hoặc xưởng phù hợp. Các lớp lý thuyết thông thường không được xếp vào phòng thí nghiệm, phòng thực hành, xưởng hoặc các phòng chuyên dụng khác.

- Nếu c là lớp thực hành/thí nghiệm/xưởng → r phải là phòng thực hành/thí nghiệm tương ứng.
- Nếu c là lớp lý thuyết → r không được là phòng chuyên dụng.

Ràng buộc này được triển khai bằng cách lọc danh sách phòng hợp lệ cho từng lớp học phần trong giai đoạn tiền xử lý dữ liệu.

3.2.7 Mô hình hoá các ràng buộc mềm trong hệ thống

Bên cạnh các ràng buộc cứng, hệ thống còn sử dụng các ràng buộc mềm nhằm nâng cao chất lượng của thời khóa biểu. Các ràng buộc này không bắt buộc phải thỏa mãn hoàn toàn nhưng được đưa vào hàm mục tiêu để tối ưu, khi biến vi phạm mỗi ràng buộc mềm sẽ sinh ra một giá trị phạt

$Penalty=w_1P_1+w_2P_2+...+w_7P_7$

trong đó w là trọng số của từng ràng buộc mềm và P giá trị phạt tương ứng của từng ràng buộc mềm

SC01. Hạn chế học vào thứ bảy

Đối với mỗi lớp học phần được xếp vào thứ bảy, hệ thống cộng thêm một chi phí phạt vào hàm mục tiêu. Nhờ đó, bộ giải sẽ ưu tiên xếp lớp vào các ngày từ thứ hai đến thứ sáu nếu vẫn còn đủ tài nguyên phòng học và khung thời gian.

Tiêu chí này không cấm hoàn toàn việc học vào thứ bảy, vì trong một số trường hợp số lượng lớp học phần lớn hoặc số lượng phòng học hạn chế, hệ thống vẫn cần sử dụng thứ bảy để tìm được phương án hợp lệ.

SC02. Giảm số ngày lên trường của giảng viên

Hệ thống xác định số ngày trong tuần mà mỗi giảng viên có lịch dạy. Mỗi ngày có phát sinh lịch dạy sẽ được tính vào chi phí phạt. Khi tối ưu, bộ giải sẽ ưu tiên gom lịch dạy của giảng viên vào ít ngày hơn, miễn là không vi phạm các ràng buộc cứng như giới hạn số tiết trong buổi và giới hạn số tiết trong ngày.

Tiêu chí này giúp thời khóa biểu thuận tiện hơn cho giảng viên và hỗ trợ giảm thời gian di chuyển.

SC03. Giảm khoảng trống trong lịch dạy của giảng viên

Khoảng trống trong lịch dạy xuất hiện khi giảng viên có các lớp trong cùng một ngày nhưng giữa các lớp có nhiều tiết trống. Hệ thống tính chi phí phạt dựa trên số lượng khoảng trống hoặc tổng số tiết trống trong lịch dạy của giảng viên.

Khi tối ưu, bộ giải sẽ ưu tiên các phương án giúp lịch dạy của giảng viên liền mạch hơn, hạn chế thời gian chờ giữa các lớp.

SC04. Hạn chế giảng viên dạy kín cả ngày

Nếu một giảng viên được xếp lịch cả buổi sáng và buổi chiều trong cùng một ngày, hệ thống có thể cộng thêm chi phí phạt. Tiêu chí này nhằm hạn chế tình trạng giảng viên phải dạy kín cả ngày, đặc biệt trong các trường hợp lịch dạy kéo dài và cường độ cao.

Tiêu chí này là ràng buộc mềm vì trong một số trường hợp, việc dạy cả ngày có thể vẫn cần thiết để bảo đảm xếp được toàn bộ lớp học phần.

SC05. Ưu tiên phòng học phù hợp sĩ số

Đối với mỗi phương án xếp lớp học phần c vào phòng học r, hệ thống tính độ chênh lệch giữa sức chứa phòng và sĩ số lớp dựa trên công thức:

Capacity(r) - StudentCount(c)

Giá trị này càng nhỏ thì phòng học càng phù hợp với lớp học phần. Nếu giá trị chênh lệch lớn, hệ thống cộng thêm chi phí phạt cao hơn. Tiêu chí này giúp hạn chế việc sử dụng phòng học lớn cho lớp có sĩ số nhỏ, từ đó nâng cao hiệu quả sử dụng cơ sở vật chất.

