

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM
KHOA: CÔNG NGHỆ THÔNG TIN



NGÔ QUỐC HOÀNG - 19477071
TRẦN TUẤN VŨ - 19474281

HỆ THỐNG QUẢN LÝ ĐỀ CƯƠNG NGÀNH HỌC

NGÀNH: KHOA HỌC DỮ LIỆU

Giảng viên hướng dẫn: ThS. Trương Vĩnh Linh

TP. HỒ CHÍ MINH, THÁNG 5 NĂM 2023

INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY
FACULTY OF INFORMATION TECHNOLOGY



NGO QUOC HOANG - 19477071

TRAN TUAN VU - 19474281

CURRICULUM MANAGEMENT SYSTEM

Major: Data Science

Supervisor: MSc. Truong Vinh Linh

HO CHI MINH CITY, JUNE 2023

CONTENT SUMMARY

Title: Curriculum Management System

Abstract:

The reason for selecting the topic: The management system for curriculum in academic disciplines" stems from the recognition that the process of drafting and managing curriculum, both within the field of Data Science and in most other disciplines, encounters numerous shortcomings. The lack of consistency in curriculum development among disciplines leads to the creation of unsynchronized academic programs, with the potential of hindering effective teaching and learning outcomes. Inconsistent practices in curriculum development give rise to issues such as information loss, deficiencies, and incoherence between different versions, along with limited communication capabilities among relevant stakeholders. These problems can result in information gaps, increased time and effort during the curriculum drafting process. Therefore, the development of a curriculum management system for academic disciplines is highly necessary. This system focuses on the management of curriculum in academic disciplines, with the objective of providing a convenient and efficient platform for managing and accessing information related to curriculum of courses and academic disciplines, specifically tailored for the field of Data Science.

Problems:

Firstly, the lack of consistency in curriculum development poses obstacles to adjusting the objectives, content, and evaluation of courses. This inconsistency hinders the coherence and comprehensiveness of the courses.

Secondly, the insufficient information and inconsistency in different versions of the curriculum impede effective updating and assessing changes over time. The lack of information or inconsistency between curriculum versions makes it challenging to locate and access important information, causing difficulties in

updating and upgrading the curriculum. This process consumes a significant amount of time and effort for instructors.

Thirdly, limited communication between relevant stakeholders during the development and implementation of the curriculum contributes to various issues. Inadequate communication and collaboration between instructors and department heads create barriers in completing the curriculum for courses. The lack of continuous exchange between department heads and instructors in curriculum development prolongs the curriculum drafting process and necessitates reevaluating the objectives of the curriculum.

Methods:

Firstly, to process data from docx and pdf files, the team utilized the Python programming language, specifically the docx and py2pdf libraries, to extract data from these files. This process enabled the team to gather information from the course curricula.

Next, the team implemented the development of a MySQL database to store and manage the curriculum information. MySQL is a popular and flexible database management system that allows the team to efficiently organize data and perform interactive data queries with ease.

In terms of developing the user interface for user interaction, the team employed TailwindCSS and ReactJS. TailwindCSS is a versatile and powerful CSS framework that enables the team to create visually appealing interfaces, while ReactJS enhances the user experience.

For the backend, the team utilized .NET and the C# programming language for development. .NET provides a robust platform for web application and web service development, and C# is a versatile and easy-to-learn programming language that allows the team to build server-side functions and handle data processing.

Result:

We have developed a curriculum management system. With the department head, they have full privileges including: managing the curriculum, managing courses, managing faculty, and viewing the dashboard. With the faculty, they can

only: view the dashboard, access information, and are allowed to edit courses upon request from the department head

Conclusion:

The curriculum management system has successfully met the initial requirements. It provides support for department heads and instructors in constructing and managing the curriculum for academic programs and courses. However, there are still some areas that require improvement, such as enhancing the interaction functionalities between department heads and instructors, upgrading the system interface to be more intuitive and user-friendly, and expanding the scope of topics covered.

LỜI CẢM ƠN

Để thực hiện và hoàn thành được Đồ án tốt nghiệp “Hệ thống quản lý đề cương ngành học”, chúng em đã nhận được sự hỗ trợ, giúp đỡ cũng như những lời quan tâm động viên đến từ các thầy/ cô là giảng viên của Khoa Công nghệ Thông tin.

Lời đầu tiên chúng em xin được bày tỏ lòng biết ơn sâu sắc đến Ban Giám hiệu Trường Đại học Công nghiệp Thành phố Hồ Chí Minh, Ban Chủ nhiệm Khoa Công nghệ Thông tin, Ban Chủ nhiệm Bộ môn Khoa học Dữ liệu đã tận tình giúp đỡ chúng em trong suốt thời gian học tập tại trường.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn chân thành đến thầy ThS. Trương Vĩnh Linh - người trực tiếp hướng dẫn chúng em hoàn thành Đồ án tốt nghiệp lần này. Cảm ơn thầy vì đã là người luôn dành nhiều thời gian, công sức, trực tiếp hướng dẫn chúng em. Nhờ có những lời hướng dẫn, dạy bảo ấy nên bài luận văn này của chúng em đã hoàn thành một cách xuất sắc nhất. Một lần nữa, chúng em xin gửi lời cảm ơn chân thành đến thầy.

Mặc dù đã cố gắng rất nhiều, nhưng Đồ án tốt nghiệp của chúng em sẽ không thể tránh khỏi những thiếu sót; rất mong nhận được sự thông cảm, chỉ dẫn, giúp đỡ và đóng góp ý kiến của Quý thầy/cô là giảng viên, những người quan tâm đến đề tài này giúp đỡ để Đồ án tốt nghiệp của chúng em được hoàn thiện hơn.

Một lần nữa, chúng em xin chân thành cảm ơn!

NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ngày tháng năm 2023
GIẢNG VIÊN HƯỚNG DẪN
(Ký và ghi rõ họ tên)

NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN 1

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ngày tháng năm 2023

GIẢNG VIÊN PHẢN BIỆN 1

(Ký và ghi rõ họ tên)

NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN 2

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ngày tháng năm 2023

GIẢNG VIÊN PHẢN BIỆN 2

(Ký và ghi rõ họ tên)

MỤC LỤC

| | |
|--|-----------|
| LỜI CẢM ƠN | 4 |
| NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN | 5 |
| NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN 1 | 6 |
| NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN 2 | 7 |
| MỤC LỤC | 8 |
| DANH MỤC HÌNH ẢNH | 15 |
| DANH MỤC CÁC BẢNG BIỂU | 17 |
| CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI | 1 |
| 1.1 Mục đích nghiên cứu đề tài | 1 |
| 1.2 Lý do chọn đề tài | 1 |
| 1.3 Phạm vi đề tài | 1 |
| 1.4 Bố cục đồ án | 1 |
| CHƯƠNG 2: CƠ SỞ LÝ THUYẾT | 3 |
| 2.1 Docker | 3 |
| 2.1.1 Docker là gì? | 3 |
| 2.1.1 Lý do nên sử dụng Docker | 3 |
| 2.1.2 Một số ứng dụng của Docker | 4 |
| 2.1.3 Kiến trúc của Docker | 4 |
| 2.1.4 Chức năng các thành phần trong Docker | 5 |
| 2.1 Reactjs | 6 |
| 2.2.1 Lịch sử và phát triển của Reactjs | 6 |
| 2.2.2 Tại sao nên sử dụng Reactjs | 7 |
| 2.2.3 Các khái niệm cơ bản của Reactjs | 8 |
| 2.2.3.1 Components và props | 8 |

| | |
|---|----|
| 2.2.3.2 State và lifecycle methods..... | 9 |
| 2.2.3.3 Virtual DOM..... | 9 |
| 2.2.3.4 JSX syntax | 11 |
| 2.2.3 Môi trường và các thư viện hỗ trợ Reactjs | 11 |
| 2.2.3.1 Cài đặt môi trường phát triển..... | 11 |
| 2.2.3.2 Một số thư viện hỗ trợ Reactjs..... | 11 |
| 2.2.4 Các kỹ thuật lập trình trong Reactjs | 15 |
| 2.2.4.1 React Hooks | 15 |
| 2.2.4.2 Higher-order components | 16 |
| 2.2.4.3 Render props | 17 |
| 2.2.3.5 Context API | 17 |
| 2.2.5 Các vấn đề thường gặp khi sử dụng Reactjs | 18 |
| 2.2.5.1 Một số lỗi thường gặp trong reactjs | 18 |
| 2.5.2 Cách khắc phục một số lỗi thường gặp..... | 19 |
| 2.5.3 Debugging và logging trong ReactJS | 20 |
| 2.3 TailwindCSS | 20 |
| 2.3.1 Giới thiệu..... | 20 |
| 2.3.1.1 Tổng quan về TailwindCSS | 20 |
| 2.3.1.2 Lịch sử của TailwindCSS..... | 21 |
| 2.3.1.3 Lý do nên dùng TailwindCSS | 21 |
| 2.3.1.4 Các tính năng của TailwindCSS | 22 |
| 2.3.2 Các khái niệm cơ bản trong TailwindCSS | 24 |
| 2.3.2.1 CSS utility classes..... | 24 |
| 2.3.2.2 Configuration file | 24 |

| | |
|--|----|
| 2.3.2.3 Responsive design..... | 25 |
| 2.3.2.4 Utility-first approach | 26 |
| 2.3.2.5 Predefined color palette..... | 26 |
| 2.3.2.6 Customization..... | 27 |
| 2.3.3 Các tính năng chính trong TailwindCSS..... | 27 |
| 2.3.3.1 Layout | 27 |
| 2.3.3.2 Typography | 28 |
| 2.3.3.3 Color | 28 |
| 2.3.3.4 Sizing | 29 |
| 2.3.3.5 Borders..... | 29 |
| 2.3.3.6 Shadows | 29 |
| 2.3.3.7 Interactivity | 29 |
| 2.3.4 So sánh Tailwindcss với các công cụ khác..... | 30 |
| 2.3.4.1 So sánh Tailwindcss với Bootstrap | 30 |
| 2.3.4.2 So sánh Tailwindcss với Foundation | 31 |
| 2.3.4.3 So sánh với CSS Grid và Flexbox..... | 33 |
| 2.3.4.4 Tổng kết | 35 |
| 2.4 ASP.NET | 36 |
| 2.4.1 Giới thiệu..... | 36 |
| 2.4.1.1 Lịch sử và các phiên bản ASP.NET | 36 |
| 2.4.1.2 Kiến trúc của ASP.NET và cách hoạt động của nó | 37 |
| 2.4.2 Các ngôn ngữ lập trình được hỗ trợ bởi ASP.NET | 38 |
| 2.4.3 Các công nghệ liên quan đến ASP.NET..... | 39 |
| 2.4.3.1 Entity Framework và ADP.NET cho việc tương tác với cơ sở dữ liệu | 39 |

| | |
|--|----|
| 2.4.3.2 ASP.NET Identity cho việc xác thực và phân quyền người dùng | 40 |
| 2.4.3.3 Các thư việnJavaScripts và CSS phổ biến được sử dụng trong ASP.NET | 40 |
| 2.4.4 Framework ASP.NET CORE API | 41 |
| 2.4.4.1 Giới thiệu về Core API | 41 |
| 2.4.4.2 Kiến trúc Core API | 42 |
| 2.4.5 EntityFramework Core trong ASP.NET | 43 |
| 2.4.5.1 Giới thiệu về Entity Framework Core | 43 |
| 2.4.5.2 Mục đích sử dụng EntityFramework Core | 43 |
| 2.4.5.3 Điểm khác biệt giữa Entity Framework và Entity Framework Core | 44 |
| 2.4.5.4 Các thành phần của Entity Framework Core | 44 |
| 2.5 C# | 45 |
| 2.5.1 Giới thiệu về ngôn ngữ C# | 45 |
| 2.5.2 Lịch sử và các phiên bản của C# | 46 |
| 2.5.3 Vì sao nên chọn C# để làm backend cho web | 46 |
| 2.5.4 Kiến thức cần có để bắt đầu làm việc với C# cho backend web | 47 |
| 2.6 Python-docx | 48 |
| 2.7 PYPDF2 | 50 |
| 2.7.1 Giới thiệu | 50 |
| 2.7.2 Các tính năng của thư viện pypdf2 | 51 |
| 2.7.3 Ưu điểm của thư viện pypdf2 | 51 |
| CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG | 53 |
| 3.1 Mô tả nghiệp vụ hệ thống | 53 |
| 3.1.1 Hỗ trợ phía người quản lý (Trưởng bộ môn) | 53 |

| | |
|---|----|
| 3.1.2 Hỗ trợ phía người dùng (Trưởng bộ môn & Giảng viên) | 53 |
| 3.2 Chức năng của hệ thống | 53 |
| 3.2.1 Người quản lí (Trưởng bộ môn) | 53 |
| 3.2.2 Người dùng (Giảng viên) | 54 |
| 3.3 Xác định yêu cầu hệ thống quản lí đề cương ngành học | 54 |
| 3.4 Các yêu cầu chức năng | 54 |
| 3.4.1 Đăng nhập | 54 |
| 3.4.2 Trưởng bộ môn | 55 |
| 3.4.3 Giảng viên | 56 |
| 3.5 Phân tích và thiết kế | 57 |
| 3.5.1 Xây dựng lược đồ use case | 57 |
| 3.5.1.1 Use Case người dùng tổng quát | 57 |
| 3.5.1.2 Use Case Đăng nhập | 57 |
| 3.5.1.3 Use Case Quản lý môn học | 58 |
| 3.5.1.4 Use Case Quản lý chương trình | 58 |
| 3.5.1.5 Use Case Quản lý giảng viên | 59 |
| 3.5.1.6 Danh sách mô tả các chức năng | 59 |
| 3.5.2 Đặc tả use case | 61 |
| 3.5.2.1 UC001 – Use Case Đăng nhập | 61 |
| 3.5.2.2 UC002 – Use Case Tìm kiếm môn học | 62 |
| 3.5.2.3 UC003 – Usecase Chỉnh sửa môn học | 62 |
| 3.5.2.4 UC004 – Usecase In môn học | 64 |
| 3.5.2.5 UC005 – Usecase Tạo môn học | 65 |
| 3.5.2.6 UC006 – Usecase Sao chép môn học | 66 |

| | |
|--|----|
| 3.5.2.7 UC007 – Usecase Xóa môn học | 67 |
| 3.5.2.8 UC008 – Usecase Tạo mới chương trình..... | 68 |
| 3.5.2.9 UC009 – Usecase Chỉnh sửa chương trình | 70 |
| 3.5.2.10 UC010 – Usecase Sao chép chương trình..... | 71 |
| 3.5.2.11 UC011 – Usecase Xóa chương trình | 72 |
| 3.5.2.12 UC012 – Usecase In chương trình | 73 |
| 3.5.2.13 UC013 – Thêm giảng viên | 74 |
| 3.5.2.14 UC014 – Reset mật khẩu giảng viên..... | 75 |
| 3.5.2.15 UC015 – Xóa giảng viên | 76 |
| 3.5.2.16 UC016 – Tạo yêu cầu giảng viên..... | 77 |
| 3.5.3 Sơ đồ activity..... | 79 |
| 3.5.3.1 Sơ đồ activity use case Đăng nhập..... | 79 |
| 3.5.2.2 Sơ đồ activity use case tìm kiếm môn học..... | 80 |
| 3.5.2.3 Sơ đồ activity use case chỉnh sửa môn học | 81 |
| 3.5.2.4 Sơ đồ activity use case in môn học..... | 82 |
| 3.5.2.5 Sơ đồ activity use case tạo môn học..... | 83 |
| 3.5.2.6 Sơ đồ activity use case sao chép môn học | 84 |
| 3.5.2.7 Sơ đồ activity use case xóa môn học | 85 |
| 3.5.2.8 Sơ đồ activity use case tạo mới chương trình | 86 |
| 3.5.2.9 Sơ đồ activity use case chỉnh sửa chương trình | 87 |
| 3.5.2.10 Sơ đồ activity use case sao chép chương trình | 88 |
| 3.5.2.11 Sơ đồ activity use case xóa chương trình | 89 |
| 3.5.2.12 Sơ đồ activity use case in chương trình..... | 90 |
| 3.5.2.13 Sơ đồ activity use case thêm giảng viên | 91 |

| | |
|--|-----|
| 3.5.2.14 Sơ đồ activity use case reset mật khẩu tài khoản giảng viên | 92 |
| 3.5.2.15 Sơ đồ activity use case xóa giảng viên | 93 |
| 3.5.2.16 Sơ đồ activity use case tạo yêu cầu giảng viên..... | 94 |
| 3.6 Mô hình tổ chức dữ liệu | 94 |
| 3.7 Mô tả chi tiết các bảng..... | 95 |
| 3.8 Lược đồ quan hệ | 99 |
| CHƯƠNG 4: THIẾT KẾ VÀ HIỆN THỰC ỨNG DỤNG..... | 101 |
| 4.1 Chuyển đổi từ file PDF, Docx sang database..... | 101 |
| 4.1.1 Trích xuất tài liệu từ file Docx..... | 101 |
| 4.1.2 Trích xuất tài liệu từ file PDF..... | 119 |
| 4.2 Tổng quan về ứng dụng..... | 127 |
| 4.3 Một số chức năng của ứng dụng..... | 128 |
| 4.3.1 Đăng nhập | 128 |
| 4.3.2 Giao diện Dashboard của trưởng bộ môn..... | 129 |
| 4.3.3 Giao diện quản lý chương trình | 129 |
| 4.3.4 Giao diện quản lý môn học | 133 |
| 4.3.5 Giao diện quản lý giảng viên | 137 |
| 4.4 Ưu điểm..... | 140 |
| 4.5 Hạn chế | 141 |
| CHƯƠNG 5: KẾT LUẬN | 142 |
| 5.1 Những điều đạt được..... | 142 |
| 5.2 Những vấn đề thuận lợi..... | 142 |
| 5.3 Hướng phát triển của đề tài..... | 142 |
| TÀI LIỆU THAM KHẢO | 144 |