SC06. Ưu tiên lớp PFIEV vào khu P

Đối với các lớp PFIEV, ngoài ràng buộc cứng là phải học trong các phòng PFIEV được phép, hệ thống còn bổ sung tiêu chí mềm là ưu tiên các phòng thuộc khu P. Nếu lớp PFIEV được xếp vào phòng hợp lệ nhưng không thuộc khu P, phương án vẫn hợp lệ nhưng sẽ bị cộng thêm chi phí phạt.

Tiêu chí này giúp hệ thống linh hoạt hơn: vẫn bảo đảm tính đúng đắn bằng ràng buộc cứng, đồng thời tối ưu chất lượng lịch bằng cách ưu tiên khu vực phù hợp.

SC07. Ưu tiên lớp Ngoại ngữ vào khu B tầng 1, 2

Đối với các lớp Ngoại ngữ, hệ thống ưu tiên bố trí vào khu B, tầng 1 hoặc tầng 2. Nếu lớp Ngoại ngữ được xếp vào phòng Ngoại ngữ hợp lệ nhưng không thuộc khu vực ưu tiên này, hệ thống cộng thêm chi phí phạt.

Tiêu chí này giúp đáp ứng tốt hơn yêu cầu tổ chức các lớp Ngoại ngữ và hỗ trợ quản lý phòng học theo khu vực.

Bảng tóm tắt các ràng buộc mềm:

Bảng 3.1: Bảng tóm tắt các ràng buộc mềm

Mã	Tên ràng buộc	Cơ chế phạt	Mục tiêu tối ưu
SC01	Hạn chế học thứ bảy	Phạt mỗi lớp học phần bị xếp lịch vào ngày thứ bảy.	Ưu tiên dồn lịch học từ thứ 2 đến thứ 6; dùng thứ 7 làm phương án dự phòng.
SC02	Giảm số ngày lên trường của GV	Phạt trên mỗi ngày trong tuần có phát sinh lịch dạy của giảng viên.	Gom lịch dạy vào số ngày ít nhất có thể, tối ưu thời gian di chuyển cho giảng viên.

SC03	Giảm khoảng trống lịch dạy	Phạt dựa trên tổng số tiết trống/khoảng trống giữa các ca dạy trong cùng một ngày.	Tạo lịch dạy liền mạch, giảm thời gian chờ đợi giữa các lớp cho giảng viên.
SC04	Hạn chế GV dạy kín cả ngày	Phạt khi giảng viên có cả lịch dạy buổi sáng và buổi chiều trong cùng một ngày.	Giảm áp lực và cường độ làm việc quá tải trong một ngày cho giảng viên.
SC05	Ưu tiên phòng phù hợp sĩ số	Phạt dựa trên độ chênh lệch thừa: $Capacity(r) - StudentCount(c)$.	Tránh lãng phí không gian, không xếp lớp nhỏ vào phòng có sức chứa lớn.
SC06	Ưu tiên lớp PFIEV vào khu P	Phạt nếu lớp PFIEV bị xếp vào phòng hợp lệ nhưng không nằm trong khu P.	Tối ưu chất lượng không gian học, gom các lớp chương trình PFIEV về đúng khu P.
SC07	Ưu tiên lớp Ngoại ngữ (Khu B, tầng 1-2)	Phạt nếu lớp Ngoại ngữ bị xếp vào phòng hợp lệ nhưng không thuộc tầng 1, 2 khu B.	Hỗ trợ vận hành và quản lý cơ sở vật chất đồng bộ theo đặc thù môn học.

3.2.8 Phân tích độ phức tạp

Bài toán lập thời khóa biểu thuộc nhóm NP-Hard.

Giả sử một học kỳ cần sắp xếp thời khoá biểu cho:

- n lớp học phần
- m phòng
- k slot

Không gian tìm kiếm xấp xỉ

$$(mk)^n$$

Nếu sử dụng thuật toán vét cạn, số lượng phương án cần kiểm tra tăng theo cấp số mũ nên không khả thi đối với dữ liệu thực tế.

Google OR-Tools CP-SAT không tìm kiếm toàn bộ không gian lời giải mà áp dụng:

- Constraint Propagation
- Lazy Clause Generation
- Conflict Learning
- Branch and Bound
- Restart Strategy

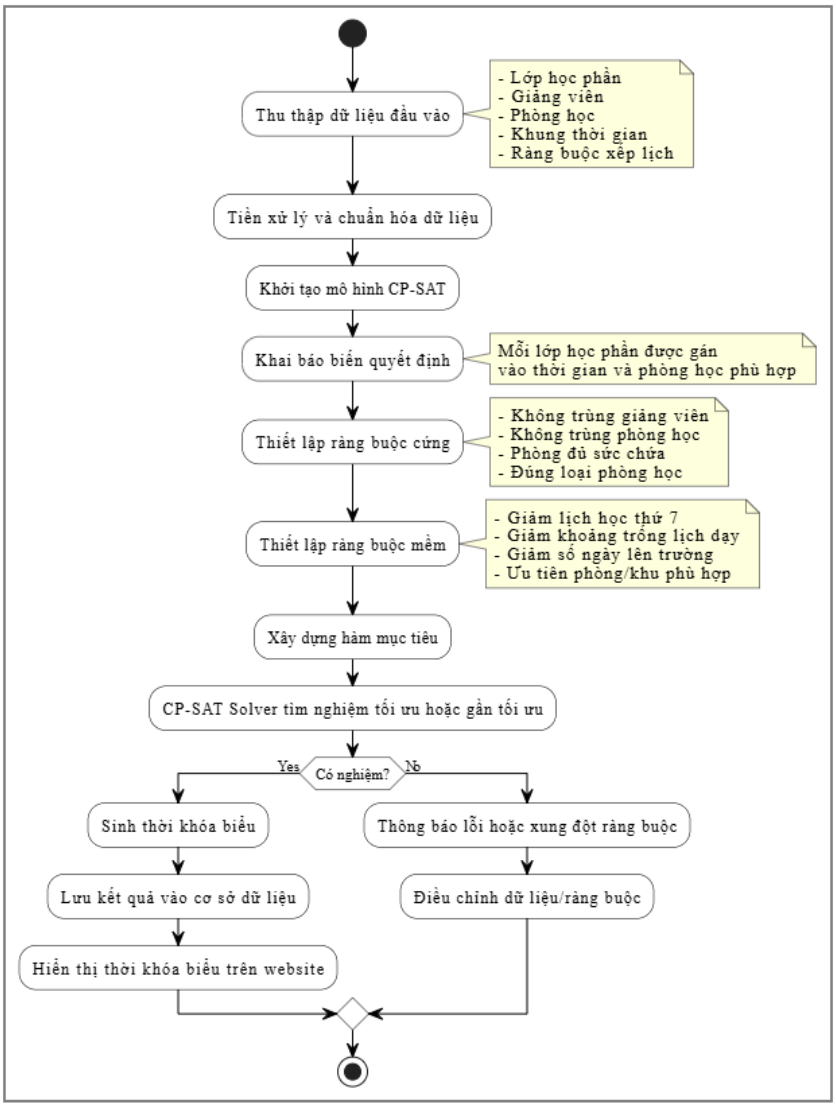
Nhờ đó nhiều nhánh không khả thi được loại bỏ ngay từ đầu.

Trong thực nghiệm của đề tài, thuật toán vẫn sinh được thời khóa biểu cho hàng nghìn lớp học phần trong thời gian chấp nhận được.

3.2.9 Quy trình giải bằng CP-SAT

Sau khi dữ liệu đầu vào được chuẩn bị đầy đủ, hệ thống thực hiện quá trình xây dựng thời khóa biểu theo các bước sau:

1. Thu thập dữ liệu giảng viên, phòng học, lớp học phần và các tham số cấu hình.
2. Tiền xử lý dữ liệu và loại bỏ các phương án gán không hợp lệ.
3. Xây dựng tập biến quyết định.
4. Thiết lập các ràng buộc cứng.
5. Thiết lập các ràng buộc mềm.
6. Xây dựng hàm mục tiêu.
7. Đưa mô hình vào bộ giải CP-SAT.
8. Tìm kiếm lời giải tối ưu hoặc gần tối ưu.
9. Lưu kết quả vào cơ sở dữ liệu thời khóa biểu.
10. Hiển thị kết quả cho người dùng.