DANH MỤC HÌNH ẢNH

| | |
|---|-----|
| Hình 2. 1 Kiến trúc của Docker | 4 |
| Hình 2. 2 Python | 48 |
| Hình 2. 3 PyPDF2..... | 50 |
| Hình 3. 1 Use case người dùng tổng quát | 57 |
| Hình 3. 2 Use case đăng nhập | 57 |
| Hình 3. 3 Use case quản lý môn học | 58 |
| Hình 3. 4 Use case quản lý chương trình | 58 |
| Hình 3. 5 Use case quản lý giảng viên..... | 59 |
| Hình 3. 6 Sơ đồ activity biểu diễn use case đăng nhập | 79 |
| Hình 3. 7 Sơ đồ activity biểu diễn use case tìm kiếm môn học..... | 80 |
| Hình 3. 8 Sơ đồ activity biểu diễn use case chỉnh sửa môn học | 81 |
| Hình 3. 9 Sơ đồ activity biểu diễn use case in môn học..... | 82 |
| Hình 3. 10 Sơ đồ activity biểu diễn use case tạo môn học | 83 |
| Hình 3. 11 Sơ đồ activity biểu diễn use case sao chép môn học..... | 84 |
| Hình 3. 12 Sơ đồ activity biểu diễn use case xóa môn học | 85 |
| Hình 3. 13 Sơ đồ activity biểu diễn use case tạo mới chương trình..... | 86 |
| Hình 3. 14 Sơ đồ activity biểu diễn use case chỉnh sửa chương trình..... | 87 |
| Hình 3. 15 Sơ đồ activity biểu diễn use case sao chép chương trình..... | 88 |
| Hình 3. 16 Sơ đồ activity biểu diễn use case xóa chương trình | 89 |
| Hình 3. 17 Sơ đồ activity biểu diễn use case in chương trình | 90 |
| Hình 3. 18 Sơ đồ activity biểu diễn use case thêm giảng viên | 91 |
| Hình 3. 19 Sơ đồ activity biểu diễn use case reset mật khẩu | 92 |
| Hình 3. 20 Sơ đồ activity biểu diễn use case xóa giảng viên..... | 93 |
| Hình 3. 21 Sơ đồ activity biểu diễn use case tạo yêu cầu giảng viên | 94 |
| Hình 3. 22 Lược đồ quan hệ môn học..... | 99 |
| Hình 3. 23 Lược đồ quan hệ hệ thống..... | 100 |
| Hình 4. 1 Mẫu Nhập môn tin học 1..... | 102 |
| Hình 4. 2 Mẫu Nhập môn tin học 2..... | 103 |

| | |
|--|-----|
| Hình 4. 3 Mẫu Nhập môn tin học 3..... | 104 |
| Hình 4. 4 Mẫu Nhập môn tin học 4..... | 105 |
| Hình 4. 5 Mẫu nhập môn tin học 5 | 106 |
| Hình 4. 6 Mẫu Nhập môn tin học 6..... | 107 |
| Hình 4. 7 Mẫu Nhập môn tin học 7..... | 108 |
| Hình 4. 8 Mẫu Nhập môn tin học 8..... | 109 |
| Hình 4. 9 Mẫu cần trích xuất..... | 120 |
| Hình 4. 10 Giao diện đăng nhập của hệ thống | 128 |
| Hình 4. 11 Giao diện Dashboard khi trưởng bộ môn đăng nhập | 129 |
| Hình 4. 12 Giao diện quản lý chương trình..... | 130 |
| Hình 4. 13 Giao diện thêm chương trình | 130 |
| Hình 4. 14 Dropdown chọn học kì tương ứng với số lượng học kì đã nhập..... | 131 |
| Hình 4. 15 Giao diện sao chép chương trình..... | 131 |
| Hình 4. 16 Thông báo khi xóa chương trình thành công | 132 |
| Hình 4. 17 Giao diện in chương trình học..... | 132 |
| Hình 4. 18 Giao diện khi chưa chọn môn học để xem thông tin | 133 |
| Hình 4. 19 Giao diện khi chọn vào môn học để xem thông tin chi tiết | 133 |
| Hình 4. 20 Giao diện thêm môn học | 134 |
| Hình 4. 21 Giao diện chỉnh sửa môn học..... | 135 |
| Hình 4. 22 Giao diện sao chép môn học | 135 |
| Hình 4. 23 Thông báo đã xóa môn học thành công | 136 |
| Hình 4. 24 Giao diện tạo yêu cầu cho giảng viên | 136 |
| Hình 4. 25 Giao diện quản lý giảng viên | 137 |
| Hình 4. 26 Giao diện thêm giảng viên | 138 |
| Hình 4. 27 Giao diện chỉnh sửa thông tin giảng viên | 139 |
| Hình 4. 28 Giao diện xác nhận khi reset mật khẩu giảng viên | 140 |
| Hình 4. 29 Giao diện xác nhận khi xóa giảng viên khỏi hệ thống..... | 140 |

DANH MỤC CÁC BẢNG BIỂU

| | |
|---|-----|
| Bảng 3. 1 Danh sách mô tả các chức năng..... | 59 |
| Bảng 3. 2 Đặc tả UC001 – Đăng nhập..... | 61 |
| Bảng 3. 3 Đặc tả UC002 - Tìm kiếm môn học..... | 62 |
| Bảng 3. 4 Đặc tả UC003 – Chinh sửa môn học | 62 |
| Bảng 3. 5 Đặc tả UC004 – In môn học | 64 |
| Bảng 3. 6 Đặc tả UC005 – Tạo môn học | 65 |
| Bảng 3. 7 Đặc tả UC006 – Sao chép môn học..... | 66 |
| Bảng 3. 8 Đặc tả UC007 – Xóa môn học..... | 67 |
| Bảng 3. 9 Đặc tả UC008 - Tạo mới chương trình..... | 68 |
| Bảng 3. 10 Đặc tả UC009 – Chinh sửa chương trình | 70 |
| Bảng 3. 11 Đặc tả UC010 – Sao chép chương trình | 71 |
| Bảng 3. 12 Đặc tả UC011 – Xóa chương trình | 72 |
| Bảng 3. 13 Đặc tả UC012 – In chương trình | 73 |
| Bảng 3. 14 Đặc tả UC013 – Thêm giảng viên | 74 |
| Bảng 3. 15 Đặc tả UC014 – Reset mật khẩu giảng viên..... | 75 |
| Bảng 3. 16 Đặc tả UC015 – Xóa giảng viên..... | 76 |
| Bảng 3. 17 Đặc tả UC016 – Tạo yêu cầu giảng viên..... | 77 |
| Bảng 3. 18 Bảng curriculums..... | 95 |
| Bảng 3. 19 Bảng evalelements | 95 |
| Bảng 3. 20 Bảng evaluates | 96 |
| Bảng 3. 21 Bảng semesters | 96 |
| Bảng 3. 22 Bảng subjectcontents | 96 |
| Bảng 3. 23 Bảng subjectcurriculums | 97 |
| Bảng 3. 24 Bảng subjectoutputstandards | 97 |
| Bảng 3. 25 Bảng subjects | 97 |
| Bảng 4. 1 Kết quả trích xuất nội dung giảng dạy môn Nhập môn tin học..... | 111 |
| Bảng 4. 2 Kết quả trích xuất chuẩn đầu ra môn học | 126 |

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1 Mục đích nghiên cứu đề tài

Mục đích của ý tưởng là nhằm xây dựng một nền tảng hỗ trợ cho giảng viên dễ dàng xây dựng một đề cương ngành học cho bậc đại học của khoa công nghệ thông tin. Hệ thống sẽ cung cấp các tính năng để quản lý thông tin các môn học, giảng viên, qua đó xây dựng một chương trình đào tạo phù hợp với chuẩn đầu ra của ngành.

Tên đề tài: Hệ thống quản lý đề cương ngành học

1.2 Lý do chọn đề tài

Việc xây dựng hệ thống quản lý đề cương ngành học có thể giúp giảm thiểu công sức và thời gian cho người quản lý, giảm thiểu thời gian xây dựng chương trình đào tạo. Nó cũng giúp cho việc quản lý, theo dõi thông tin các khóa học, giảng viên giảng dạy trở nên dễ dàng và thuận tiện hơn. Bên cạnh đó việc sử dụng công nghệ thông tin trong việc quản lý chương trình đào tạo cũng phù hợp với xu hướng phát triển của xã hội hiện đại, giúp cho việc quản lý trở nên hiệu quả và tiết kiệm chi phí.

1.3 Phạm vi đề tài

Hệ thống quản lý đề cương ngành học cho bộ môn Khoa học Dữ liệu là một hệ thống cung cấp quyền truy cập và quản lý cho trưởng bộ môn và các giảng viên trực thuộc bộ môn Khoa học dữ liệu. Với hệ thống này, trưởng bộ môn và giảng viên có thể dễ dàng truy cập, chỉnh sửa, quản lý và tạo mới các đề cương môn học cũng như đề cương liên quan đến bộ môn Khoa học Dữ liệu. Hệ thống quản lý đề cương ngành học được thiết kế để đáp ứng các nhu cầu của trưởng bộ môn và giảng viên. Trưởng bộ môn có quyền tạo đề cương môn học, đề cương ngành học, chỉnh sửa môn học và yêu cầu giảng viên tạo đề cương môn học mới khi cần. Trong khi đó đối với giảng viên khi chưa có sự cho phép của trưởng bộ môn chỉ được xem nội dung các đề cương môn học và xuất file.

1.4 Bố cục đồ án

Nội dung đồ án chuyên ngành được trình bày theo bố cục gồm 5 chương như sau:

- Chương 1: Tổng quan đề tài

- Chương 2: Cơ sở lý thuyết
- Chương 3: Phân tích và thiết kế hệ thống
- Chương 4: Thiết kế và thực hiện ứng dụng
- Chương 5: Kết luận

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Docker

2.1.1 Docker là gì?

Docker là một công nghệ ảo hóa phần mềm được sử dụng để tạo ra các container - một môi trường đóng gói ứng dụng và các phụ thuộc của nó. Các container này giúp cho việc triển khai ứng dụng trên nhiều máy chủ khác nhau trở nên dễ dàng hơn. Với Docker, người dùng có thể đóng gói tất cả các thành phần của ứng dụng vào trong một container, bao gồm cả các thư viện, các phụ thuộc, các tập tin cấu hình và mã nguồn.

Docker cung cấp một giải pháp để đóng gói và phân phối ứng dụng dưới dạng các container độc lập với hệ thống máy chủ. Điều này giúp cho việc triển khai ứng dụng trở nên nhanh chóng và đáng tin cậy hơn, vì các container có thể chạy trên bất kỳ máy chủ nào có Docker được cài đặt. Docker cũng cho phép người dùng quản lý các container của mình, điều này bao gồm cả khởi động, dừng, xóa và cập nhật chúng.

2.1.1 Lý do nên sử dụng Docker

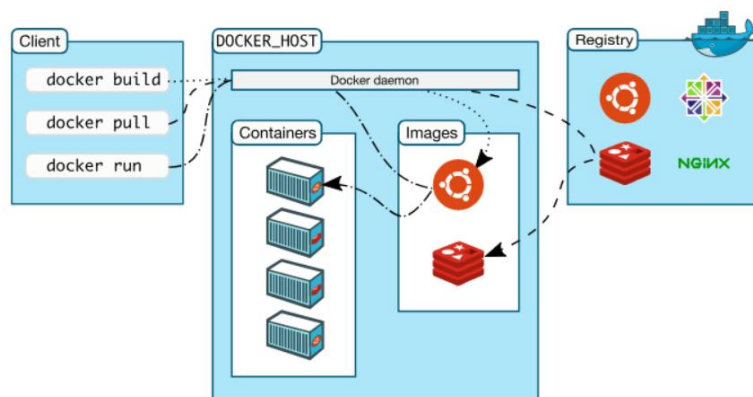
- Đóng gói và phân phối ứng dụng dễ dàng: Docker cho phép người dùng đóng gói tất cả các thành phần của ứng dụng vào trong một container, bao gồm cả các thư viện, các phụ thuộc, các tập tin cấu hình và mã nguồn. Điều này giúp cho việc triển khai ứng dụng trở nên nhanh chóng và đáng tin cậy hơn, vì các container có thể chạy trên bất kỳ máy chủ nào có Docker được cài đặt.
- Đảm bảo tính nhất quán và độ tin cậy của ứng dụng: Docker cung cấp một môi trường đóng gói độc lập với hệ thống máy chủ, giúp đảm bảo rằng ứng dụng sẽ chạy một cách nhất quán và đáng tin cậy trên nhiều nền tảng khác nhau.
- Tích hợp liên kết: Docker có thể được sử dụng để tạo ra các ứng dụng phức tạp bằng cách kết hợp nhiều container với nhau. Điều này giúp cho việc triển khai các ứng dụng phức tạp trở nên dễ dàng hơn, bởi vì mỗi container có thể được tối ưu cho một nhiệm vụ cụ thể.

- Tích hợp với các công cụ khác: Docker có thể tích hợp với các công cụ khác như Kubernetes, Docker Swarm, Ansible và Puppet, giúp cho việc triển khai và quản lý các container trở nên dễ dàng hơn.
- Tiết kiệm tài nguyên: Docker cho phép người dùng chia sẻ các tài nguyên giữa các container một cách hiệu quả, giúp giảm thiểu chi phí về tài nguyên và thời gian triển khai. [1]

2.1.2 Một số ứng dụng của Docker

- Triển khai ứng dụng: Docker cho phép người dùng đóng gói ứng dụng của mình vào trong các container và triển khai chúng trên nhiều máy chủ khác nhau một cách dễ dàng và đáng tin cậy.
- Phát triển ứng dụng: Bằng cách sử dụng Docker, người dùng có thể tạo ra các môi trường phát triển đáng tin cậy để phát triển ứng dụng mà không cần lo lắng về sự khác biệt giữa các môi trường phát triển.
- Kiểm thử ứng dụng: Docker cung cấp các container độc lập nhau, cho phép người dùng kiểm thử ứng dụng của mình trong nhiều môi trường khác nhau một cách dễ dàng.
- Tích hợp liên kết: Docker có thể được sử dụng để tạo ra các ứng dụng phức tạp bằng cách kết hợp nhiều container với nhau.

2.1.3 Kiến trúc của Docker



Hình 2. 1 Kiến trúc của Docker

Như hình ở trên chúng ta có thể thấy kiến trúc của Docker bao gồm các phần sau:

- Docker Client: Là giao diện dòng lệnh hoặc giao diện đồ họa để giao tiếp với Docker Daemon và thực hiện các lệnh Docker.
- Docker Daemon: Là một tiến trình nền chạy trên máy chủ Docker, quản lý các container, các hình ảnh và các mạng Docker.
- Docker Images: Là một tập hợp các lớp (layers) chỉ đọc được (read-only) tạo nên một ứng dụng, mỗi lớp chứa các thay đổi đối với lớp trên đó. Docker Image được sử dụng để tạo ra các container Docker.
- Docker Container: Là một môi trường thực thi ứng dụng độc lập với hệ thống máy chủ được tạo ra bằng cách sử dụng Docker Image. Container chứa tất cả các thành phần cần thiết để chạy ứng dụng, bao gồm cả các thư viện, phụ thuộc và tập tin cấu hình.
- Docker Registry: Là nơi lưu trữ các Docker Images. Có thể sử dụng Docker Hub để hỗ trợ hoặc các registry khác như Amazon ECR hoặc các registry tự xây dựng.
- Docker Compose: Là công cụ cho phép mô tả và chạy nhiều Container liên kết với nhau, hoạt động như một ứng dụng duy nhất.

Tất cả những phần trên tạo nên một môi trường đóng gói và triển khai ứng dụng, giúp cho việc triển khai và quản lý ứng dụng dễ dàng hơn và đáng tin cậy hơn.

2.1.4 Chức năng các thành phần trong Docker

- Docker Client: Là một giao diện dòng lệnh hoặc đồ họa để tương tác với Docker Daemon và thực hiện các lệnh Docker. Ví dụ, người dùng có thể sử dụng Docker CLI để tạo một container từ một Docker image bằng lệnh “docker run” hoặc xóa một container bằng lệnh “docker rm.”
- Docker Daemon: Là một tiến trình nền chạy trên máy chủ Docker, quản lý các container, các hình ảnh và các mạng Docker. Ví dụ, Docker Daemon có thể lưu trữ các image Docker trên máy chủ và cho phép Docker Client tạo và quản lý các container từ các image này.

- **Docker Images:** Là một tập hợp các lớp chỉ đọc được (read-only) tạo nên một ứng dụng, mỗi lớp chứa các thay đổi đối với lớp trên đó. Docker image được sử dụng để tạo ra các container Docker. Ví dụ, người dùng có thể sử dụng Dockerfile để định nghĩa một image Docker bao gồm các phần mềm, các tài nguyên và các cấu hình cần thiết cho ứng dụng .
- **Docker Container:** Là một môi trường thực thi ứng dụng độc lập với hệ thống máy chủ, được tạo ra bằng cách sử dụng Docker image. Container chứa tất cả các thành phần cần thiết để chạy ứng dụng, bao gồm cả các thư viện, phụ thuộc và tập tin cấu hình. Ví dụ, người dùng có thể sử dụng Docker để tạo ra một container chứa WordPress và MySQL để triển khai một trang web độc lập với hệ thống máy chủ.
- **Docker Registry:** Là nơi lưu trữ các Docker images. Có thể sử dụng Docker Hub (miễn phí) hoặc các registry khác như Amazon ECR, Google Container Registry, hay các registry tự xây dựng. Ví dụ, người dùng có thể sử dụng Docker Hub để lưu trữ các image Docker và chia sẻ chúng với người dùng khác.
- **Docker Compose:** Là công cụ cho phép mô tả và chạy nhiều container liên kết với nhau, hoạt động như một ứng dụng duy nhất. Ví dụ, người dùng có thể sử dụng Docker Compose để triển khai một ứng dụng web bao gồm các container cho Frontend, Backend và Database.

2.1 Reactjs

2.2.1 Lịch sử và phát triển của Reactjs

Reactjs là một thư viện JavaScript được phát triển bởi Facebook. Ban đầu, nó được phát hành vào năm 2013 và được phát triển dựa trên một số ứng dụng web lớn của Facebook. Reactjs đã nhanh chóng trở thành một trong những công cụ phát triển web phổ biến nhất trên thế giới. Sau đó, vào năm 2015, Facebook đã công bố phiên bản Reactjs mới nhất của họ, với sự giới thiệu của React Native. React Native là một framework cho phép lập trình viên sử dụng Reactjs để xây dựng ứng dụng di động cho cả iOS và Android. Năm 2017, Reactjs đã được cải tiến với phiên bản React 16,

trong đó có một số tính năng mới như React Context API và React Hooks, giúp cho việc quản lý trạng thái của ứng dụng trở nên dễ dàng hơn. Tính đến hiện tại, Reactjs tiếp tục được cải tiến và phát triển để đáp ứng nhu cầu ngày càng tăng của các lập trình viên. Thêm vào đó, Reactjs cũng đã được cải tiến với việc giới thiệu một số tính năng mới như React Fiber, một bản cập nhật quan trọng giúp cải thiện hiệu suất và khả năng đáp ứng của ứng dụng React. Ngoài ra, Reactjs cũng có một cộng đồng phát triển rất lớn và chủ động đóng góp vào việc cải tiến và phát triển thư viện. Điều này đã giúp Reactjs trở thành một trong những công cụ phát triển web được ưa chuộng nhất hiện nay, được sử dụng bởi hàng ngàn lập trình viên trên khắp thế giới. Năm 2020 nhóm React đã công bố ứng cử viên phát hành đầu tiên cho React v17.0, đáng chú ý là bản phát hành lớn đầu tiên mà không có những thay đổi lớn đối với API của nhà phát. Hiện tại, phiên bản mới nhất của Reactjs là React 18, được ra mắt vào cuối năm 2021, với một số tính năng mới như cải thiện hiệu suất và khả năng tương thích, cũng như khả năng cải thiện quy trình phát triển của lập trình viên. Tóm lại, Reactjs là một thư viện JavaScript mạnh mẽ và phổ biến, đã trải qua nhiều giai đoạn phát triển và cải tiến để đáp ứng nhu cầu ngày càng tăng của lập trình viên và người dùng. Nó đã trở thành một công cụ cực kỳ quan trọng trong phát triển ứng dụng web và di động hiện đại, với một cộng đồng phát triển rộng lớn và chủ động đóng góp vào việc phát triển thư viện. Với việc liên tục cải tiến và cập nhật phiên bản mới, Reactjs vẫn đang tiếp tục phát triển và có sự bền vững trong tương lai.

2.2.2 Tại sao nên sử dụng Reactjs

Reactjs là một thư viện JavaScript mạnh mẽ được sử dụng để xây dựng các ứng dụng web động, đặc biệt là các ứng dụng đơn trang (single-page applications). Dưới đây là một số lý do tại sao nên sử dụng Reactjs:

- **Hiệu suất cao:** Reactjs được thiết kế để tối ưu hóa hiệu suất và khả năng đáp ứng của ứng dụng. Các ứng dụng Reactjs có thể hoạt động rất nhanh và đáp ứng nhanh chóng với các tương tác người dùng.
- **Quản lý trạng thái dễ dàng:** Reactjs cung cấp các công cụ mạnh mẽ để quản lý trạng thái của ứng dụng. Việc quản lý trạng thái được thực hiện thông qua các

“state” của các component, giúp cho việc cập nhật trạng thái của ứng dụng trở nên dễ dàng và linh hoạt hơn.