Hình 3.12. Quy trình xây dựng thời khóa biểu bằng bộ giải CP-SAT

Việc sử dụng CP-SAT giúp hệ thống xử lý hiệu quả bài toán có số lượng lớn lớp học phần và nhiều ràng buộc phức tạp. Kết quả thực nghiệm cho thấy thuật toán có khả năng tạo ra thời khóa biểu hợp lệ trong thời gian chấp nhận được và đáp ứng tốt các yêu cầu nghiệp vụ của nhà trường.

3.3 Kết quả thực nghiệm

3.3.1 Dữ liệu thực nghiệm

Để đánh giá khả năng hoạt động của hệ thống, đề tài sử dụng dữ liệu thực tế của Trường Đại học Bách khoa – Đại học Đà Nẵng. Dữ liệu được tổng hợp từ các học phần mở trong năm học 2025–2026, bao gồm thông tin về giảng viên, lớp học phần, phòng học và các tham số phục vụ quá trình xây dựng thời khóa biểu.

Dữ liệu đầu vào được chuẩn hóa và lưu trữ trong cơ sở dữ liệu của hệ thống trước khi đưa vào thuật toán xếp lịch. Trong quá trình thực nghiệm, hệ thống tiến hành kiểm tra tính hợp lệ của dữ liệu nhằm loại bỏ các trường hợp thiếu thông tin hoặc không đáp ứng điều kiện đầu vào.

Bảng 3.2. Thông số dữ liệu thực nghiệm

Thành phần dữ liệu	Số lượng
Số khoa/bộ môn	62
Số giảng viên	1.647
Số phòng học	241
Số lớp học phần	4.665
Số lớp lý thuyết	4.423
Số lớp thực hành	3
Số lớp thí nghiệm	239
Số lớp PFIEV	320
Số lớp Ngoại ngữ	163
Số khung thời gian trong tuần	60
Số học kỳ thử nghiệm	2

Bảng 3.1 trình bày các thông số chính của bộ dữ liệu được sử dụng trong quá trình đánh giá hệ thống. Các thông số này là cơ sở để đánh giá quy mô bài toán và khả năng xử lý của thuật toán.

3.3.2 Kết quả xếp thời khóa biểu

Sau khi thực hiện quá trình tối ưu hóa bằng bộ giải CP-SAT, hệ thống sinh ra các bản ghi thời khóa biểu cho các lớp học phần thuộc học kỳ được lựa chọn.

Kết quả thực nghiệm cho thấy phần lớn các lớp học phần có thể được bố trí thành công trong phạm vi các tài nguyên hiện có của nhà trường. Các ràng buộc cứng được bảo đảm thỏa mãn hoàn toàn, đồng thời các tiêu chí tối ưu của ràng buộc mềm cũng được cải thiện.

Bảng 3.3. Kết quả xếp thời khóa biểu học kỳ

Tiêu chí đánh giá	Kết quả
Tổng số lớp học phần cần xếp	4.582
Số lớp học phần xếp thành công	4.501
Số lớp học phần chưa xếp được	81
Tỷ lệ xếp thành công	98,2%
Số xung đột giảng viên	0
Số xung đột phòng học	0
Số lớp vi phạm sức chứa phòng	0
Số lớp vi phạm loại phòng	0
Số lớp PFIEV vi phạm phòng được phép	0
Số lớp Ngoại ngữ vi phạm phòng được phép	0
Thời gian chạy thuật toán	1200 giây

Bảng 3.3. Kết quả xếp thời khóa biểu theo học kỳ theo từng kì

Học kì	Cần xếp	Thành công	Chưa xếp	Tỷ lệ
HK1	2452	2405	47	98,1%
HK2	2130	2096	34	98,4%
Cả năm	4582	4501	81	98,2%

Bảng 3.2 và 3.3 thể hiện số lượng lớp học phần đã được xếp lịch thành công, số lượng lớp học phần chưa thể bố trí và các thống kê liên quan đến quá trình xử lý. Việc số xung đột giảng viên, xung đột phòng học và vi phạm sức chứa đều bằng 0 cho thấy mô hình ràng buộc cứng được xây dựng phù hợp với yêu cầu nghiệp vụ.

3.3.3 Đánh giá chất lượng thời khóa biểu

Để đánh giá chất lượng của thời khóa biểu được tạo ra, đề tài tiến hành phân tích dựa trên các tiêu chí tối ưu đã được xây dựng trong hệ thống.