- Tích hợp dễ dàng: Reactjs có thể được tích hợp vào các ứng dụng web hiện có một cách dễ dàng, mà không làm ảnh hưởng đến cấu trúc của ứng dụng. Nó cũng tương thích với nhiều thư viện và framework khác nhau, giúp cho việc tích hợp trở nên thuận tiện hơn.
- Cộng đồng phát triển lớn: Reactjs có một cộng đồng phát triển lớn và chủ động đóng góp vào việc phát triển thư viện. Điều này đảm bảo rằng Reactjs luôn được cập nhật và cải tiến để đáp ứng nhu cầu ngày càng tăng của các lập trình viên và người dùng.
- Hỗ trợ cho ứng dụng di động: Reactjs có thể được sử dụng để xây dựng các ứng dụng di động thông qua React Native, một framework cho phép lập trình viên sử dụng Reactjs để xây dựng ứng dụng di động cho cả iOS và Android.
- JSX: JSX là một định dạng mã nguồn mới được giới thiệu bởi Reactjs, cho phép lập trình viên viết mã HTML trong JavaScript. Điều này giúp cho việc phát triển các ứng dụng trở nên dễ dàng hơn, vì các phần HTML và JavaScript được viết cùng một chỗ và dễ dàng nhận biết được các thành phần của ứng dụng.
- Thử nghiệm dễ dàng: Reactjs có các công cụ thử nghiệm rất mạnh mẽ, giúp cho việc kiểm tra và phát triển các ứng dụng trở nên dễ dàng và nhanh chóng hơn.
- Cấu trúc component: Reactjs sử dụng cấu trúc component để phát triển ứng dụng, cho phép lập trình viên tái sử dụng các thành phần của ứng dụng và phát triển ứng dụng theo cách modulization, giúp cho việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn.

2.2.3 Các khái niệm cơ bản của Reactjs

2.2.3.1 Components và props

Components là một phần quan trọng của React và là cơ bản để xây dựng giao diện người dùng trong React. Nó được coi là các khối xây dựng của giao diện người dùng,

được tạo ra để tái sử dụng trong ứng dụng. Mỗi Component là một đối tượng độc lập, có thể có các thuộc tính (properties) và trạng thái (state) của nó, và có thể hiển thị trên trang web hoặc trên ứng dụng di động. Các Component trong React có thể được tạo ra bằng cách sử dụng hàm hoặc class. Các Component được tạo ra bằng class có thể có nhiều phương thức, chẳng hạn như phương thức render để hiển thị HTML của Component đó.

Props (viết tắt của “properties”) là một đối tượng được sử dụng để truyền dữ liệu từ một Component cha xuống một Component con. Props là bất biến (immutable), có nghĩa là giá trị của chúng không thể thay đổi bên trong Component đó. Props có thể được truyền dưới dạng tham số khi khởi tạo một Component, và được sử dụng để định nghĩa các thuộc tính của Component đó. [2]

2.2.3.2 State và lifecycle methods

State là một đối tượng JavaScript chứa các dữ liệu mà React sử dụng để hiển thị trên giao diện người dùng. Nó là một phần quan trọng của React và giúp cho việc quản lý trạng thái của ứng dụng trở nên dễ dàng hơn. Các trạng thái của ứng dụng có thể được cập nhật và thay đổi theo thời gian, và React sẽ tự động cập nhật lại giao diện người dùng tương ứng.

Lifecycle methods là các phương thức được gọi bởi React khi một component được tạo ra hoặc bị xóa khỏi giao diện người dùng. Chúng cho phép người dùng thực hiện các hoạt động nhất định tại các điểm khác nhau trong vòng đời của component. Các phương thức này có thể được sử dụng để khởi tạo dữ liệu, cập nhật dữ liệu, hoặc làm bất kỳ điều gì mà người dùng muốn trong quá trình sử dụng component. [3]

2.2.3.3 Virtual DOM

Virtual DOM là một khái niệm trong ReactJS, nó được sử dụng để quản lý DOM một cách hiệu quả và tối ưu hóa hiệu suất ứng dụng. Virtual DOM là một bản sao của DOM thật, được lưu trữ trong bộ nhớ và được cập nhật mỗi khi có sự thay đổi trong ứng dụng.

Virtual DOM hoạt động bằng cách so sánh các trạng thái trước và sau khi có sự thay đổi, tạo ra một cây DOM mới dựa trên các sự thay đổi đó và chỉ cập nhật những phần cần thiết trên trang web thực tế. Điều này giúp tối ưu hóa hiệu suất của ứng dụng, giảm thiểu việc phải tạo lại toàn bộ DOM sau mỗi lần cập nhật và tránh tình trạng xảy ra xung đột giữa các thành phần.

Virtual DOM cũng cung cấp khả năng tương thích với nhiều trình duyệt khác nhau, giúp cho ứng dụng có thể hoạt động trên nhiều nền tảng khác nhau một cách dễ dàng. Đây là một phần rất quan trọng trong cách ReactJS hoạt động. Virtual DOM là một bản sao của DOM thực tế (hay còn gọi là real DOM) được lưu trữ trong bộ nhớ và được ReactJS sử dụng để đối chiếu với DOM thực tế, từ đó cập nhật DOM một cách nhanh chóng mà không cần phải tải lại trang hoặc chuyển đổi toàn bộ cây DOM. Khi có sự thay đổi trong state hoặc props của một component, ReactJS sẽ cập nhật Virtual DOM thay vì DOM thực tế. Sau đó, ReactJS sẽ so sánh hai bản sao này và chỉ cập nhật những phần thay đổi duy nhất trong DOM thực tế, giúp tăng tốc độ hiển thị của ứng dụng. Sự khác biệt giữa Virtual DOM và DOM thực tế là Virtual DOM rất nhanh và hiệu quả trong việc cập nhật, vì nó được lưu trữ trong bộ nhớ và không phải tải lại toàn bộ trang web. Trong khi đó, DOM thực tế thì chậm hơn do phải tương tác với trình duyệt để thay đổi trên trang web. Sử dụng Virtual DOM, ReactJS có thể tăng hiệu suất của ứng dụng, giảm thời gian phản hồi và tối ưu hóa trải nghiệm người dùng.

Ví dụ, khi một user nhập thông tin vào một form, ReactJS sẽ cập nhật giá trị của state của component. Sau đó, ReactJS sử dụng Virtual DOM để so sánh sự khác biệt giữa Virtual DOM và DOM thực tế. Nếu có sự khác biệt, ReactJS chỉ cập nhật phần thay đổi trong DOM thực tế, giúp tránh tình trạng phải tải lại toàn bộ trang web hoặc chuyển đổi toàn bộ cây DOM, làm giảm thời gian phản hồi của ứng dụng và tăng hiệu suất của ứng dụng.

Tổng kết, Virtual DOM là một trong những đặc điểm quan trọng của ReactJS, giúp tăng tốc độ hiển thị của ứng dụng, giảm thời gian phản hồi và tối ưu hóa trải nghiệm người dùng. [4]

2.2.3.4 JSX syntax

JSX là một phần cú pháp (syntax) trong ReactJS cho phép lập trình viên sử dụng các thẻ HTML giống như trong việc tạo các thành phần (components) trong ứng dụng. JSX giúp đơn giản hóa quá trình viết mã cho ReactJS, vì nó cho phép lập trình viên viết các thành phần trong cú pháp HTML thay vì phải viết các đoạn mã JavaScript rườm rà hơn.

2.2.3 Môi trường và các thư viện hỗ trợ Reactjs

2.2.3.1 Cài đặt môi trường phát triển

Để xây dựng ứng dụng và trang web, người dùng cần có môi trường phát triển đầy đủ và hoạt động tốt. Dưới đây là một số bước để cài đặt môi trường phát triển cho các ứng dụng ReactJS:

- Cài đặt Node.js: Đầu tiên, người dùng cần phải cài đặt Node.js, một nền tảng cho phép chạy JavaScript ở phía máy chủ và quản lý các gói phụ thuộc cho ứng dụng.
- Cài đặt trình quản lý gói npm: npm là một trình quản lý gói phần mềm được sử dụng để quản lý các phụ thuộc của ứng dụng. Nó được cài đặt cùng với Node.js
- Tạo ứng dụng React với Create React App: Create React App là một công cụ giúp người dùng tạo ra các ứng dụng React đơn giản một cách nhanh chóng.
- Cài đặt trình biên dịch Babel: Babel là một trình biên dịch JavaScript giúp người dùng biên dịch các mã nguồn JavaScript hiện đại sang các mã nguồn tương thích với các trình duyệt cũ hơn.
- Cài đặt trình đóng gói Webpack: Webpack là một công cụ đóng gói và tối ưu hóa mã nguồn của ứng dụng.

2.2.3.2 Một số thư viện hỗ trợ Reactjs

- Redux là một thư viện quản lý trạng thái của ứng dụng cho các ứng dụng ReactJS. Thư viện này giúp cho việc quản lý dữ liệu trở nên dễ dàng và có tổ chức hơn, đồng thời cũng giúp cho việc truyền dữ liệu giữa các thành phần

trong ứng dụng trở nên thuận tiện hơn. Redux hoạt động dựa trên một kiến trúc Flux, với ba thành phần chính là Actions, Reducers và Store:

- Actions: là các đối tượng mô tả một hành động sẽ xảy ra trong ứng dụng, ví dụ như “thêm một sản phẩm vào giỏ hàng” hoặc “đăng nhập vào hệ thống”. Mỗi action bao gồm một loại (type) và một số thông tin liên quan (payload).
- Reducers: là các hàm xử lý trạng thái (state) của ứng dụng dựa trên các action tương ứng. Mỗi reducer nhận vào trạng thái hiện tại của ứng dụng và một action, sau đó trả về trạng thái mới được cập nhật dựa trên action đó.
- Store: là nơi lưu trữ trạng thái của toàn bộ ứng dụng, được quản lý bởi Redux. Mỗi khi có action được gửi đến, store sẽ gọi tất cả các reducer để cập nhật trạng thái mới cho ứng dụng.
- React Router là một thư viện quan trọng và được sử dụng rộng rãi để quản lý định tuyến trên ứng dụng React. Thư viện này được thiết kế để giúp các nhà phát triển tạo các ứng dụng đơn trang một cách dễ dàng và thuận tiện. Một trong những tính năng quan trọng của React Router là quản lý lịch sử duyệt web (browser history) trong ứng dụng React. Khi người dùng thực hiện hành động như bấm nút back hoặc forward trên trình duyệt web, React Router sẽ tự động cập nhật trạng thái của ứng dụng để phù hợp với lịch sử duyệt web. React Router cung cấp ba thành phần chính: BrowserRouter, Route và Link:
 - BrowserRouter: Đây là thành phần chính của React Router, nó cung cấp API để đồng bộ hóa trạng thái ứng dụng với lịch sử trình duyệt. Nó sẽ theo dõi trạng thái duyệt web và cập nhật lại giao diện người dùng khi có thay đổi.
 - Route: Thành phần này giúp định nghĩa các tuyến đường (routes) cho ứng dụng. Khi URL của ứng dụng được thay đổi, React Router sẽ hiển thị component tương ứng với URL đó. Route cũng cho phép xác định những tham số động trên URL, giúp cho việc truy cập và quản lý dữ liệu trên trang trở nên dễ dàng hơn.

- Link: Thành phần này cho phép tạo các liên kết giữa các trang trong ứng dụng. Khi người dùng bấm vào một liên kết, React Router sẽ tự động cập nhật URL của trình duyệt và hiển thị component tương ứng với URL đó.
- Nested Routing: Cho phép định nghĩa các tuyến đường lồng nhau, giúp cho việc quản lý ứng dụng trở nên dễ dàng hơn.
- Route Guards: Cung cấp khả năng bảo vệ các tuyến đường khỏi truy cập trái phép, giúp cho ứng dụng.
- Material-UI là một thư viện UI cho ứng dụng React, được thiết kế để giúp tạo ra các giao diện đẹp và đáp ứng nhanh chóng. Thư viện này sử dụng các thành phần được thiết kế theo hướng vật liệu (Material Design) của Google, giúp tạo ra các giao diện màu sắc đẹp mắt và tương tác người dùng dễ dàng. Material-UI cung cấp nhiều thành phần giao diện khác nhau, từ các nút đơn giản, bảng dữ liệu cho đến các hộp thoại thông báo. Các thành phần này đã được thiết kế sẵn và có thể sử dụng trực tiếp trong ứng dụng, giúp tiết kiệm thời gian và công sức thiết kế giao diện. Một số thành phần chính của Material-UI bao gồm:
 - Button: Cho phép tạo các nút và các nút tương tự trong ứng dụng.
 - Checkbox: Cho phép tạo các ô chọn trong ứng dụng.
 - TextField: Cho phép tạo các trường văn bản trong ứng dụng.
 - Table: Cho phép hiển thị dữ liệu dưới dạng bảng.
 - Modal: Cho phép tạo các hộp thoại thông báo hoặc cửa sổ hiển thị dữ liệu.

Material-UI cũng cung cấp một hệ thống chữ (Typography) để giúp tạo ra các đoạn văn bản và tiêu đề đẹp mắt và dễ đọc. Thư viện này cũng có khả năng tùy chỉnh cao, cho phép người dùng tùy chỉnh các thuộc tính của các thành phần để phù hợp với thiết kế. Ngoài ra, Material-UI còn hỗ trợ các tính năng như responsive design, RTL support, theme customization và cung cấp các hàm hỗ trợ để giúp quản lý trạng thái và tương tác người dùng trong ứng dụng. Tóm lại, Material-UI là một thư viện UI đầy đủ và mạnh mẽ, giúp tạo ra các giao diện đẹp mắt và dễ sử dụng trong ứng dụng React.

- Axios là một thư viện HTTP client được sử dụng trong ứng dụng web để gửi và nhận các yêu cầu HTTP với server. Nó cung cấp cho chúng ta một API đơn giản và dễ sử dụng để tương tác với các API RESTful và các API HTTP khác. Với Axios, chúng ta có thể thực hiện các yêu cầu HTTP GET, POST, PUT, DELETE, PATCH và HEAD. Nó cũng hỗ trợ các tính năng như cancel request, interceptors, authentication và XSRF protection. Một số tính năng của Axios:
 - Chỉ định cấu hình mặc định cho tất cả các yêu cầu.
 - Tự động chuyển đổi các định dạng dữ liệu như JSON, XML, FormData, và các định dạng tệp.
 - Hỗ trợ cả Promise và async/await để xử lý các yêu cầu bất đồng bộ.
 - Hỗ trợ cả browser và Node.js.
 - Hỗ trợ các interceptors để chuyển đổi các yêu cầu và phản hồi.
 - Hỗ trợ cancel request để dễ dàng quản lý các yêu cầu không cần thiết.

Để sử dụng Axios trong ứng dụng React, chúng ta có thể import nó vào trong component và sử dụng các phương thức như `axios.get()`, `axios.post()`, `axios.put()`, `axios.delete()`, và `axios.patch()` để gửi các yêu cầu HTTP đến server.

- Testing Library là một thư viện JavaScript giúp chúng ta thực hiện kiểm thử giao diện người dùng của ứng dụng web. Nó được thiết kế để tạo ra các kiểm thử có ý nghĩa và dễ đọc, đồng thời tập trung vào hành vi người dùng thay vì chi tiết cài đặt của ứng dụng. Testing Library cung cấp một bộ công cụ cho phép chúng ta tương tác với các thành phần UI của ứng dụng, giống như người dùng thực tế. Các phương thức của Testing Library như `getByLabelText()`, `getByText()`, `getByTestId()`, và `getByRole()` cho phép chúng ta tìm kiếm các thành phần UI bằng cách sử dụng các tiêu chí như nhãn, văn bản, test ID, và vai trò. Sau đó, chúng ta có thể tương tác với các thành phần này bằng cách sử dụng các phương thức như `click()`, `change()`, `submit()`, và `type()`. Testing Library cũng tập trung vào việc kiểm tra hành vi người dùng, chứ không phải kiểm tra các chi tiết cài đặt của ứng dụng. Ví dụ, thay vì kiểm tra xem một

thành phần đã render với một class CSS cụ thể, chúng ta có thể kiểm tra xem thành phần đó đã render đúng với các nội dung mong đợi hoặc đã kích hoạt các hành vi đúng trong ứng dụng. Testing Library được sử dụng rộng rãi trong cộng đồng React để kiểm thử các ứng dụng React. Nó là một phần của thư viện React Testing Library, một bộ công cụ kiểm thử được thiết kế đặc biệt cho React. Testing Library được sử dụng bởi các công ty lớn như Facebook, Airbnb, và Shopify để kiểm thử các ứng dụng web của họ.

2.2.4 Các kỹ thuật lập trình trong Reactjs

2.2.4.1 React Hooks

React Hooks là một tính năng mới được giới thiệu trong React 16.8, giúp cho việc quản lý state và lifecycle trong function components dễ dàng hơn. Các hooks là những hàm có sẵn được cung cấp bởi React, giúp chúng ta quản lý state và lifecycle logic của component một cách dễ dàng hơn. Dưới đây là một số hooks phổ biến trong React:

- `useState()`: `useState()` được sử dụng để quản lý state trong function components. Hook này nhận vào một giá trị khởi tạo cho state và trả về một array gồm hai phần tử: Giá trị hiện tại của state và một hàm để cập nhật giá trị state mới.
- `useEffect()`: `useEffect()` được sử dụng để thực hiện các side effect trong function components. Side effect là các tác động không đáng có trên giao diện, chẳng hạn như gọi API, thay đổi title của trang, và nhiều hơn nữa. Hook này nhận vào một callback function và chạy nó sau mỗi lần render component.
- `useContext()`: `useContext()` được sử dụng để truy cập vào context trong function components. Context là một cách để chia sẻ state giữa các component trong ứng dụng. Hook này nhận vào một context object và trả về giá trị của context đó.
- `useCallback()`: `useCallback()` được sử dụng để giảm thiểu việc render lại các child components trong một component cha. Hook này nhận vào một callback và một mảng dependencies và trả về một callback mới được memoize (lưu lại

kết quả tính toán của callback trước đó), giúp tránh việc render lại các child components nếu các dependencies không thay đổi.

2.2.4.2 Higher-order components

Higher-order components (HOC) là một kỹ thuật lập trình React cho phép tái sử dụng logic giữa các components khác nhau. Kỹ thuật này là một function nhận vào một component và trả về một component mới có tính năng mới. HOC giúp cho việc tái sử dụng logic trở nên dễ dàng hơn bằng cách cho phép chia sẻ logic giữa các components không cùng cha mẹ hoặc các components khác nhau. HOC có thể được sử dụng để tái sử dụng logic trong các tình huống sau:

- Logic state management: HOC có thể được sử dụng để quản lý state và lifecycle logic của các components. Ví dụ, chúng ta có thể tạo một HOC để đếm số lần click của một component, hoặc để truyền một vài props chung cho các components khác nhau.
- Authentication: HOC có thể được sử dụng để kiểm tra quyền truy cập của người dùng, hoặc để xác định liệu người dùng đã đăng nhập hay chưa.
- Code reuse: HOC cũng có thể được sử dụng để tái sử dụng code giữa các components khác nhau.

Tuy nhiên, khi sử dụng HOC, chúng ta cần lưu ý một số điểm sau đây:

- Code reuse: HOC cũng có thể được sử dụng để tái sử dụng code giữa các components khác nhau.
- Chúng ta cần sử dụng HOC một cách cẩn thận để tránh trở thành việc dư thừa hoặc không cần thiết. Việc sử dụng quá nhiều HOC có thể làm cho code của chúng ta trở nên phức tạp và khó hiểu.
- HOC không nên được sử dụng như một phương tiện để giải quyết các vấn đề về hiệu năng của ứng dụng. Thay vào đó, chúng ta nên sử dụng các phương tiện khác như React.memo hoặc PureComponent để giải quyết các vấn đề liên quan đến hiệu năng.

- Khi sử dụng HOC, chúng ta cần chú ý đến việc truyền các props vào các components để tránh xung đột và lỗi trong quá trình render. Chúng ta nên sử dụng các props có tên khác nhau hoặc sử dụng các giá trị mặc định để tránh xung đột.

2.2.4.3 Render props

Render props là một kỹ thuật lập trình trong ReactJS cho phép chia sẻ code giữa các components. Kỹ thuật này cho phép truyền một function như là một prop vào trong một component và function này sẽ trả về một component khác.

Cách hoạt động của Render props là tạo một function trong component cha, sau đó truyền function này như là một prop vào trong component con. Trong phần render của component cha, ta gọi function được truyền vào và truyền các giá trị cần thiết vào function này để trả về một component con. Kết quả của function này được render trong component cha.

Render props là một kỹ thuật lập trình mạnh mẽ cho phép tái sử dụng code giữa các components khác nhau. Nó giúp tạo ra các components có tính tái sử dụng cao và giảm thiểu sự trùng lặp code. Tuy nhiên, khi sử dụng Render props, cần lưu ý để tránh xung đột và lỗi trong quá trình render.

2.2.3.5 Context API

Context API là một trong những kỹ thuật lập trình của ReactJS, cho phép chúng ta chia sẻ dữ liệu giữa các component một cách dễ dàng. Thay vì truyền props từ component cha xuống các component con, Context API cho phép chúng ta truyền dữ liệu từ component cha xuống bất kỳ component con nào trong cây component một cách nhanh chóng. Context API hoạt động bằng cách tạo ra một context object trong component cha, lưu trữ dữ liệu mà chúng ta muốn chia sẻ. Context object này được truyền xuống các component con bằng cách sử dụng Provider component. Provider component sẽ cung cấp dữ liệu cho các component con, giúp chúng ta không cần phải truyền dữ liệu qua nhiều level của cây component. Một số ưu điểm của Context API trong ReactJS:

- Giúp chia sẻ dữ liệu giữa các component một cách dễ dàng và hiệu quả.
- Không cần sử dụng props để truyền dữ liệu qua lại giữa các component, giúp giảm bớt sự phức tạp của code.
- Cho phép sử dụng Provider và Consumer component hoặc hook useContext() để truy cập vào giá trị của context object.
- Cho phép sử dụng nhiều Provider component để chia sẻ nhiều giá trị khác nhau.
- Giúp tăng tính tái sử dụng của component, vì chúng ta có thể sử dụng cùng một context object trong nhiều component khác nhau.

2.2.5 Các vấn đề thường gặp khi sử dụng Reactjs

2.2.5.1 Một số lỗi thường gặp trong reactjs

- “Cannot read property of undefined”: Lỗi này thường xảy ra khi chúng ta cố gắng truy cập thuộc tính của một đối tượng không tồn tại hoặc không được khởi tạo.
- “Rendered more hooks than during the previous render”: Lỗi này xảy ra khi chúng ta sử dụng quá nhiều React hooks trong một component.
- “Maximum update depth exceeded”: Lỗi này thường xảy ra khi chúng ta gọi hàm setState() trong một vòng lặp vô hạn.
- “Invariant Violation: Minified React error #31”: Lỗi này xảy ra khi chúng ta sử dụng React DOM và React Native cùng một lúc.
- “Unexpected token < in JSON at position 0”: Lỗi này xảy ra khi chúng ta cố gắng parse một JSON response từ server nhưng chuỗi JSON đó không hợp lệ.
- “TypeError: Cannot read property ‘map’ of undefined”: Lỗi này thường xảy ra khi chúng ta sử dụng phương thức map() của một mảng không được khởi tạo hoặc không tồn tại.
- “Warning: Each child in a list should have a unique ‘key’ prop”: Cảnh báo này thường xảy ra khi chúng ta sử dụng một danh sách các components trong React mà không cung cấp thuộc tính key.

- “Warning: Failed prop type: Invalid prop propName of type propTypes supplied to componentName, expected expectedType”: Cảnh báo này xảy ra khi chúng ta không định nghĩa kiểu dữ liệu cho props hoặc truyền sai kiểu dữ liệu cho props trong một component.

2.5.2 Cách khắc phục một số lỗi thường gặp

- “Cannot read property of undefined”: Kiểm tra xem đối tượng đó đã được khởi tạo hay chưa, nếu chưa thì hãy khởi tạo nó trước khi sử dụng. Nếu đối tượng đã tồn tại, hãy đảm bảo rằng thuộc tính đó cũng tồn tại và được định nghĩa.
- “Rendered more hooks than during the previous render”: Kiểm tra xem chúng ta đã sử dụng quá nhiều React hooks trong một component chưa. Nếu đúng vậy, hãy giảm số lượng hooks hoặc chia nhỏ component.
- “Maximum update depth exceeded”: Kiểm tra xem chúng ta có gọi setState() trong một vòng lặp vô hạn không. Nếu đúng vậy, hãy thay đổi cách sử dụng setState() để tránh vòng lặp vô hạn.
- “Invariant Violation: Minified React error #31”: Đảm bảo rằng chúng ta chỉ sử dụng React DOM hoặc React Native trong một project duy nhất.
- “Unexpected token < in JSON at position 0”: Kiểm tra xem chuỗi JSON đó có đúng định dạng hay không. Nếu không, hãy xử lý chuỗi đó trước khi parse nó.
- “TypeError: Cannot read property ‘map’ of undefined”: Kiểm tra xem mảng đó đã được khởi tạo hay chưa. Nếu chưa, hãy khởi tạo mảng đó trước khi sử dụng phương thức map(). Nếu đã khởi tạo mảng nhưng vẫn bị lỗi, hãy đảm bảo rằng mảng đó có giá trị null hoặc undefined.
- “Warning: Each child in a list should have a unique ‘key’ prop”: Đảm bảo rằng mỗi component trong danh sách đều có thuộc tính key duy nhất để React có thể xác định được component nào đã thay đổi.
- “Warning: Failed prop type: Invalid prop propName of type propTypes supplied to componentName, expected expectedType”: Đảm bảo rằng kiểu dữ liệu của props được định nghĩa đúng cách trong component. Kiểm tra lại xem các kiểu

dữ liệu của props đang được truyền vào có phù hợp với định nghĩa trong component hay không. Nếu không, hãy sửa lại kiểu dữ liệu của props.

2.5.3 Debugging và logging trong ReactJS

Sử dụng trình duyệt DevTools: Trình duyệt DevTools được tích hợp sẵn trong hầu hết các trình duyệt hiện nay, và cho phép chúng ta xem và chỉnh sửa các thành phần của ứng dụng trong quá trình chạy. DevTools cũng cung cấp các công cụ để debug và log các lỗi trong ứng dụng. Sử dụng React Developer Tools: React Developer Tools là một công cụ mở rộng cho trình duyệt Chrome và Firefox, được phát triển bởi Facebook. Công cụ này cho phép chúng ta xem các component của ứng dụng và các props của chúng trong quá trình chạy, giúp giải quyết các lỗi liên quan đến ReactJS.

2.3 TailwindCSS

2.3.1 Giới thiệu

2.3.1.1 Tổng quan về TailwindCSS

Tailwind CSS là một thư viện CSS mã nguồn mở phát triển bởi Adam Wathan, Steve Schoger, và Jonathan Reinink, được thiết kế để tăng tốc độ phát triển của các dự án web bằng cách cung cấp các class CSS tiện ích để tạo ra các giao diện đáp ứng và tùy chỉnh dễ dàng. So với các phương pháp thiết kế giao diện truyền thống sử dụng tệp CSS riêng biệt, Tailwind CSS sử dụng một phương pháp thiết kế giao diện dựa trên các lớp CSS tiện ích, cho phép người dùng thiết kế các giao diện mà không cần viết CSS tùy chỉnh. Thay vào đó, người dùng sẽ sử dụng các lớp CSS được đặt tên mô tả chức năng của chúng để tạo ra các giao diện phức tạp một cách nhanh chóng và dễ dàng. Với Tailwind CSS, người dùng không cần phải suy nghĩ về kiểu dáng và bố cục của các phần tử HTML như trước đây. Thay vào đó, người dùng tập trung vào các chức năng và tính năng của giao diện của mình và sử dụng các lớp CSS tiện ích để tạo ra các đối tượng giao diện thích hợp. Mỗi lớp CSS tiện ích trong Tailwind CSS có một tên mô tả chức năng của nó, giúp người dùng dễ dàng tìm và sử dụng chúng trong dự án của mình. Ví dụ, nếu người dùng muốn thêm màu nền cho một phần tử HTML, người dùng có thể sử dụng lớp bg-[tên-màu] (ví dụ: bg-red-500) để thêm màu đỏ cho phần tử đó. Nếu người dùng muốn thêm một đường viền cho phần tử đó, người

dùng có thể sử dụng lớp `border-[tên-màu]` (ví dụ: `border-gray-400`) để thêm đường viền màu xám cho phần tử đó.

Tóm lại, Tailwind CSS là một công cụ hữu ích cho các nhà phát triển web để thiết kế giao diện nhanh chóng và dễ dàng, giúp tiết kiệm thời gian và tăng năng suất trong quá trình phát triển.

2.3.1.2 Lịch sử của TailwindCSS

Tailwind CSS được phát triển bởi Adam Wathan vào năm 2017. Adam Wathan là một nhà phát triển web độc lập và tác giả sách về lập trình, đã có nhiều kinh nghiệm trong việc xây dựng các dự án web cho các khách hàng khác nhau.

Ý tưởng ban đầu của Tailwind CSS xuất phát từ việc phát triển một thư viện CSS cho một dự án riêng của Adam. Adam cảm thấy rằng việc sử dụng các lớp CSS tiện ích là một cách tốt để giúp tăng tốc độ phát triển của các dự án web bằng cách giảm thiểu việc phải viết CSS tùy chỉnh. Tuy nhiên, việc tạo ra các lớp CSS tiện ích thủ công cho mỗi dự án của mình là một công việc đòi hỏi rất nhiều thời gian và công sức. Vì vậy, Adam đã quyết định tạo ra một thư viện CSS cho phép các nhà phát triển web sử dụng các lớp CSS tiện ích đã được định nghĩa trước để tạo ra các giao diện nhanh chóng và dễ dàng. Adam đã chia sẻ thư viện của mình trên Github và nhận được sự quan tâm của nhiều nhà phát triển web khác. Với sự quan tâm và yêu cầu của cộng đồng, Tailwind CSS nhanh chóng trở thành một thư viện CSS được sử dụng rộng rãi trên toàn thế giới. Tailwind CSS cung cấp cho các nhà phát triển web một cách tiếp cận mới và tiện lợi trong việc xây dựng các giao diện đáp ứng và tùy chỉnh.

2.3.1.3 Lý do nên dùng TailwindCSS

Tailwind CSS đang nổi lên là một trong những ứng viên hàng đầu cho sự lựa chọn để xây dựng front-end cho website. Dưới đây là các lý do vì sao nên dùng Tailwindcss:

- **Tăng tốc độ phát triển:** Tailwind CSS cung cấp các class CSS tiện ích cho phép người dùng tạo ra các giao diện đáp ứng và tùy chỉnh nhanh chóng hơn so với việc phải viết CSS tùy chỉnh.

- Dễ dàng tùy chỉnh: Các class CSS của Tailwind CSS được đặt tên mô tả chức năng của chúng, cho phép người dùng dễ dàng tìm và tùy chỉnh chúng trong trường hợp cần thiết.
- Khả năng tái sử dụng: Các class CSS của Tailwind CSS có thể được tái sử dụng trong nhiều dự án khác nhau, giúp tiết kiệm thời gian và công sức.
- Giảm thiểu lỗi: Sử dụng các class CSS của Tailwind CSS giúp giảm thiểu lỗi trong quá trình phát triển và bảo trì, vì các lớp CSS được đặt tên rõ ràng và mô tả chức năng của chúng.
- Tiết kiệm thời gian: Tailwind CSS giúp tiết kiệm thời gian cho các nhà phát triển bằng cách giảm thiểu việc phải viết CSS tùy chỉnh và cho phép tập trung vào việc tạo nội dung và tích hợp chức năng.
- Hỗ trợ đa nền tảng: Tailwind CSS hỗ trợ đa nền tảng và có thể sử dụng trong các dự án web trên nhiều trình duyệt và thiết bị khác nhau.
- Dễ dàng tích hợp: Tailwind CSS có thể dễ dàng tích hợp với các công cụ khác như React, Vue, Angular và các framework khác.

Với những lý do trên, Tailwind CSS trở thành một lựa chọn phổ biến cho các nhà phát triển web muốn tăng tốc độ phát triển và giảm thiểu lỗi trong quá trình phát triển và bảo trì các dự án web. [5]

2.3.1.4 Các tính năng của TailwindCSS

TailwindCSS cung cấp 1 số tính năng như sau:

- Class CSS tiện ích: Tailwind CSS cung cấp hơn 2000 class CSS tiện ích, cho phép người dùng tạo ra các giao diện đáp ứng và tùy chỉnh nhanh chóng hơn so với việc phải viết CSS tùy chỉnh.
- Thiết kế đáp ứng: Tailwind CSS được thiết kế để hỗ trợ thiết kế đáp ứng. Các class CSS tiện ích cho phép người dùng dễ dàng kiểm soát bố cục, độ rộng, độ cao, padding, margin, font-size, font-weight, color, background-color và nhiều thuộc tính CSS khác để tạo ra các giao diện đáp ứng.

- Thư viện màu sắc: Tailwind CSS cung cấp một thư viện màu sắc phong phú, cho phép người dùng sử dụng các màu sắc đẹp và tương phản với dễ dàng.
- Customization: Tailwind CSS cho phép người dùng tùy chỉnh và thay đổi các class CSS mặc định, cũng như thêm các class CSS tùy chỉnh của riêng người dùng. Điều này cho phép người dùng tạo ra các giao diện độc đáo và phù hợp với phong cách .
- Dark mode: Tailwind CSS hỗ trợ chế độ tối (dark mode) bằng cách sử dụng các class CSS tiện ích được đặt tên riêng cho chế độ tối.
- Thư viện định dạng văn bản: Tailwind CSS cung cấp các class CSS tiện ích để định dạng văn bản, cho phép người dùng tạo ra các tiêu đề, đoạn văn bản, danh sách và nhiều loại khác của văn bản trong giao diện .
- Thư viện biểu tượng: Tailwind CSS cung cấp một thư viện biểu tượng phong phú, cho phép người dùng sử dụng các biểu tượng đẹp và phù hợp với phong cách.
- Thư viện đường viền (border): Tailwind CSS cung cấp các class CSS tiện ích để định dạng đường viền, cho phép người dùng tạo ra các đường viền đơn giản hoặc phức tạp trong giao diện.
- Thư viện hiệu ứng: Tailwind CSS cung cấp các class CSS tiện ích để tạo ra các hiệu ứng động trong giao diện, bao gồm các hiệu ứng hover và focus.

Ngoài ra còn một số các tính năng khác:

- Plugins: Tailwind CSS cho phép người dùng sử dụng các plugins để mở rộng chức năng của nó, bao gồm các plugin cho các framework như React, Vue và Angular.
- Dark mode: Tailwind CSS hỗ trợ chế độ tối (dark mode) với các class CSS tiện ích để thay đổi các giá trị màu sắc, nền và các thuộc tính khác.
- Tích hợp dễ dàng: Tailwind CSS tích hợp dễ dàng với các công cụ như Webpack, Gulp và PostCSS, giúp đơn giản hóa quá trình phát triển.
- Hỗ trợ tùy chỉnh: Tailwind CSS cho phép người dùng tùy chỉnh các giá trị mặc định của nó bằng cách chỉnh sửa tệp cấu hình và thay đổi các biến CSS.

- Đi kèm với tài liệu hướng dẫn chi tiết: Tailwind CSS cung cấp một tài liệu hướng dẫn chi tiết và đầy đủ, giúp cho việc sử dụng và tùy chỉnh thư viện trở nên dễ dàng hơn. [6]

2.3.2 Các khái niệm cơ bản trong TailwindCSS

2.3.2.1 CSS utility classes

CSS utility classes là các lớp CSS được thiết kế để áp dụng các kiểu dáng cụ thể cho các phần tử HTML. Các lớp CSS này thường có tên gọi ngắn gọn và thể hiện tính năng hoặc kiểu dáng được áp dụng.

Việc sử dụng các lớp CSS tiện ích giúp chúng ta viết CSS nhanh hơn và giảm thiểu số lượng CSS cần được viết và bảo trì. Tailwind CSS cung cấp rất nhiều lớp CSS tiện ích khác nhau, giúp người dùng tạo ra các kiểu dáng phong phú và đa dạng cho trang web của mình.

Tuy nhiên, việc sử dụng CSS utility classes cũng có nhược điểm của nó. Việc đặt tên cho các lớp CSS tiện ích thường khá khó hiểu và dài, có thể gây khó khăn cho việc bảo trì và cập nhật sau này. Ngoài ra, việc sử dụng các lớp CSS tiện ích cũng có thể gây ra sự trùng lặp và đôi khi dẫn đến CSS không cần thiết, do một số lớp CSS không được sử dụng trong trang web. [7]

2.3.2.2 Configuration file

Configuration file là một tập tin cấu hình JSON hoặc JavaScript được sử dụng để cấu hình và tùy chỉnh các thiết lập của Tailwind CSS. Trong tệp cấu hình này, người dùng có thể chỉ định các giá trị màu, font, padding, margin, width, height, breakpoint, v.v. để sử dụng cho các lớp CSS tiện ích trong Tailwind CSS. Tệp cấu hình được sử dụng để định nghĩa các lớp CSS tiện ích mặc định của Tailwind CSS, cũng như để thiết lập các giá trị tùy chỉnh cho các tính năng khác của Tailwind CSS. Nó cho phép người dùng tùy chỉnh các giá trị mặc định và định nghĩa các lớp CSS tiện ích tùy chỉnh dựa trên các thiết lập được cung cấp. Tệp cấu hình của Tailwind CSS có thể được định nghĩa trong nhiều định dạng, bao gồm JSON, JavaScript, và YAML. Tùy thuộc vào định dạng, người dùng có thể sử dụng các khai báo và thiết lập khác nhau để định

nghĩa các giá trị trong tệp cấu hình. Một số ví dụ về thiết lập cấu hình trong Tailwind CSS bao gồm:

- Thiết lập các màu sắc tùy chỉnh cho các lớp CSS tiện ích
- Thiết lập các kích thước phông chữ, margin, padding, v.v. cho các lớp CSS tiện ích
- Thiết lập các breakpoint tùy chỉnh để sử dụng cho responsive design
- Định nghĩa các lớp CSS tiện ích tùy chỉnh dựa trên các giá trị cấu hình đã thiết lập.

2.3.2.3 Responsive design

Responsive design là một phương pháp thiết kế web nhằm đảm bảo rằng trang web sẽ tương thích và hiển thị tốt trên nhiều loại thiết bị khác nhau, từ máy tính để bàn đến điện thoại di động. Nó là một phương pháp quan trọng để cải thiện trải nghiệm người dùng và đảm bảo rằng trang web sẽ đáp ứng được nhu cầu của đa dạng người dùng. Tailwind CSS cung cấp nhiều lớp CSS được thiết kế để hỗ trợ responsive design. Một số lớp phổ biến trong Tailwind CSS bao gồm:

- Lớp **sm**, **md**, **lg**, **xl**, và **2xl**: Đây là các lớp breakpoint để xác định kích thước của các thiết bị khác nhau. Người dùng có thể sử dụng chúng để thiết kế các trang web phù hợp với kích thước màn hình từ nhỏ đến lớn.
- Lớp **w-**, **h-**, và **max-w-**: Đây là các lớp để thiết lập kích thước cho các phần tử HTML như hình ảnh, video, đường viền, v.v. Người dùng có thể sử dụng chúng để điều chỉnh kích thước các phần tử sao cho phù hợp với các thiết bị khác nhau.
- Lớp **flex**, **flex-col**, **justify-center**, **items-center**: Đây là các lớp để thiết lập định dạng và căn chỉnh các phần tử HTML. Người dùng có thể sử dụng chúng để điều chỉnh vị trí các phần tử trên trang web, để chúng hiển thị tốt trên các thiết bị khác nhau.

Tailwind CSS cũng hỗ trợ các tính năng khác để hỗ trợ responsive design, chẳng hạn như các breakpoint tùy chỉnh và lớp CSS tiện ích để hiển thị hoặc ẩn các phần tử trên

các thiết bị khác nhau. Nhờ vào các lớp CSS và tính năng này, Tailwind CSS cho phép người dùng thiết kế các trang web phản hồi đẹp mắt và đáp ứng nhu cầu của nhiều loại thiết bị khác nhau.