Bảng 3.5. Kết quả đánh giá chất lượng thời khóa biểu

Tiêu chí đánh giá	Kết quả
Số lớp học vào thứ bảy	818 lớp
Số giảng viên có lịch dạy trên 4 ngày/tuần	292
Tổng số khoảng trống trong lịch giảng viên	1.631
Số trường hợp giảng viên dạy kín cả ngày	295
Mức sử dụng phòng học trung bình	31,1%
Số lớp PFIEV được xếp vào khu P	80 / 318
Số lớp Ngoại ngữ được xếp vào khu B tầng 1, 2	49 / 160

Kết quả đánh giá cho thấy:

- Số lượng lớp học phần được bố trí vào ngày thứ bảy được hạn chế.
- Lịch giảng dạy của giảng viên được gom lại hợp lý hơn.
- Khoảng trống giữa các buổi giảng dạy được giảm.
- Việc sử dụng phòng học phù hợp hơn với quy mô lớp học.
- Các lớp PFIEV và lớp Ngoại ngữ được ưu tiên bố trí đúng khu vực mong muốn.

Nhờ đó, thời khóa biểu được tạo ra không chỉ hợp lệ mà còn đáp ứng tốt hơn các yêu cầu thực tế của nhà trường.

3.3.4 Đánh giá hiệu năng hệ thống

Bên cạnh chất lượng thời khóa biểu, hiệu năng xử lý của hệ thống cũng được đánh giá. Trong quá trình thực nghiệm, hệ thống được triển khai trên môi trường phát triển với cơ sở dữ liệu SQL Server và bộ giải CP-SAT được tích hợp trong Backend ASP.NET Core.

Bảng 3.6. Kết quả đánh giá hiệu năng hệ thống

Tiêu chí đánh giá	Kết quả
Thời gian tải danh sách lớp học phần	2,71 giây
Thời gian tải danh sách phòng học	0,11 giây
Thời gian tải danh sách giảng viên	0,09 giây
Thời gian chạy thuật toán trung bình	25,5 giây
Số batch xử lý	282
Số lớp xử lý trung bình mỗi batch	16
Thời gian phản hồi API tra cứu lịch giảng viên	0,08 giây

Kết quả thực nghiệm cho thấy thời gian xử lý tăng gần tuyến tính theo số lượng lớp học phần trong từng đợt xử lý (batch). Việc chia dữ liệu thành nhiều batch giúp giảm đáng kể kích thước mô hình CP-SAT ở mỗi lần giải, hạn chế hiện tượng bùng nổ số lượng biến quyết định và ràng buộc.

Ngoài ra, việc lưu ngay kết quả của từng batch vào cơ sở dữ liệu giúp giảm nguy cơ mất dữ liệu khi quá trình tối ưu bị gián đoạn, đồng thời cho phép người quản trị tiếp tục chạy các batch còn lại mà không phải thực hiện lại toàn bộ quá trình xếp lịch.

Trong các lần thử nghiệm, bộ giải CP-SAT đều tìm được lời giải hợp lệ trong giới hạn thời gian cấu hình. Điều này cho thấy mô hình được xây dựng phù hợp với quy mô dữ liệu thực tế của Trường Đại học Bách khoa – Đại học Đà Nẵng.

3.4 Đánh giá kết quả đạt được

Phần này đánh giá tổng quan các kết quả đạt được sau khi triển khai hệ thống, đồng thời chỉ ra các hạn chế còn tồn tại và đề xuất hướng phát triển.

3.4.1 Kết quả đạt được

Sau quá trình nghiên cứu và phát triển, đồ án đã đạt được các kết quả chính sau đây:

- Xây dựng thành công hệ thống DUT Smart Schedule dưới dạng ứng dụng web.
- Quản lý tập trung dữ liệu giảng viên, phòng học và lớp học phần.
- Hỗ trợ nhập dữ liệu từ các nguồn hiện có của nhà trường.
- Xây dựng cơ chế xác thực bằng tài khoản Microsoft và phân quyền người dùng.
- Xây dựng dashboard thống kê phục vụ công tác quản lý và theo dõi dữ liệu đào tạo.
- Xây dựng thành công thuật toán hỗ trợ sắp xếp thời khóa biểu dựa trên bộ giải CP-SAT.
- Mô hình hóa các ràng buộc thực tế của nhà trường và tích hợp vào thuật toán xếp lịch.
- Tạo được thời khóa biểu đáp ứng các ràng buộc về giảng viên, phòng học, sức chứa và loại phòng.
- Hỗ trợ giảng viên tra cứu lịch giảng dạy trực tuyến.
- Cung cấp các chức năng thống kê và theo dõi kết quả xếp lịch.

Các kết quả đạt được cho thấy hệ thống có khả năng hỗ trợ hiệu quả công tác quản lý đào tạo và xây dựng thời khóa biểu tại Trường Đại học Bách khoa – Đại học Đà Nẵng.

3.4.2 Hạn chế

Mặc dù đạt được các mục tiêu đề ra, hệ thống vẫn còn một số hạn chế nhất định.

Thứ nhất, thuật toán hiện mới tập trung vào bài toán xây dựng thời khóa biểu giảng dạy và chưa hỗ trợ bài toán xếp lịch thi.

Thứ hai, một số tiêu chí tối ưu hiện mới được xây dựng dưới dạng các hệ số phạt đơn giản và chưa phản ánh đầy đủ toàn bộ nhu cầu thực tế của nhà trường.

Thứ ba, khi quy mô dữ liệu tăng rất lớn, thời gian xử lý của bộ giải cũng tăng theo và có thể ảnh hưởng đến hiệu năng của hệ thống.

Thứ tư, chất lượng lời giải phụ thuộc vào dữ liệu đầu vào và bộ trọng số cấu hình. Nếu dữ liệu thiếu, sai định dạng hoặc các ràng buộc cứng quá chặt, hệ thống có thể không tìm được lời giải hợp lệ.

Ngoài ra, hệ thống hiện vẫn hoạt động độc lập và chưa tích hợp trực tiếp với các hệ thống quản lý đào tạo hiện có của nhà trường.

3.4.3 Hướng phát triển

Trong tương lai, hệ thống có thể được mở rộng theo các hướng sau:

- Tối ưu hóa thêm các tiêu chí chất lượng của thời khóa biểu.
- Hỗ trợ xây dựng lịch thi và phân bổ phòng thi.
- Tích hợp với hệ thống quản lý đào tạo của nhà trường.
- Tự động đồng bộ dữ liệu từ các nguồn đào tạo hiện có.
- Hỗ trợ điều chỉnh thời khóa biểu theo thời gian thực.
- Bổ sung chức năng ghi nhận nguyện vọng lịch dạy của giảng viên.
- Bổ sung chức năng phân tích nguyên nhân không tìm được lời giải.
- Nghiên cứu áp dụng các phương pháp tối ưu hóa nâng cao nhằm cải thiện chất lượng lời giải.

Việc tiếp tục phát triển theo các hướng trên sẽ giúp nâng cao khả năng ứng dụng thực tế của hệ thống và hỗ trợ tốt hơn cho công tác quản lý đào tạo trong môi trường đại học.

Chương này đã trình bày quá trình triển khai hệ thống DUT Smart Schedule, các chức năng đã được xây dựng và mô hình thuật toán được sử dụng để giải quyết bài toán sắp xếp thời khóa biểu. Bên cạnh đó, chương cũng trình bày kết quả thực nghiệm trên dữ liệu thực tế của Trường Đại học Bách khoa – Đại học Đà Nẵng và đánh giá hiệu quả hoạt động của hệ thống. Các kết quả thu được cho thấy hệ thống có khả năng hỗ trợ tốt công tác quản lý dữ liệu đào tạo và xây dựng thời khóa biểu, đồng thời tạo nền tảng cho các nghiên cứu và phát triển tiếp theo trong tương lai.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong quá trình thực hiện đồ án tốt nghiệp với đề tài **"Xây dựng website quản lý và sắp xếp thời khóa biểu, phòng học tại trường đại học"**, em đã nghiên cứu bài toán lập thời khóa biểu trong môi trường đại học, khảo sát thực tế tại Trường Đại học Bách khoa – Đại học Đà Nẵng và tiến hành xây dựng hệ thống DUT Smart Schedule nhằm hỗ trợ công tác quản lý đào tạo và sắp xếp thời khóa biểu.