2.3.2.4 Utility-first approach

Phương pháp Utility-first approach là một phương pháp thiết kế CSS tập trung vào việc tạo ra một bộ sưu tập lớp CSS tiện ích (utility classes), tức là các lớp CSS đã được định nghĩa trước và thường được sử dụng lại nhiều lần để thực hiện các tính năng cụ thể. Thay vì tạo ra các tệp CSS lớn với nhiều luật CSS được định nghĩa trước và sử dụng tên lớp tùy chỉnh để áp dụng các kiểu cho các phần tử HTML, ta sẽ sử dụng các lớp CSS tiện ích được định nghĩa sẵn để áp dụng các kiểu cho các phần tử HTML. Tailwind CSS cung cấp cho người dùng một số lượng lớn các lớp CSS tiện ích được định nghĩa sẵn, cho phép người dùng áp dụng các kiểu dáng và bố cục cho các phần tử HTML một cách nhanh chóng và dễ dàng. Người dùng có thể sử dụng các lớp này để thiết lập kích thước, độ rộng, độ cao, độ dày, màu sắc, độ trong suốt, độ trôi chảy, khoảng cách và vị trí của các phần tử HTML một cách nhanh chóng và hiệu quả. Phương pháp Utility-first approach giúp người dùng giảm thiểu thời gian và công sức cần thiết để viết CSS tùy chỉnh bằng cách sử dụng các lớp CSS tiện ích được định nghĩa sẵn. Nó cũng giúp người dùng tối ưu hóa kích thước tệp CSS và tăng tốc độ tải trang của trang web.

2.3.2.5 Predefined color palette

Predefined color palette (bảng màu được định nghĩa trước) là một tập hợp các màu được xác định trước và được sử dụng trong thiết kế hoặc ứng dụng. Với Tailwind CSS, người dùng có thể sử dụng các bảng màu được định nghĩa trước để dễ dàng chọn và sử dụng màu sắc trong các phần tử của trang web. Tailwind CSS cung cấp nhiều bảng màu khác nhau, bao gồm:

- Grayscale: bảng màu đơn sắc với các màu đen trắng và xám.
- Red, yellow, green, blue: các bảng màu đơn sắc với các màu đỏ, vàng, xanh lá cây và xanh dương.
- Indigo: bảng màu đơn sắc với các màu tím đậm.

- Purple: bảng màu đơn sắc với các màu tím nhạt.
- Pink: bảng màu đơn sắc với các màu hồng.

Mỗi bảng màu bao gồm các biến màu tương ứng, ví dụ như gray-200, red-500, indigo-900,... Các biến này có thể được sử dụng trong CSS để đặt màu cho các phần tử trên trang web. Bên cạnh các bảng màu được định nghĩa trước, cũng có thể tạo các bảng màu tùy chỉnh của riêng mình trong tệp cấu hình của Tailwind CSS.

2.3.2.6 Customization

Customization: là quá trình tùy chỉnh hoặc cá nhân hóa thiết kế hoặc ứng dụng để phù hợp với nhu cầu hoặc sở thích của người dùng. Tailwind CSS cho phép người dùng tùy chỉnh và điều chỉnh các thiết lập để tạo ra trang web hoặc ứng dụng độc đáo và tùy chỉnh. Tailwind CSS cung cấp cho người dùng một tập tin cấu hình được gọi là `tailwind.config.js`, cho phép người dùng tùy chỉnh các thiết lập của Tailwind CSS để phù hợp với nhu cầu.

2.3.3 Các tính năng chính trong TailwindCSS

2.3.3.1 Layout

Layout là một trong những tính năng quan trọng nhất trong Tailwind CSS, giúp xây dựng các bố cục trang web linh hoạt và chính xác. Tailwind cung cấp một hệ thống grid đáp ứng với nhiều tùy chọn cấu hình để có thể tạo ra các bố cục theo ý muốn của mình. Hệ thống grid của Tailwind CSS được xây dựng dựa trên CSS Grid và được tích hợp sẵn trong các lớp CSS của framework. Người dùng có thể sử dụng các lớp CSS để xác định các cột và hàng của một lưới, sắp xếp các phần tử trên trang web vào các ô trong lưới. Ngoài ra, Tailwind còn cung cấp các lớp CSS để tạo ra các bố cục linh hoạt khác, chẳng hạn như chia sẻ các phần tử thành các cột bằng cách sử dụng Flexbox hoặc tạo ra các trình đơn đa cấp. Nếu người dùng muốn tạo ra các bố cục phức tạp hơn, Tailwind cung cấp một hệ thống các lớp CSS để điều chỉnh vị trí, độ rộng, độ cao, khoảng cách, và các tính năng khác của các phần tử trên trang web. Với tính năng layout của Tailwind, có thể thiết kế các trang web đẹp và chuyên nghiệp với các bố cục linh hoạt và chính xác, dễ dàng điều chỉnh và tùy biến để phù hợp với mục đích.

2.3.3.2 Typography

Là một tính năng quan trọng trong Tailwind CSS, Typography giúp người dùng xây dựng các kiểu chữ đẹp và chính xác trên trang web của mình. Tailwind cung cấp một hệ thống lớp CSS để tùy chỉnh front chữ, kích thước, khoảng cách, màu sắc và các tính năng khác của văn bản trên trang web. Một trong những đặc điểm nổi bật của tính năng Typography trong Tailwind là hệ thống phân cấp các lớp CSS, giúp người dùng dễ dàng xác định kiểu chữ, cỡ chữ và màu sắc của văn bản trên trang web. Người dùng có thể sử dụng các lớp CSS để định dạng các phần tử văn bản, chẳng hạn như tiêu đề, đoạn văn bản, liên kết và nhiều hơn nữa. Tailwind cũng cung cấp một số lớp CSS để tạo kiểu chữ linh hoạt hơn, chẳng hạn như định dạng văn bản in đậm, in nghiêng, gạch chân và gạch xuyên giữa. Người dùng cũng có thể sử dụng các lớp CSS để tạo ra các hiệu ứng chữ khác, chẳng hạn như đổ bóng, độ trong suốt và 3D. Với tính năng Typography của Tailwind, người dùng có thể tạo ra các kiểu chữ đẹp và chính xác trên trang web của mình, dễ dàng tùy chỉnh và tùy biến để phù hợp với mục đích của người dùng.

2.3.3.3 Color

Tính năng Colors trong Tailwind CSS dùng để định dạng màu sắc trên trang web của người dùng. Tailwind cung cấp một hệ thống màu sắc đa dạng và linh hoạt với hơn 300 màu sắc được định nghĩa sẵn, giúp người dùng dễ dàng tạo ra các bảng màu sắc và sử dụng chúng trên trang web của mình. Một trong những đặc điểm nổi bật của tính năng Colors trong Tailwind là hệ thống tên màu sắc đơn giản, cho phép người dùng nhanh chóng định dạng màu sắc trên trang web của mình bằng cách sử dụng các tên màu sắc được đặt tên cụ thể. Người dùng có thể sử dụng các tên màu sắc đặc biệt, chẳng hạn như màu sắc chính (primary), màu sắc phụ (secondary), màu sắc thành công (success) và nhiều hơn nữa. Ngoài ra, Tailwind cũng cung cấp một hệ thống màu sắc tùy chỉnh, cho phép người dùng tạo ra các bảng màu sắc riêng của mình bằng cách chỉ định các giá trị màu HEX hoặc RGB. Người dùng có thể sử dụng các lớp CSS để định dạng các phần tử trên trang web của mình với các màu sắc tùy chỉnh này. Với tính năng Colors của Tailwind, người dùng có thể tạo ra các bảng màu sắc

đẹp và đồng nhất trên trang web của mình, dễ dàng sử dụng và tùy biến để phù hợp với mục đích của người dùng.

2.3.3.4 Sizing

Tính năng Sizing trong Tailwind CSS cung cấp các lớp CSS để định nghĩa kích thước cho các phần tử. Các giá trị này có thể được sử dụng cho các thuộc tính như width và height.

2.3.3.5 Borders

Tính năng Borders trong Tailwind CSS cung cấp các lớp CSS để định nghĩa các thuộc tính border (đường viền) cho các phần tử. Đặc điểm:

- Các giá trị được đặt tên theo kiểu đường viền và độ dày: solid, dashed, dotted, double, none, ...
- Các lớp CSS sử dụng tiền tố để định nghĩa kiểu border (vd: border- để định nghĩa kiểu đường viền), theo sau là các giá trị độ dày và kiểu đường viền với tên tương ứng (vd: border-2-solid để định nghĩa border với độ dày 2 và kiểu đường viền solid).

2.3.3.6 Shadows

Một số đặc điểm nổi bật của tính năng Shadows trong Tailwind CSS bao gồm:

- Có thể định nghĩa nhiều lớp đổ bóng cho một phần tử.
- Các giá trị độ sâu của đổ bóng được định nghĩa bằng các tên độ sâu tương ứng với số pixel tương ứng.
- Có thể định nghĩa kiểu đổ bóng là inset (đổ bóng lõm) bằng cách sử dụng tiền tố inset.

2.3.3.7 Interactivity

Tính năng Interactivity trong Tailwind CSS cung cấp các lớp CSS để định nghĩa các thuộc tính tương tác cho các phần tử, chẳng hạn như hover, focus, active, và disabled. Với các đặc điểm bao gồm:

- Các lớp tương tác có thể được kết hợp với các lớp khác để tạo ra các hiệu ứng đa dạng.
- Các giá trị của thuộc tính được định nghĩa bằng các hàm được đặt tên tương ứng với các trạng thái tương tác.
- Tailwind CSS cung cấp các hàm để xác định các kiểu trạng thái khác nhau, bao gồm hover, focus, active, và disabled.

2.3.4 So sánh Tailwindcss với các công cụ khác

2.3.4.1 So sánh Tailwindcss với Bootstrap

TailwindCSS và Bootstrap là hai trong số các framework phổ biến để phát triển giao diện người dùng. Mặc dù cả hai đều cung cấp những tính năng tuyệt vời để giảm thiểu việc viết mã CSS, nhưng chúng khác nhau về cách tiếp cận và cách thức triển khai. Điểm khác biệt chính giữa TailwindCSS và Bootstrap đó là;

- Tiếp cận thiết kế
 - **Bootstrap** hướng tới việc đơn giản hóa việc phát triển bằng cách cung cấp một loạt các thành phần và lớp CSS định sẵn để người dùng có thể sử dụng mà không cần phải tạo ra nhiều mã tùy chỉnh.
 - **TailwindCSS** cung cấp một bộ công cụ thiết kế linh hoạt hơn, cho phép người dùng tùy chỉnh chi tiết hơn và tạo ra giao diện độc đáo hơn thông qua việc sử dụng các lớp CSS được định nghĩa sẵn.
- Cú pháp và cách thức triển khai
 - **Bootstrap** sử dụng một cú pháp đơn giản hơn với các lớp CSS định sẵn được đặt tên rõ ràng, điều này làm cho việc triển khai trở nên dễ dàng hơn đối với người mới bắt đầu.
 - **TailwindCSS** sử dụng một cú pháp phức tạp hơn, cung cấp các lớp CSS được đặt tên ngắn gọn nhưng có thể khó đọc và hiểu hơn. Nhưng nó cho phép người dùng tùy chỉnh linh hoạt hơn.
- Các tính năng
 - **Bootstrap** cung cấp nhiều tính năng và thành phần được định sẵn để người dùng có thể sử dụng mà không cần phải tạo ra nhiều mã tùy

chỉnh. Nó cũng đi kèm với các plugin và thư viện JavaScript để giúp người dùng thực hiện các tính năng phức tạp hơn.

- **TailwindCSS** tập trung vào việc cung cấp một bộ công cụ linh hoạt để người dùng có thể tùy chỉnh và tạo ra giao diện độc đáo hơn. Nó không đi kèm với các plugin hoặc thư viện JavaScript, nhưng nó cung cấp các lớp CSS được định nghĩa sẵn để người dùng có thể sử dụng chúng trong các tệp JavaScript của riêng mình.
- **Hiệu suất**
 - TailwindCSS cho phép người dùng tối ưu hóa giao diện của mình bằng cách chọn chọn lớp CSS được sắp xếp trước. Điều này giúp giảm thiểu kích thước file CSS và tăng tốc độ tải trang, giúp cho trang web của người dùng có thể tải nhanh hơn.
 - Bootstrap cũng cung cấp các tùy chọn tối ưu hóa để giảm kích thước file CSS, nhưng cách tiếp cận của Bootstrap khác với TailwindCSS. Bootstrap sử dụng Sass để định nghĩa các lớp CSS và cung cấp các bản tùy chỉnh để người dùng có thể chỉ chọn các phần cần thiết cho trang web của mình.

2.3.4.2 So sánh Tailwindcss với Foundation

Tailwindcss và Foundation đều là các framework CSS được sử dụng rộng rãi để xây dựng các trang web đáp ứng và hiện đại. Tuy nhiên, hai công cụ này có những điểm khác biệt cơ bản về triết lý thiết kế, tính năng, hiệu suất và khả năng tùy chỉnh.

- **Triết lý thiết kế**
 - **Tailwindcss:** Tailwindcss có triết lý thiết kế “utility-first” (tập trung vào các lớp tiện ích), cho phép người dùng tùy chỉnh các kiểu thiết kế thông qua việc kết hợp các lớp CSS sẵn có, thay vì viết CSS tùy chỉnh từ đầu. Điều này giúp giảm thiểu thời gian viết CSS và tối ưu hóa kích thước tệp CSS.
 - **Foundation:** Foundation có triết lý thiết kế “component-first” (tập trung vào các thành phần), với các thành phần CSS sẵn có để giúp người dùng

xây dựng các trang web đáp ứng nhanh chóng và hiệu quả. Điều này giúp người dùng tiết kiệm thời gian trong việc thiết kế và phát triển.

- **Tính năng**
 - **Tailwindcss:** Tailwindcss cung cấp các lớp CSS tiện ích đa dạng, từ việc xử lý khoảng cách đến tính năng tạo hiệu ứng nổi bật. Ngoài ra, Tailwindcss cũng cung cấp các tính năng như responsive design và dark mode, giúp người dùng tùy chỉnh giao diện đa dạng và linh hoạt.
 - **Foundation:** Foundation cung cấp các thành phần CSS sẵn có để xây dựng các trang web đáp ứng, từ việc xử lý lưới đến các thành phần như nút, biểu mẫu và thanh trượt. Foundation cũng cung cấp tính năng tùy chỉnh và responsive design
- **Hiệu suất**
 - **Tailwindcss:** Tailwindcss có kích thước tệp CSS nhỏ hơn so với Foundation và các framework CSS khác, do sử dụng lớp CSS tiện ích và tái sử dụng lớp CSS nhiều lần.
 - **Foundation:** Foundation có kích thước tệp CSS lớn hơn so với Tailwindcss và các framework CSS khác, do sử dụng các thành phần CSS riêng biệt và không sử dụng lại các lớp CSS.
- **Khả năng tùy chỉnh**
 - **Tailwindcss:** Tailwindcss cho phép người dùng tùy chỉnh mọi chi tiết của giao diện, từ màu sắc, kích thước, đường viền cho đến khoảng cách và hiệu ứng đổ bóng. Nó cung cấp cho người dùng hơn 200 lớp tiện ích được định nghĩa sẵn, đáp ứng nhu cầu tùy chỉnh của người dùng. Bên cạnh đó, Tailwindcss cũng cung cấp các công cụ để tạo ra các kiểu tùy chỉnh nhanh chóng.
 - **Foundation:** Foundation cũng cho phép người dùng tùy chỉnh các yếu tố của giao diện, bao gồm màu sắc, kích thước, khoảng cách và kiểu chữ. Nó cũng cung cấp một số lớp tiện ích để giúp tùy chỉnh các thành phần của giao diện. Tuy nhiên, số lượng các lớp tiện ích

và tính năng tùy chỉnh trong Foundation không nhiều bằng Tailwindcss.

Tóm lại, Tailwindcss cho phép tùy chỉnh giao diện một cách nhanh chóng và linh hoạt hơn, với nhiều lớp tiện ích và các công cụ tùy chỉnh khác. Trong khi đó, Foundation cũng cung cấp tính năng tùy chỉnh, tuy nhiên, nó không nhiều và tiện ích hơn Tailwindcss.

2.3.4.3 So sánh với CSS Grid và Flexbox

CSS Grid và Flexbox là hai công nghệ quan trọng trong thiết kế giao diện và layout của trang web. Chúng đều được sử dụng để xây dựng cấu trúc layout của trang web, tuy nhiên, cách tiếp cận và tính năng của chúng khác nhau. Dưới đây là so sánh chi tiết giữa TailwindCSS và CSS Grid/Flexbox.

- Tính năng
 - **CSS Grid** và **Flexbox** đều cung cấp tính năng xây dựng layout của trang web. Tuy nhiên, Flexbox thường được sử dụng để xây dựng layout tuyến tính (linear layout) và định hướng theo chiều dọc hoặc ngang của trang web, trong khi CSS Grid cho phép người dùng xác định kích thước và vị trí của mỗi phần tử trong lưới (grid) 2 chiều.
 - **TailwindCSS** cung cấp các lớp và hàm để xây dựng layout và các thành phần giao diện khác. Nó cũng có tính năng tạo ra các kiểu tùy chỉnh nhanh chóng, cho phép người dùng thiết kế giao diện theo ý muốn.
- Khả năng tùy chỉnh
 - **CSS Grid** và **Flexbox** cung cấp khả năng tùy chỉnh linh hoạt cho layout của trang web, cho phép người dùng xác định vị trí và kích thước của các phần tử trên trang web một cách chính xác.
 - **TailwindCSS** cung cấp các lớp và hàm để tùy chỉnh giao diện, cho phép người dùng dễ dàng tạo ra các kiểu tùy chỉnh.
- Hiệu suất
 - **CSS Grid** và **Flexbox** là các công nghệ CSS cơ bản, do đó chúng có tốc độ render nhanh và tương thích tốt trên các trình duyệt khác nhau.

- **TailwindCSS** có thể tạo ra các tệp CSS khá lớn, tuy nhiên nó cung cấp các tính năng tối ưu để giảm kích thước tệp CSS và cải thiện hiệu suất của trang web.
- Khả năng sử dụng
 - **CSS Grid** và **Flexbox** yêu cầu một số kiến thức về CSS và thiết kế layout để sử dụng hiệu quả.
 - **TailwindCSS** cung cấp các lớp và hàm để sử dụng, giúp người dùng dễ dàng thiết kế giao diện mà không cần nhiều kiến thức về CSS.

Tóm lại, TailwindCSS, CSS Grid và Flexbox đều là các công nghệ quan trọng trong thiết kế giao diện web hiện đại. Tuy nhiên, chúng có những đặc điểm khác nhau và có thể được sử dụng để giải quyết các vấn đề khác nhau trong quá trình thiết kế.

TailwindCSS: TailwindCSS cung cấp một bộ công cụ mạnh mẽ để tạo kiểu cho các phần tử HTML bằng cách sử dụng các lớp CSS tiền định nghĩa. Nó cho phép người dùng tùy chỉnh mọi chi tiết của giao diện và có khả năng tương thích với nhiều framework khác. Tuy nhiên, việc sử dụng TailwindCSS có thể đòi hỏi một thời gian dài để học cách sử dụng và tối ưu hóa CSS của người dùng.

CSS Grid: CSS Grid là một công nghệ phổ biến trong thiết kế giao diện web hiện đại. Nó cung cấp một cách thức để tạo ra các bố cục phức tạp với sự linh hoạt cao hơn so với các phương pháp truyền thống như float và position. CSS Grid cũng cung cấp khả năng tương thích với các thiết bị đa nền tảng và là một công nghệ đáng xem xét khi thiết kế giao diện web.

Flexbox: Flexbox cũng là một công nghệ phổ biến trong thiết kế giao diện web hiện đại. Nó cung cấp một cách thức để tạo ra các bố cục linh hoạt và phản ứng với các thay đổi kích thước của các phần tử. Flexbox là một công nghệ đơn giản và dễ sử dụng, tuy nhiên, nó có hạn chế khi xử lý các bố cục phức tạp hơn.

2.3.4.4 Tổng kết

Tailwind CSS là một framework CSS phổ biến được sử dụng để tạo ra các giao diện người dùng phức tạp một cách nhanh chóng và dễ dàng. Dưới đây là một số điểm mạnh và yếu của Tailwind CSS so với các công nghệ khác:

Điểm mạnh:

- **Tốc độ phát triển:** Tailwind CSS cho phép người dùng tạo các giao diện người dùng phức tạp một cách nhanh chóng và dễ dàng hơn bằng cách cung cấp các class tùy chỉnh để tạo các kiểu giao diện khác nhau.
- **Cấu trúc rõ ràng:** Tailwind CSS giúp người dùng tạo các class tùy chỉnh có cấu trúc rõ ràng hơn, giúp mã của người dùng dễ đọc và dễ bảo trì hơn.
- **Tính tùy chỉnh cao:** Tailwind CSS cho phép người dùng tùy chỉnh hoàn toàn giao diện của mình bằng cách sử dụng các class tùy chỉnh và cấu hình.
- **Tương thích với các framework khác:** Tailwind CSS có thể được sử dụng cùng với các framework khác như React, Vue và Angular.
- **Thiết kế đáp ứng:** Tailwind CSS cung cấp các class cho thiết kế đáp ứng, giúp người dùng dễ dàng tạo ra các giao diện người dùng đáp ứng cho các thiết bị khác nhau.

Điểm yếu:

- **Cú pháp phức tạp:** Tailwind CSS sử dụng cú pháp phức tạp và khó nhớ, điều này có thể khiến cho việc sử dụng Tailwind CSS ban đầu khó khăn đối với một số người.
- **Kích thước tập tin lớn:** Do Tailwind CSS cung cấp nhiều class, kích thước tập tin CSS có thể lớn hơn so với các framework khác.
- **Khó để tùy chỉnh các class tùy chỉnh:** Mặc dù Tailwind CSS cung cấp cho người dùng khả năng tùy chỉnh giao diện của mình bằng cách sử dụng các class tùy chỉnh, nhưng việc tìm kiếm và chỉnh sửa các class tùy chỉnh có thể khó khăn đối với người mới bắt đầu.

2.4 ASP.NET

2.4.1 Giới thiệu

2.4.1.1 Lịch sử và các phiên bản ASP.NET

ASP.NET là một framework ứng dụng web được phát triển bởi Microsoft để phát triển các ứng dụng web động. Nó ra đời lần đầu tiên vào năm 2002 với phiên bản đầu tiên là ASP.NET 1.0.

Trước đó, Microsoft đã phát triển hai công nghệ web khác là Active Server Pages (ASP) và Visual InterDev. Tuy nhiên, chúng không đủ mạnh để xử lý các ứng dụng web phức tạp. Vì vậy, Microsoft đã quyết định phát triển một framework mới cho phép phát triển các ứng dụng web động một cách dễ dàng hơn, linh hoạt hơn và có khả năng xử lý các ứng dụng phức tạp hơn. Kết quả của quá trình phát triển đó là ASP.NET.

Các phiên bản ASP.NET sau đó đã được phát triển, cải tiến và cung cấp nhiều tính năng mới như:

- ASP.NET 1.1 (2003): cung cấp tính năng mới như Web Parts, DataGrid, Calendar và Multi-lingual support.
- ASP.NET 2.0 (2005): cung cấp tính năng mới như Master Pages, Membership API, Role Management, Personalization, Web Parts, cải thiện việc tạo và quản lý các trang web.
- ASP.NET 3.5 (2007): cung cấp tính năng mới như AJAX, LINQ, Entity Framework, ADO.NET Data Services, MVC Framework.
- ASP.NET 4.0 (2010): cung cấp tính năng mới như Routing, Chart Controls, SEO Optimization, Web Forms Model Binding, Dynamic Data.
- ASP.NET 4.5 (2012): cung cấp tính năng mới như WebSockets, SignalR, Async/Await, Web API.
- ASP.NET Core (2016): một phiên bản ASP.NET hoàn toàn mới, được thiết kế lại để hỗ trợ cross-platform, performance cao hơn, linh hoạt hơn.

Dần dần ASP.NET đã trở thành một trong những framework ứng dụng web phổ biến nhất trên thế giới và được sử dụng rộng rãi trong việc phát triển các ứng dụng web trên nhiều nền tảng khác nhau. Hiện tại, phiên bản mới nhất của ASP.NET là ASP.NET Core 6, vẫn được phát triển và nâng cấp liên tục.

2.4.1.2 Kiến trúc của ASP.NET và cách hoạt động của nó

ASP.NET có kiến trúc theo mô hình MVC (Model-View-Controller) và là một framework ứng dụng web server-side. Kiến trúc của ASP.NET MVC bao gồm ba thành phần chính: Model, View và Controller. Các thành phần này đóng vai trò quan trọng trong việc xử lý yêu cầu và phản hồi của các ứng dụng web.

- **Model:** là thành phần chịu trách nhiệm xử lý dữ liệu. Nó chứa các đối tượng và logic xử lý dữ liệu, bao gồm cả các thao tác CRUD (Create, Read, Update, Delete) với cơ sở dữ liệu. Model đóng vai trò trung gian giữa Controller và Database.
- **View:** là thành phần chịu trách nhiệm hiển thị giao diện cho người dùng. Nó sử dụng các template để tạo ra các trang web và hiển thị dữ liệu được lấy từ Model.
- **Controller:** là thành phần điều phối việc xử lý yêu cầu của người dùng. Nó nhận yêu cầu từ người dùng thông qua URL và định tuyến nó đến một phương thức xử lý trong Controller. Sau đó, Controller sử dụng Model để lấy dữ liệu và View để hiển thị kết quả cho người dùng.

Cách hoạt động của ASP.NET theo mô hình MVC như sau:

- Người dùng tạo một yêu cầu trên trình duyệt. Yêu cầu này bao gồm các thông tin như URL, dữ liệu form hoặc các thông số truyền vào qua URL.
- Yêu cầu được gửi đến Server.
- Server sử dụng Routing để xác định Controller phù hợp với yêu cầu này. Routing là quá trình xác định Controller và phương thức xử lý yêu cầu dựa trên URL được yêu cầu.

- Controller được khởi tạo và phương thức xử lý yêu cầu được gọi. Phương thức này sẽ tương tác với Model để lấy dữ liệu, thực hiện các thao tác xử lý dữ liệu và chuẩn bị dữ liệu cho View.
- View được gọi và sử dụng dữ liệu được chuẩn bị từ Controller để hiển thị trang web. View sử dụng các template để tạo giao diện người dùng.
- Kết quả được trả về cho Server và được gửi trở lại cho người dùng thông qua trình duyệt.

2.4.2 Các ngôn ngữ lập trình được hỗ trợ bởi ASP.NET

- Ngôn ngữ C#
 - Là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft.
 - Được sử dụng rộng rãi trong phát triển ứng dụng web ASP.NET.
 - C# có cú pháp đơn giản, dễ hiểu và dễ học.
 - Hỗ trợ nhiều tính năng mới như async/await, LINQ, dynamic, extension methods.
- Ngôn ngữ VB.NET
 - Là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft.
 - Được sử dụng trong phát triển ứng dụng web ASP.NET, nhưng ít phổ biến hơn so với C#.
 - Cú pháp của VB.NET giống với cú pháp của ngôn ngữ BASIC, dễ hiểu và dễ học cho người mới bắt đầu.
 - Hỗ trợ các tính năng như đa luồng, LINQ, delegate.
- Ngôn ngữ F#
 - Là một ngôn ngữ lập trình hướng hàm và hướng đối tượng được phát triển bởi Microsoft.
 - Được sử dụng trong phát triển ứng dụng web ASP.NET, nhưng ít phổ biến hơn so với C# và VB.NET.

- F# hỗ trợ tính năng xử lý dữ liệu song song, giúp cải thiện hiệu suất ứng dụng.
- F# có cú pháp đơn giản, dễ hiểu và dễ học cho những người có kinh nghiệm với lập trình hướng hàm.

Tùy thuộc vào mục đích và đặc điểm của ứng dụng web, các nhà phát triển có thể lựa chọn sử dụng C#, VB.NET hoặc F# để phát triển ứng dụng của mình. Mỗi ngôn ngữ đều có những ưu điểm và hạn chế riêng. Tuy nhiên, trong phát triển ứng dụng web ASP.NET, C# là ngôn ngữ được sử dụng phổ biến nhất và có sự ưu tiên hỗ trợ nhiều tính năng mới nhất từ Microsoft.

2.4.3 Các công nghệ liên quan đến ASP.NET

2.4.3.1 Entity Framework và ADP.NET cho việc tương tác với cơ sở dữ liệu

Entity Framework: Entity Framework là một ORM (Object-Relational Mapping) framework được phát triển bởi Microsoft. Nó cho phép các nhà phát triển ứng dụng web ASP.NET tương tác với cơ sở dữ liệu một cách dễ dàng hơn bằng cách ánh xạ các đối tượng .NET với các bảng trong cơ sở dữ liệu. Entity Framework cung cấp một số đối tượng để thao tác với cơ sở dữ liệu như DbContext, DbSet, EntityObject. Để sử dụng Entity Framework, các nhà phát triển cần phải định nghĩa các lớp đại diện cho các bảng trong cơ sở dữ liệu và thiết lập kết nối cơ sở dữ liệu. Sau đó, các nhà phát triển có thể sử dụng các phương thức có sẵn trong Entity Framework để truy vấn và thao tác với cơ sở dữ liệu.

ADO.NET: ADO.NET (ActiveX Data Objects .NET) là một bộ công cụ được sử dụng để tương tác với cơ sở dữ liệu trong ứng dụng web ASP.NET. Nó cung cấp các đối tượng để kết nối, truy vấn và thao tác với cơ sở dữ liệu như SqlCommand, SqlConnection, SqlDataReader, SqlDataAdapter. Nó cũng hỗ trợ nhiều loại cơ sở dữ liệu như SQL Server, Oracle, MySQL. ADO.NET thường được sử dụng khi cần phải tương tác trực tiếp với cơ sở dữ liệu bằng cách viết các câu lệnh SQL.

So sánh ADO.NET và Entity Framework:

- Tính trừu tượng hóa cao hơn: Entity Framework tạo ra một lớp trừu tượng giữa các đối tượng .NET và cơ sở dữ liệu, giúp các nhà phát triển tập trung vào việc phát triển ứng dụng mà không phải lo lắng về cách tương tác với cơ sở dữ liệu.
- Tự động sinh mã: Entity Framework có khả năng tự động sinh ra các câu lệnh SQL từ các đối tượng .NET, giúp giảm thiểu số lượng mã cần phải viết.
- Định dạng tốt hơn: Các đối tượng được tạo bởi Entity Framework có thể được định dạng tốt hơn so với các đối tượng DataTable truyền thống của ADO.NET.

Nhìn chung Entity Framework là một công cụ ORM rất mạnh mẽ trong việc tương tác với cơ sở dữ liệu trong ASP.NET. Nó cung cấp một cách tiếp cận trực quan và dễ dàng để thao tác với cơ sở dữ liệu và giúp giảm thiểu số lượng mã cần phải viết, đồng thời cải thiện hiệu suất của ứng dụng.

2.4.3.2 ASP.NET Identity cho việc xác thực và phân quyền người dùng

ASP.NET Identity là một bộ thư viện được tích hợp sẵn trong ASP.NET, cung cấp các tính năng xác thực và phân quyền cho ứng dụng web. Nó hỗ trợ nhiều phương thức xác thực và lưu trữ thông tin người dùng, cũng như quản lý quyền truy cập người dùng vào các tính năng và nội dung của ứng dụng. ASP.NET Identity cung cấp các thành phần như UserManager, RoleManager, SignInManager, UserStore, RoleStore,... giúp cho việc quản lý người dùng và phân quyền dễ dàng hơn.

2.4.3.3 Các thư viện JavaScripts và CSS phổ biến được sử dụng trong ASP.NET

Trong ASP.NET, có nhiều thư viện JavaScript và các framework CSS phổ biến được sử dụng để phát triển các ứng dụng web. Một số thư viện JavaScript và framework CSS phổ biến được sử dụng trong ASP.NET:

- jQuery: jQuery là một thư viện JavaScript phổ biến và được sử dụng rộng rãi trong phát triển web. Nó cung cấp các chức năng xử lý sự kiện, tương tác với DOM và AJAX một cách dễ dàng và thuận tiện. Trong ASP.NET, jQuery thường được sử dụng để tạo các hiệu ứng và xử lý sự kiện trên các trang web.

- **AngularJS:** AngularJS là một framework JavaScript phổ biến được phát triển bởi Google. Nó được sử dụng để phát triển các ứng dụng web động và đáp ứng. AngularJS cung cấp các tính năng như two-way data binding, dependency injection, routing và directives để giúp giảm thiểu số lượng mã cần phải viết và tăng tính tái sử dụng của mã.
- **React:** React là một thư viện JavaScript phổ biến được phát triển bởi Facebook. Nó được sử dụng để phát triển các ứng dụng web đơn giản và dễ bảo trì. React cung cấp các tính năng như Virtual DOM, component-based architecture và JSX để tạo ra các ứng dụng web nhanh và hiệu quả.
- **Vue.js:** Vue.js là một framework JavaScript phổ biến được sử dụng để phát triển các ứng dụng web đơn giản. Nó cung cấp các tính năng như two-way data binding, component-based architecture và directives để giúp giảm thiểu số lượng mã cần phải viết và tăng tính tái sử dụng của mã.
- **Tailwind CSS:** Tailwind CSS là một framework CSS phổ biến được sử dụng để phát triển các ứng dụng web động và đáp ứng. Nó cung cấp các class CSS để tạo các thiết kế hiện đại và tương tác với các thành phần khác nhau trên trang web. Tailwind CSS cho phép nhà phát triển tùy chỉnh và tạo ra các thiết kế theo ý muốn của mình.

2.4.4 Framework ASP.NET CORE API

2.4.4.1 Giới thiệu về Core API

ASP.NET Core API là một framework phát triển ứng dụng dành cho các dịch vụ web và ứng dụng API. Nó là một phần của ASP.NET Core và được xây dựng dựa trên nền tảng .NET Core mới nhất. Mục đích chính của Core API là cung cấp cho các nhà phát triển một cách tiện lợi để xây dựng các dịch vụ web và ứng dụng API độc lập và linh hoạt. Mục đích chính của Core API là giúp cho các nhà phát triển phát triển các ứng dụng web độc lập nền tảng dễ dàng hơn. Core API hỗ trợ phát triển các API RESTful, cho phép các nhà phát triển tạo các dịch vụ web linh hoạt và mở rộng được tích hợp với các ứng dụng web hoặc ứng dụng di động. Core API cũng cung cấp nhiều tính năng tiên tiến để quản lý việc xác thực, phân quyền, bảo mật và ghi nhật ký, giúp đơn

giản hóa quá trình phát triển và quản lý ứng dụng. Sử dụng Core API mang lại nhiều lợi ích cho các nhà phát triển. Đầu tiên, Core API cho phép các nhà phát triển tạo ra các ứng dụng web độc lập nền tảng, giúp giảm chi phí và độ phức tạp của quá trình phát triển. Core API cũng cung cấp một khung làm việc tiên tiến và dễ sử dụng để phát triển các dịch vụ web và các API RESTful. Ngoài ra, Core API còn hỗ trợ các tính năng quản lý việc xác thực, phân quyền, bảo mật và ghi nhật ký, giúp đơn giản hóa quá trình phát triển và quản lý ứng dụng.

2.4.4.2 Kiến trúc Core API

Kiến trúc của Core API bao gồm các thành phần cơ bản để xây dựng ứng dụng Web API, bao gồm:

- **Middleware:** là các thành phần của Core API được sử dụng để xử lý các yêu cầu từ khách hàng và trả về các phản hồi. Các Middleware được kết nối với nhau để tạo thành một chuỗi xử lý yêu cầu đầy đủ. Mỗi Middleware đóng vai trò như một bộ lọc trung gian giữa yêu cầu đến và phản hồi trả về.
- **Routing:** được sử dụng để xác định cách các yêu cầu của khách hàng được xử lý bởi ứng dụng. Nó cung cấp các cơ chế để map các yêu cầu đến các xử lý yêu cầu cụ thể, như các action của controller hoặc các endpoint của API.
- **Controllers:** là các thành phần chính của Core API và được sử dụng để xử lý các yêu cầu đến từ khách hàng. Mỗi controller cung cấp một số action để xử lý các yêu cầu và trả về các phản hồi.
- **Models:** được sử dụng để định nghĩa các đối tượng dữ liệu được sử dụng trong ứng dụng. Nó cho phép định nghĩa các kiểu dữ liệu, các thuộc tính và các phương thức để thực hiện xử lý trên dữ liệu.
- **Services:** được sử dụng để xử lý các logic nghiệp vụ của ứng dụng. Chúng thực hiện các tác vụ xử lý dữ liệu hoặc tương tác với các dịch vụ bên ngoài

2.4.5 EntityFramework Core trong ASP.NET

2.4.5.1 Giới thiệu về Entity Framework Core

Entity Framework (EF) Core là một trong những công nghệ Object-Relational Mapping (ORM) phổ biến nhất dành cho .NET Framework. EF Core là một phiên bản tái thiết kế hoàn toàn của Entity Framework truyền thống và được thiết kế để chạy trên nhiều nền tảng, bao gồm .NET Framework, .NET Core, Xamarin, và Universal Windows Platform (UWP).

Việc ra đời của EF Core được xem như là một sự cải tiến lớn đối với phiên bản trước đó vì nó có một số tính năng mới và cải thiện hiệu suất của ứng dụng. EF Core được thiết kế để đơn giản hóa việc phát triển các ứng dụng liên quan đến cơ sở dữ liệu bằng cách loại bỏ các yếu tố phức tạp như lập trình bằng SQL, đồng bộ hoá dữ liệu và hỗ trợ mô hình hóa đối tượng. Điều này cho phép các nhà phát triển tập trung vào các khía cạnh quan trọng hơn của ứng dụng, như tích hợp dữ liệu và logic kinh doanh. Việc loại bỏ các hoạt động không cần thiết này giúp tăng tốc độ phát triển, giảm thời gian và chi phí phát triển, giảm rủi ro cho các lỗi do phần mềm. EF Core được thiết kế để làm việc tốt với các cơ sở dữ liệu quan hệ, bao gồm Microsoft SQL Server, MySQL, PostgreSQL và Oracle. Việc hỗ trợ cho nhiều cơ sở dữ liệu này giúp cho nhà phát triển có thể sử dụng EF Core cho nhiều dự án khác nhau, tùy theo yêu cầu của từng dự án.

2.4.5.2 Mục đích sử dụng EntityFramework Core

Mục đích chính của Entity Framework Core (EF Core) là cung cấp cho các nhà phát triển một cách tiếp cận ổn định và linh hoạt để tương tác với cơ sở dữ liệu trong các ứng dụng .NET Core. EF Core là một ORM (Object-Relational Mapping) framework, cho phép các nhà phát triển tương tác với cơ sở dữ liệu bằng cách sử dụng các đối tượng và các thao tác CRUD (Create, Read, Update, Delete) thay vì viết các truy vấn SQL trực tiếp.