Đề tài đã nghiên cứu bài toán lập thời khóa biểu và phân bổ phòng học trong môi trường đại học, đồng thời xây dựng thành công hệ thống DUT Smart Schedule dưới dạng ứng dụng web. Hệ thống đáp ứng đầy đủ các chức năng quản lý dữ liệu đào tạo, quản lý giảng viên, phòng học, lớp học phần, cấu hình tham số, sinh thời khóa biểu tự động và tra cứu kết quả.

Về mặt thuật toán, đề tài đã mô hình hóa bài toán dưới dạng bài toán tối ưu hóa ràng buộc và áp dụng bộ giải Google OR-Tools CP-SAT để giải quyết. Hệ thống hỗ trợ nhiều ràng buộc cứng như không trùng giảng viên, không trùng phòng học, đủ sức chứa, đúng loại phòng học và nhiều ràng buộc mềm nhằm nâng cao chất lượng thời khóa biểu như hạn chế học vào thứ bảy, giảm số ngày lên trường của giảng viên, giảm khoảng trống lịch dạy và tối ưu việc sử dụng phòng học.

Kết quả thực nghiệm cho thấy hệ thống có khả năng xử lý dữ liệu với quy mô lớn, sinh thời khóa biểu hợp lệ trong thời gian phù hợp và hỗ trợ hiệu quả công tác quản lý đào tạo.

Bên cạnh những kết quả đạt được, đề tài vẫn còn một số hạn chế nhất định. Hệ thống hiện mới tập trung vào bài toán thời khóa biểu giảng dạy và chưa hỗ trợ các nghiệp vụ khác như xếp lịch thi hoặc điều chỉnh lịch học theo thời gian thực. Ngoài ra, một số tiêu chí tối ưu vẫn có thể được cải thiện nhằm nâng cao hơn nữa chất lượng của thời khóa biểu được tạo ra.

Trong tương lai, hệ thống có thể tiếp tục mở rộng theo các hướng như tích hợp bài toán xếp lịch thi, hỗ trợ điều chỉnh thời khóa biểu theo thời gian thực, bổ sung các tiêu chí tối ưu mới và kết nối trực tiếp với các hệ thống quản lý đào tạo hiện có của nhà trường.

Thông qua quá trình thực hiện đồ án, em không chỉ vận dụng được các kiến thức về phân tích thiết kế hệ thống, cơ sở dữ liệu, phát triển ứng dụng web và tối ưu hóa mà còn tích lũy thêm nhiều kinh nghiệm thực tế trong việc xây dựng một hệ thống phần mềm hoàn chỉnh. Đây là nền tảng quan trọng để em tiếp tục học tập, nghiên cứu và phát triển trong lĩnh vực công nghệ phần mềm sau khi tốt nghiệp.

TÀI LIỆU THAM KHẢO

[1] Trường Đại học Bách khoa – Đại học Đà Nẵng, *Quy chế đào tạo trình độ đại học áp dụng đối với các khóa tuyển sinh từ năm 2021 trở về sau*, ban hành kèm Quyết định số 2244/QĐ-ĐHBK.

[2] Đại học Đà Nẵng, *Kế hoạch năm học 2025–2026*, Phụ lục lịch tuần năm học 2025–2026.

[3] Google, *Google OR-Tools Documentation*, Available: https://developers.google.com/optimization

[4] Google, *CP-SAT Solver Guide*, Available: https://developers.google.com/optimization/cp/cp_solver

[5] Microsoft, *ASP.NET Core Documentation*, Available: https://learn.microsoft.com/aspnet/core

[6] Microsoft, *Entity Framework Core Documentation*, Available: https://learn.microsoft.com/ef/core

[7] Microsoft, *SQL Server Documentation*, Available: https://learn.microsoft.com/sql

[8] Microsoft, *Microsoft Identity Platform Documentation*, Available: https://learn.microsoft.com/entra/identity-platform

[9] React Team, *React Documentation*, Available: https://react.dev

[10] OpenJS Foundation, *Node.js Documentation*, Available: https://nodejs.org

PHỤ LỤC

Mã nguồn của hệ thống

[1] Mã nguồn Frontend:

Đường dẫn: https://github.com/hahung200903/dut-smartschedule-fe

[2] Mã nguồn Backend:

Đường dẫn: https://github.com/hahung200903/dut-smartschedule-be