2.4.5.3 Điểm khác biệt giữa Entity Framework và Entity Framework Core

Cả Entity Framework và Entity Framework Core đều là các công cụ được sử dụng để tương tác với cơ sở dữ liệu trong ứng dụng .NET. Tuy nhiên, có một số điểm khác biệt quan trọng giữa hai công cụ này, bao gồm:

- **Độc lập với nền tảng:** Entity Framework Core được thiết kế để chạy trên nhiều nền tảng khác nhau, bao gồm .NET Framework và .NET Core, trong khi Entity Framework chỉ hỗ trợ .NET Framework.
- **Kích thước nhỏ gọn:** Entity Framework Core có kích thước nhỏ hơn so với Entity Framework, giúp giảm thiểu kích thước của ứng dụng và thời gian cài đặt.
- **Hiệu suất tốt hơn:** Entity Framework Core được thiết kế để cải thiện hiệu suất so với Entity Framework, bao gồm thời gian phản hồi nhanh hơn và khả năng xử lý tốt hơn các tác vụ liên quan đến cơ sở dữ liệu.
- **Thay đổi cú pháp:** Entity Framework Core có một số thay đổi cú pháp so với Entity Framework, bao gồm việc sử dụng các phương thức mới như `ToListAsync()` để tối ưu hóa hiệu suất.
- **Hỗ trợ linh hoạt:** Entity Framework Core cung cấp các tính năng linh hoạt hơn để tùy chỉnh cách mà nó tương tác với cơ sở dữ liệu, bao gồm khả năng chọn loại cơ sở dữ liệu và triển khai tùy chỉnh của nó.

2.4.5.4 Các thành phần của Entity Framework Core

- **DbContext:** Là một lớp trừu tượng được sử dụng để tương tác với cơ sở dữ liệu thông qua các đối tượng. Nó làm nhiệm vụ quản lý các đối tượng, theo dõi thay đổi và quản lý việc lưu trữ và truy xuất dữ liệu.
- **Entities:** Là các đối tượng đại diện cho các bảng trong cơ sở dữ liệu. Những đối tượng này được định nghĩa trong model và được sử dụng để thao tác với dữ liệu trong cơ sở dữ liệu.
- **DbSet:** Là một thuộc tính của DbContext, đại diện cho tập hợp của các đối tượng trong cơ sở dữ liệu. Nó được sử dụng để truy xuất và thao tác với các đối tượng liên quan đến một bảng cụ thể.

- Migrations: Là một công cụ cho phép tạo ra các phiên bản của cơ sở dữ liệu và áp dụng các thay đổi vào cơ sở dữ liệu một cách dễ dàng. Migrations cung cấp cho nhà phát triển một cách tiện lợi và an toàn để quản lý cấu trúc của cơ sở dữ liệu.
- LINQ to Entities: Là một công cụ giúp thao tác với các đối tượng Entity Framework Core bằng cách sử dụng ngôn ngữ LINQ (Language-Integrated Query). LINQ to Entities cho phép các câu truy vấn được viết bằng cách sử dụng các phương thức của LINQ để tương tác với cơ sở dữ liệu một cách hiệu quả.

2.5 C#

2.5.1 Giới thiệu về ngôn ngữ C#

C# là một ngôn ngữ lập trình phổ biến và mạnh mẽ được phát triển bởi Microsoft. Với các tính năng mạnh mẽ và hiệu suất cao, C# đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất cho việc phát triển các ứng dụng backend cho website.

C# được thiết kế để có thể chạy trên nền tảng .NET Framework của Microsoft, cho phép các lập trình viên phát triển các ứng dụng được tối ưu hóa để chạy trên môi trường Windows. Điều này cũng đồng nghĩa với việc C# có thể tích hợp được với các công nghệ và dịch vụ của Microsoft, bao gồm các dịch vụ đám mây Azure và các ứng dụng Windows.

C# cũng cung cấp các tính năng mạnh mẽ để xử lý và xử lý dữ liệu, bao gồm các thư viện hỗ trợ để truy vấn cơ sở dữ liệu và xử lý JSON. Ngoài ra, C# cũng có tính năng quản lý bộ nhớ hiệu quả, đặc biệt là với sự hỗ trợ của .NET Core, một phiên bản đa nền tảng của .NET Framework.

Với các tính năng mạnh mẽ và hiệu suất cao, C# đã trở thành lựa chọn phổ biến cho việc phát triển các ứng dụng backend cho website. Sử dụng C# có thể giúp cho các lập trình viên tối ưu hóa hiệu suất ứng dụng và nhanh chóng triển khai các tính năng mới, đồng thời tích hợp được với các dịch vụ và công nghệ của Microsoft.

2.5.2 Lịch sử và các phiên bản của C#

C# (C Sharp) là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft vào năm 2000. Ngôn ngữ này được thiết kế dựa trên C++ và Java và là một phần của .NET Framework. C# đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trong lĩnh vực phát triển ứng dụng Windows và web.

C# được phát hành với phiên bản đầu tiên là C# 1.0 trong năm 2002, sau đó là C# 2.0 vào năm 2005, C# 3.0 vào năm 2007, C# 4.0 vào năm 2010, C# 5.0 vào năm 2012, C# 6.0 vào năm 2015, C# 7.0 vào năm 2017, C# 8.0 vào năm 2019 và C# 9.0 vào năm 2020. Mỗi phiên bản mới của C# đều được cải tiến và bổ sung thêm các tính năng mới, cải thiện hiệu suất và đơn giản hóa cú pháp để giúp lập trình viên phát triển ứng dụng một cách nhanh chóng và dễ dàng hơn.

Với sự phát triển của các công nghệ web và ứng dụng web, C# được sử dụng rộng rãi trong lĩnh vực backend web để xử lý các yêu cầu từ phía client, truy vấn và quản lý cơ sở dữ liệu, xử lý nghiệp vụ và trả về kết quả cho client. C# cũng được sử dụng trong các lĩnh vực khác như phát triển game, ứng dụng desktop và mobile, và các ứng dụng đa nền tảng.

2.5.3 Vì sao nên chọn C# để làm backend cho web

- Đa nền tảng: C# có thể được sử dụng trên nhiều nền tảng khác nhau như Windows, Linux, macOS. Nó là một phần của .NET Framework và .NET Core, hai nền tảng phát triển đa năng cho việc xây dựng các ứng dụng web, ứng dụng di động, ứng dụng máy tính để bàn và các ứng dụng khác.
- Hiệu suất cao: C# được biên dịch trước khi chạy, cho phép tối ưu hóa và cải thiện hiệu suất của ứng dụng. Nó cũng hỗ trợ các tính năng như đa luồng và bộ nhớ quản lý, giúp cho các ứng dụng C# chạy mượt mà và ổn định hơn.
- Quản lý tài nguyên: C# có hỗ trợ tính năng quản lý tài nguyên tự động như garbage collection, giúp cho các nhà phát triển không phải lo lắng về việc giải phóng bộ nhớ và tài nguyên khác sau khi sử dụng chúng.
- Thư viện đa dạng: C# có sẵn một bộ thư viện lớn và đa dạng, giúp cho việc phát triển ứng dụng backend trở nên dễ dàng hơn. Nó cũng hỗ trợ các công

nghe như LINQ và Entity Framework, giúp cho việc truy xuất cơ sở dữ liệu trở nên đơn giản hơn.

- Cộng đồng hỗ trợ: C# có một cộng đồng lớn và đầy đủ tài nguyên hỗ trợ, từ các tài liệu hướng dẫn đến các thư viện phát triển. Điều này giúp cho việc học và phát triển ứng dụng backend trở nên dễ dàng hơn.

2.5.4 Kiến thức cần có để bắt đầu làm việc với C# cho backend web

- Lập trình hướng đối tượng: C# là một ngôn ngữ lập trình hướng đối tượng, do đó người lập trình cần phải hiểu về các khái niệm cơ bản của lập trình hướng đối tượng như đối tượng, lớp, kế thừa, đa hình, trừu tượng. Ngoài ra, người lập trình cần phải biết sử dụng các tính năng OOP của C# như interface, abstract class, delegate, event.
- Các khái niệm cơ bản của .NET Framework: C# là một ngôn ngữ lập trình phát triển trên .NET Framework, do đó người lập trình cần phải nắm vững các khái niệm cơ bản của .NET Framework như Common Language Runtime (CLR), Intermediate Language (IL), Assembly, Namespace, Class Library. Ngoài ra, người lập trình cũng cần biết sử dụng các công cụ phát triển .NET Framework như Visual Studio, NuGet Package Manager.
- Các kỹ thuật xử lý ngoại lệ: Khi phát triển ứng dụng, không tránh khỏi các lỗi xảy ra trong quá trình chạy. C# cung cấp các cơ chế để xử lý các ngoại lệ (exception) như try-catch-finally, throw, catch. Người lập trình cần phải nắm vững các kỹ thuật này để xử lý các lỗi trong ứng dụng.
- Kiến thức về web development: Nếu làm việc với C# cho backend web, người lập trình cần phải có kiến thức về web development như HTTP, HTML, CSS, JavaScript, jQuery, AJAX, RESTful API. Ngoài ra, người lập trình cũng cần phải nắm vững các framework phổ biến cho web development như ASP.NET Core, Entity Framework Core.
- Kiến thức về cơ sở dữ liệu: C# là một trong những ngôn ngữ lập trình phổ biến trong việc phát triển ứng dụng kết nối đến cơ sở dữ liệu. Do đó,

người lập trình cần phải có kiến thức về cơ sở dữ liệu như SQL, MySQL, SQL Server, PostgreSQL. Ngoài ra, người lập trình cũng cần phải nắm vững các công cụ ORM (Object-Relational Mapping) để tương tác với cơ sở dữ liệu như Entity Framework Core, Dapper.

2.6 Python-docx



Hình 2. 2 Python

Python là một ngôn ngữ lập trình phổ biến và mạnh mẽ được sử dụng rộng rãi trong nhiều lĩnh vực, bao gồm phân tích dữ liệu, trí tuệ nhân tạo, phát triển web nhiều hơn nữa. Một trong những lợi thế của Python là cộng đồng phát triển phong phú, cung cấp hàng ngàn thư viện và module cho các tác vụ cụ thể.

Thư viện python-docx là một trong số các thư viện phổ biến được phát triển cho Python và nó giúp làm việc với tệp tin định dạng DOCX, được sử dụng bởi Microsoft Word. Mục đích chính của thư viện này là cung cấp các công cụ và chức năng cho việc tạo, chỉnh sửa và đọc các tệp tin DOCX trong mã Python.

Với python-docx, người dùng có thể tạo và tùy chỉnh các thành phần trong tệp tin DOCX như tiêu đề, đoạn văn bản, bảng, hình ảnh, danh sách, định dạng văn bản, và

nhiều hơn nữa. Thư viện này giúp người dùng tự động hóa công việc tạo và chỉnh sửa tệp tin Word, đồng thời giảm bớt công sức và thời gian so với việc thủ công.

Các tính năng chính của thư viện:

- Tự động tạo báo cáo, tài liệu.
- Trích xuất dữ liệu từ tệp tin Docx và thực hiện phân tích hoặc xử lý dữ liệu.
- Tạo các biểu đồ hoặc bảng tóm tắt dữ liệu trong file word.
- Tự động thay đổi hoặc định dạng các phần tử trong tệp tin Docx dựa trên các quy tắc được xác định trước

Đặc điểm và ưu điểm của thư viện:

- Đặc điểm:
 - Đọc và ghi tệp tin DOCX: Thư viện python-docx cho phép người dùng đọc và ghi các tệp tin DOCX một cách dễ dàng, giúp người dùng truy cập và tương tác với nội dung của tài liệu Word.
 - Tạo và tùy chỉnh phần tử: Người dùng có thể tạo và tùy chỉnh các phần tử trong tệp tin DOCX như tiêu đề, đoạn văn bản, bảng, hình ảnh, danh sách, hyperlink và định dạng văn bản.
 - Định dạng văn bản: Thư viện cung cấp các công cụ để định dạng văn bản trong tệp tin DOCX, bao gồm kiểu chữ, cỡ chữ, màu sắc, định dạng đoạn, định dạng đặc biệt (in đậm, nghiêng, gạch chân), và căn chỉnh văn bản.
 - Xử lý hình ảnh: Người dùng có thể thêm hình ảnh vào tài liệu DOCX và điều chỉnh vị trí, kích thước và định dạng của hình ảnh đó.
 - Tương thích: python-docx tương thích với các phiên bản Microsoft Word từ Word 2007 trở lên, bao gồm cả Word 2010, Word 2013, Word 2016 và Word 2019.
- Ưu điểm:
 - Dễ sử dụng: Thư viện có cấu trúc dễ hiểu và API trực quan, giúp người dùng dễ dàng tạo và chỉnh sửa tệp tin DOCX.

- Mạnh mẽ và linh hoạt: python-docx cung cấp nhiều tính năng và chức năng để làm việc với tệp tin DOCX, cho phép người dùng tạo ra các tài liệu phong phú và tùy chỉnh theo nhu cầu cụ thể.
- Tích hợp tốt với Python: Thư viện được phát triển bằng Python và tương thích với các phiên bản Python phổ biến, đồng thời tương thích với các thư viện và công cụ khác trong hệ sinh thái Python.
- Cộng đồng hỗ trợ: python-docx được sử dụng rộng rãi và có một cộng đồng phát triển lớn, do đó người dùng có thể tìm thấy tài liệu, hướng dẫn và ví dụ phong phú

2.7 PYPDF2

2.7.1 Giới thiệu

PyPDF2



Hình 2. 3 PyPDF2

PyPDF2 là một thư viện mã nguồn mở phổ biến trong ngôn ngữ lập trình Python, được sử dụng để làm việc với tập tin PDF. Với PyPDF2 người dùng có thể viết và chỉnh sửa các tài liệu PDF một cách dễ dàng và linh hoạt. Với khả năng đọc tập tin PDF, PyPDF2 cho phép người dùng trích xuất thông tin quan trọng từ các tài liệu PDF đã có. Người dùng có thể truy cập và lấy thông tin như tiêu đề, tác giả, số trang và nhiều thông tin khác từ tập tin PDF. Điều này rất hữu ích cho việc xử lý dữ liệu hoặc hiển thị thông tin trong các ứng dụng.

2.7.2 Các tính năng của thư viện pypdf2

- **Đọc tập tin PDF:** PyPDF2 cho phép người dùng đọc nội dung của tập tin PDF có sẵn. Người dùng có thể truy cập các trang trong tập tin PDF và trích xuất thông tin từ chúng như văn bản và hình ảnh.
- **Truy xuất thông tin từ tập tin PDF:** PyPDF2 cung cấp các phương thức để truy xuất thông tin như tiêu đề, tác giả, số trang và nhiều thông tin khác từ tập tin PDF. Điều này giúp người dùng xử lý dữ liệu hoặc hiển thị thông tin trong ứng dụng của mình.
- **Tạo tài liệu PDF mới:** Bằng PyPDF2, Người dùng có thể tạo tài liệu PDF mới. Ngoài ra có thể tạo các trang, thêm văn bản, hình ảnh và các phần tử khác vào tài liệu PDF theo ý muốn. Người dùng cũng có thể tùy chỉnh thuộc tính của trang như kích thước và hướng.
- **Tạo tài liệu PDF mới:** Bằng PyPDF2, người dùng có thể tạo tài liệu PDF mới. Không chỉ thế, người dùng có thể tạo các trang, thêm văn bản, hình ảnh và các phần tử khác vào tài liệu PDF theo ý muốn.
- **Ghi tập tin PDF:** Khi hoàn thành chỉnh sửa hoặc tạo tài liệu PDF mới, PyPDF2 cho phép ghi tập tin này thành một tập tin PDF mới.
- **Tương thích:** PyPDF2 tương thích với các phiên bản Python khác nhau và hoạt động trên nhiều hệ điều hành, bao gồm Windows, macOS và Linux.

2.7.3 Ưu điểm của thư viện pypdf2

- **Dễ sử dụng:** PyPDF2 có cú pháp đơn giản và dễ hiểu, giúp người dùng dễ dàng tiếp cận và sử dụng thư viện một cách nhanh chóng.

- Hỗ trợ đa nền tảng: PyPDF2 hoạt động trên nhiều hệ điều hành phổ biến như Windows, macOS và Linux. Điều này cho phép người dùng sử dụng thư viện trên nhiều môi trường phát triển
- Tích hợp dễ dàng: PyPDF2 có thể dễ dàng tích hợp vào các dự án Python hiện có. Người dùng có thể sử dụng nó trong các ứng dụng web, dự án máy tính cá nhân hoặc các công cụ tự động hóa.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1 Mô tả nghiệp vụ hệ thống

3.1.1 Hỗ trợ phía người quản lý (Trưởng bộ môn)

- Quản lý thông tin về các đề cương môn học: Hệ thống cho phép người quản lý tạo, chỉnh sửa, xóa, tìm kiếm các đề cương môn học cũng như chương trình học trong hệ thống. Người quản lý có thể xem và cập nhập thông tin các môn học. Ngoài ra khi cần người quản lý có thể yêu cầu giảng viên bộ môn soạn các đề cương môn học do giảng viên đó phụ trách bằng cách gửi yêu cầu trong hệ thống.
- Quản lý thông tin về giảng viên: Hệ thống cho phép người quản lý quản lý thông tin về giảng viên bao gồm thông tin liên lạc của họ. Người quản lý có thể sử dụng thông tin này để liên hệ với giảng viên. Hệ thống chỉ cho phép người quản lý cấp tài khoản cho giảng viên, giảng viên không tự ý tạo tài khoản được.

3.1.2 Hỗ trợ phía người dùng (Trưởng bộ môn & Giảng viên)

- Hệ thống quản lý đề cương ngành học cho phép giảng viên có thể xem đề cương các môn học một cách nhanh chóng. Ngoài ra hệ thống còn hỗ trợ giảng viên tìm kiếm môn học dễ dàng.
- Hệ thống quản lý đề cương ngành học cho phép giảng viên soạn đề cương môn học một cách dễ dàng thông qua mẫu mà hệ thống đã có sẵn.

3.2 Chức năng của hệ thống

3.2.1 Người quản lý (Trưởng bộ môn)

- Đăng nhập theo admin
- Xem dữ liệu tổng quát trên dashboard
- Quản lý đề cương môn học (bao gồm tạo mới, chỉnh sửa, sao chép, xóa và tìm kiếm)
- Quản lý đề cương ngành học (bao gồm tạo mới, chỉnh sửa, sao chép, xóa và tìm kiếm)

- Quản lý người dùng (giảng viên) trong hệ thống (bao gồm cập nhập, thêm mới, xóa)
- Yêu cầu người dùng (giảng viên) tạo môn học

3.2.2 Người dùng (Giảng viên)

- Đăng nhập theo người dùng
- Xem dữ liệu tổng quát trên dashboard
- Xem nội dung các đề cương môn học
- Tìm kiếm đề cương môn học
- Chỉnh sửa đề cương môn học (khi có yêu cầu từ người quản lý)
- Xem thông tin cá nhân

3.3 Xác định yêu cầu hệ thống quản lý đề cương ngành học

- Quản lý đề cương môn học và ngành học: Hệ thống cần cho phép quản lý đề cương các môn học, bao gồm tạo mới, chỉnh sửa, sao chép và xóa đề cương (đối với quản lý). Đối với giảng viên chỉ được phép tạo đề cương môn học khi có yêu cầu từ người quản lý còn không thì chỉ xem và tìm kiếm các đề cương môn học.
- Quản lý giảng viên: Hệ thống cần cho phép quản lý thông tin về giảng viên, bao gồm tạo mới, chỉnh sửa và xóa thông tin.
- Tính năng tìm kiếm: Hệ thống cần cung cấp tính năng tìm kiếm để giúp người dùng tìm kiếm thông tin một cách nhanh chóng và chính xác.
- Báo cáo và thống kê: Hệ thống cần cung cấp các báo cáo và thống kê về số lượng các môn học, chương trình học và giảng viên.

3.4 Các yêu cầu chức năng

3.4.1 Đăng nhập

Người dùng truy cập vào đường dẫn đến trang đăng nhập.

Sau đó tiến hành nhập email và password.

Hệ thống sẽ kiểm tra thông tin và trả về 3 kết quả như sau:

- Nếu thông tin đăng nhập là trưởng bộ môn: trả về trang giao diện của trưởng bộ môn.
- Nếu thông tin đăng nhập là giảng viên: trả về trang giao diện của giảng viên.
- Nếu người dùng nhập sai email hoặc password: thông báo và yêu cầu người dùng nhập lại thông tin.

3.4.2 Trưởng bộ môn

Sau khi đăng nhập thành công, trưởng bộ môn sẽ có các chức năng sau:

- Xem thông tin môn học trên dashboard
 - Tổng số lượng giảng viên.
 - Tổng số lượng môn học.
 - Tổng số chương trình học.
 - Danh sách các môn học.
- Xem thông tin cơ bản của giảng viên như: email, khoa mà giảng viên thuộc vào, ngày sinh, nơi sinh.
- Quản lý chương trình học
 - Tạo mới chương trình học.
 - Chỉnh sửa chương trình học.
 - Xóa chương trình học.
 - In chương trình học.
- Quản lý môn học
 - Giao diện hiển thị danh sách các môn học. Người dùng có thể tìm kiếm môn học theo “tên môn” trên ô tìm kiếm.
 - Chọn một môn học bên danh sách các môn học để xem chi tiết nội dung môn học đó.
 - Với mỗi môn học, trưởng bộ môn có thể sử dụng các nút “Tạo môn học”, “Chỉnh sửa môn học”, “Sao chép môn học”, “In môn học”, “Xóa môn học”.
 - Chi tiết các chức năng:

- Tạo môn học: Khi nhấn vào nút “Tạo môn học”, trang web sẽ hiển thị giao diện cho phép nhập các thông tin của môn học với các khung trống.
- Chỉnh sửa môn học: Khi nhấn nút “Chỉnh sửa môn học”, trang web sẽ hiển thị giao diện giống như tạo môn học, chỉ khác là ở các trường dữ liệu đã có sẵn thông tin môn học.
- Quản lý giảng viên
 - Thêm giảng viên mới
 - Chỉnh sửa thông tin giảng viên
 - Reset mật khẩu giảng viên
 - Xóa giảng viên

3.4.3 Giảng viên

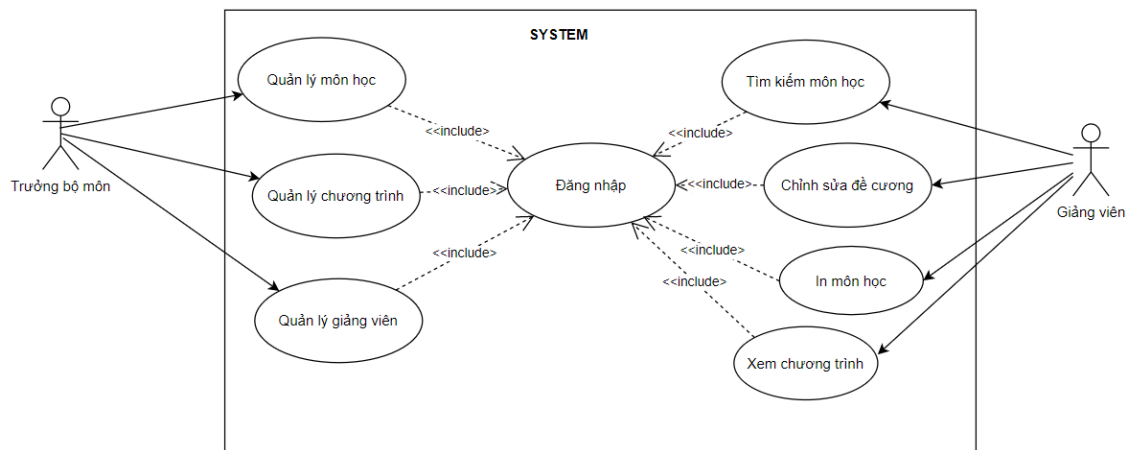
Sau khi đăng nhập thành công, giảng viên sẽ có các chức năng như sau:

- Xem thông tin môn học trên dashboard
 - Tổng số lượng giảng viên.
 - Tổng số lượng môn học.
 - Tổng số chương trình học.
 - Danh sách các môn học.
- Quản lý môn học
 - Xem thông tin các môn học
 - Chỉnh sửa môn học
- Quản lý chương trình

3.5 Phân tích và thiết kế

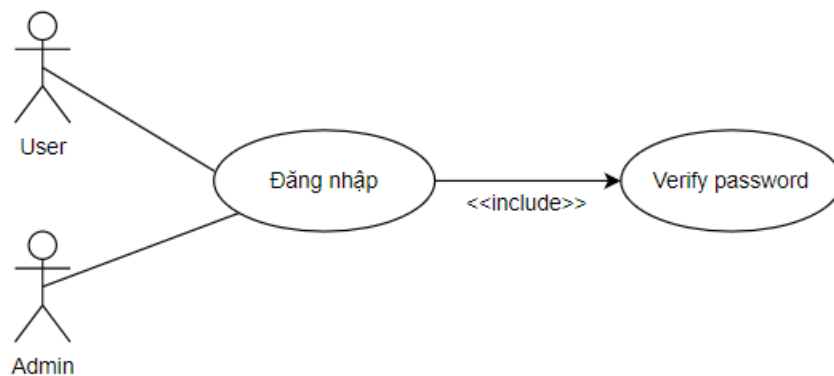
3.5.1 Xây dựng lược đồ use case

3.5.1.1 Use Case người dùng tổng quát



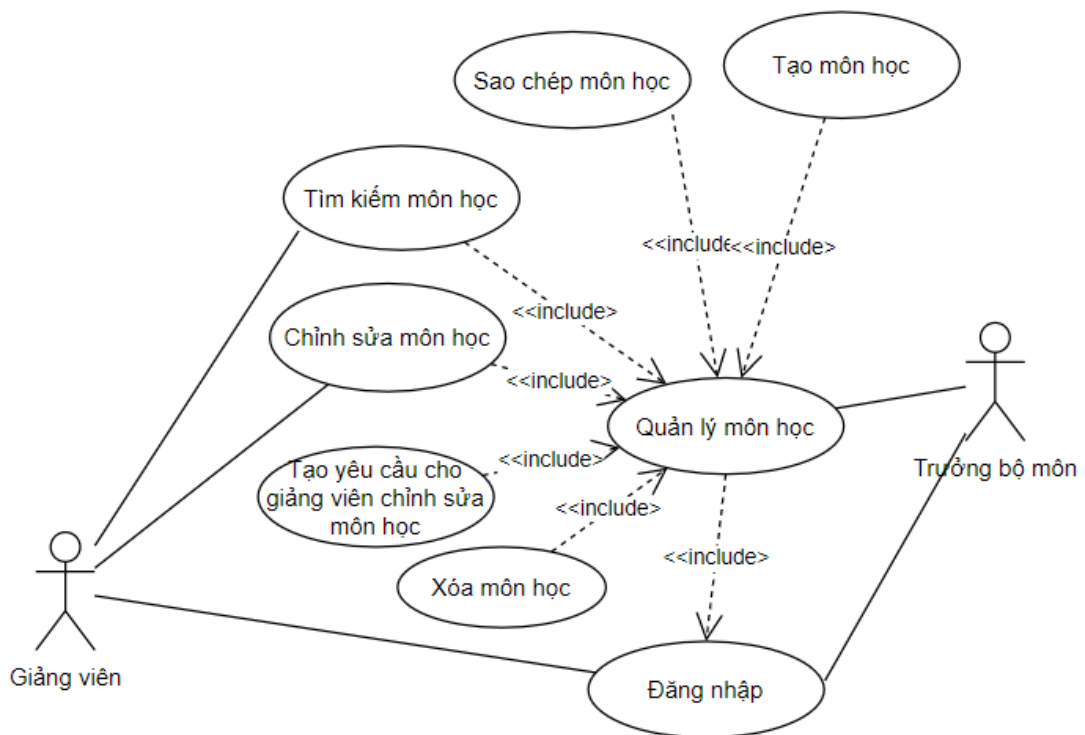
Hình 3. 1 Use case người dùng tổng quát

3.5.1.2 Use Case Đăng nhập



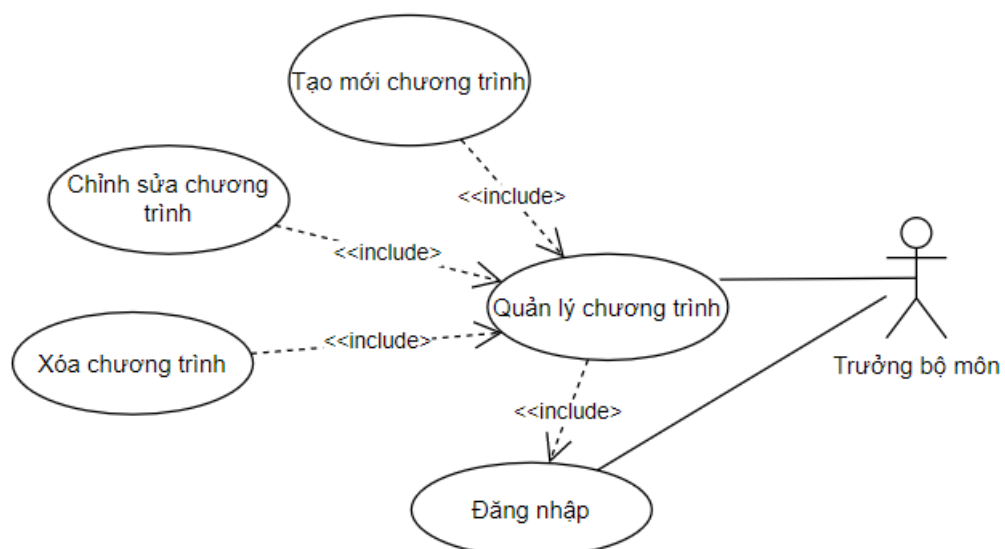
Hình 3. 2 Use case đăng nhập

3.5.1.3 Use Case Quản lý môn học



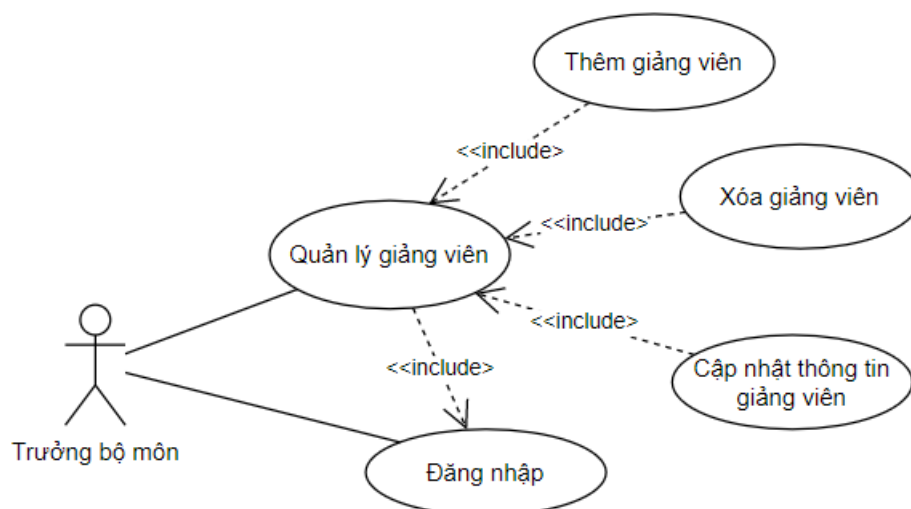
Hình 3. 3 Use case quản lý môn học

3.5.1.4 Use Case Quản lý chương trình



Hình 3. 4 Use case quản lý chương trình

3.5.1.5 Use Case Quản lý giảng viên



Hình 3. 5 Use case quản lý giảng viên

3.5.1.6 Danh sách mô tả các chức năng

Bảng 3. 1 Danh sách mô tả các chức năng

| ID | Tên Use Case | Mô tả ngắn gọn về Use case | Ghi chú |
|-------|-------------------|---|---------|
| UC001 | Đăng nhập | Trưởng bộ môn và giảng viên đăng nhập vào hệ thống | |
| UC002 | Tìm kiếm môn học | Trưởng bộ môn và giảng viên tìm kiếm môn học | |
| UC003 | Chỉnh sửa môn học | Trưởng bộ môn và giảng viên chỉnh sửa môn học | |
| UC004 | In môn học | Cho phép Trưởng bộ môn và Giảng viên môn học | |
| UC005 | Tạo môn học | Trưởng bộ môn tạo môn học | |
| UC006 | Sao chép | Cho phép Trưởng bộ môn và giảng viên sao chép nội dung môn học và tạo môn học mới | |

| | | | |
|-------|------------------------|---|--|
| UC007 | Xóa môn học | Cho phép Trưởng bộ môn xóa môn học khỏi hệ thống | |
| UC008 | Tạo chương trình | Cho phép Trưởng bộ môn tạo chương trình mới | |
| UC009 | Chỉnh sửa chương trình | Cho phép Trưởng bộ môn chỉnh sửa chương trình đã có | |
| UC010 | Sao chép chương trình | Cho phép Trưởng bộ môn sao chép chương trình để tạo chương trình mới | |
| UC011 | Xóa chương trình | Cho phép Trưởng bộ môn xóa chương trình khỏi hệ thống | |
| UC012 | In chương trình | Cho phép Trưởng bộ môn và Giảng viên in chương trình đã có | |
| UC013 | Thêm giảng viên | Cho phép Trưởng bộ môn thêm giảng viên mới vào hệ thống | |
| UC014 | Reset mật khẩu | Cho phép Trưởng bộ môn reset mật khẩu đăng nhập của giảng viên về mật khẩu mặc định | |
| UC015 | Xóa giảng viên | Cho phép Trưởng bộ môn xóa giảng viên khỏi hệ thống | |
| UC016 | Tạo yêu cầu giảng viên | Cho phép trưởng bộ môn tạo yêu cầu cho giảng viên chỉnh sửa môn học | |

3.5.2 Đặc tả use case

3.5.2.1 UC001 – Use Case Đăng nhập

Bảng 3. 2 Đặc tả UC001 – Đăng nhập

| | |
|--------------------------------------|---|
| ID | UC001 |
| Use case | Đăng nhập |
| Tác nhân | Trưởng bộ môn / Giảng viên |
| Mô tả | Cho phép người dùng đăng nhập vào hệ thống (Trưởng bộ môn, Giảng viên) |
| Điều kiện tiên quyết | Người dùng có tài khoản trong hệ thống |
| Điều kiện sau | Nếu use case thành công, người dùng đăng nhập vào hệ thống theo phân quyền. Ngược lại nếu không thành công, trạng thái của hệ thống không thay đổi. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng truy cập vào trang web | 2. Hệ thống hiện thị form đăng nhập |
| 3. Người dùng nhập email và mật khẩu | |
| 4. Người dùng nhấn nút “Sign in” | 5. Hệ thống kiểm tra xác minh thông tin tài khoản |
| | 6. Hệ thống kiểm tra xem là tài khoản của Trưởng bộ môn hay Giảng viên |
| | 7. Hệ thống hiển thị giao diện dashboard của Trưởng bộ môn |
| Luồng sự kiện thay thế | |
| | 5.1 Hệ thống hiển thị thông báo lỗi |

| | |
|-------------------------|--|
| 5.2 Người dùng xác nhận | 5.3 Hệ thống quay lại bước 2 |
| | 6.1 Hệ thống hiển thị giao diện dashboard của Giảng viên |

3.5.2.2 UC002 – Use Case Tìm kiếm môn học

Bảng 3. 3 Đặc tả UC002 - Tìm kiếm môn học

| | |
|---|--|
| ID | UC002 |
| Use case | Tìm kiếm môn học |
| Tác nhân | Trưởng bộ môn / Giảng viên |
| Mô tả | Người dùng tìm kiếm và xem thông tin về các môn học trong hệ thống |
| Điều kiện tiên quyết | Hệ thống đã khởi động và giao diện tìm kiếm đã hiển thị |
| Điều kiện sau | Người dùng đã tìm kiếm và xem thông tin về môn học trong hệ thống |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng nhập từ khóa vào ô tìm kiếm | 2. Hệ thống truy xuất và tìm kiếm môn học dựa trên từ khóa đã nhập |
| | 3. Hệ thống hiển thị danh sách các môn học phù hợp với từ khóa đã tìm kiếm |
| Luồng sự kiện thay thế | |
| | 2.1 Hệ thống không hiển thị |

3.5.2.3 UC003 – Usecase Chỉnh sửa môn học

Bảng 3. 4 Đặc tả UC003 – Chỉnh sửa môn học

| | |
|-----------|-------|
| ID | UC003 |
|-----------|-------|

| | |
|--|---|
| Use case | Chỉnh sửa môn học |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền chỉnh sửa thông tin về một môn học trong hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống • Môn học cần chỉnh sửa đã được tạo và tồn tại trong hệ thống |
| Điều kiện sau | <ul style="list-style-type: none"> • Thông tin về môn học đã được cập nhật và lưu lại trong hệ thống • Người dùng đã hoàn thành quá trình chỉnh sửa môn học |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn môn học cần chỉnh sửa | 2. Hệ thống hiển thị thông tin môn học mà người dùng vừa chọn |
| 3. Người dùng bấm nút “Chỉnh sửa môn học” | 4. Hệ thống hiển thị form chỉnh sửa môn học |
| 5. Người dùng thực hiện cập nhật thông tin mới cho môn học | |
| 6. Người dùng bấm nút “Lưu” để lưu các thay đổi | 7. Hệ thống thông báo đã chỉnh sửa thành công |
| | |
| Luồng sự kiện thay thế | |

| | |
|--|--|
| | |
|--|--|

3.5.2.4 UC004 – Usecase In môn học

Bảng 3. 5 Đặc tả UC004 – In môn học

| | |
|---|--|
| ID | UC004 |
| Use case | In môn học |
| Tác nhân | Trưởng bộ môn / Giảng viên |
| Mô tả | Người dùng có quyền chỉnh xuất thông tin môn học dưới dạng file PDF. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Môn học cần xuất file PDF đã được tạo và tồn tại trong hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> • Thông tin về môn học đã được xuất thành công và được tải về máy người dùng. • Người dùng đã hoàn thành quá trình xuất file PDF môn học. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn môn học cần xuất file PDF | |
| 2. Người dùng nhấn nút “Export PDF” trên môn học đó | 3. Hệ thống thực hiện export file pdf cho người dùng tải về |
| Luồng sự kiện thay thế | |
| | |

3.5.2.5 UC005 – Usecase Tạo môn học

Bảng 3. 6 Đặc tả UC005 – Tạo môn học

| | |
|---|--|
| ID | UC005 |
| Use case | Tạo môn học |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền tạo mới môn học trong hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền tạo môn học |
| Điều kiện sau | <ul style="list-style-type: none"> • Môn học mới đã tạo thành công và lưu trữ trong cơ sở dữ liệu. • Người dùng đã hoàn thành quá trình tạo môn học. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý môn học | 2. Hệ thống hiển thị giao diện quản lý môn học |
| 3. Người dùng bấm vào nút “Tạo môn học” ở giao diện thông tin môn học | 4. Hệ thống hiển thị một form cho người dùng nhập thông tin môn học. |
| 5. Người dùng điền thông tin cần thiết cho môn học | |

| | |
|---|---|
| 6. Người dùng nhấn nút “Lưu” để tiến hành tạo mới môn học | 7. Hệ thống lưu thông tin môn học vào cơ sở dữ liệu và thông báo cho người dùng đã tạo thành công |
| Luồng sự kiện thay thế | |
| | |

3.5.2.6 UC006 – Usecase Sao chép môn học

Bảng 3. 7 Đặc tả UC006 – Sao chép môn học

| | |
|--|---|
| ID | UC006 |
| Use case | Sao chép môn học |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền sao chép môn học trong hệ thống để tạo môn học mới. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền sao chép |
| Điều kiện sau | <ul style="list-style-type: none"> • Môn học mới đã được tạo thành công và lưu trữ trong cơ sở dữ liệu. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý môn học | 2. Hệ thống hiển thị giao diện quản lý môn học |
| 3. Người dùng bấm vào nút “Sao chép môn học” ở giao diện | 4. Hệ thống hiển thị một form cho người dùng chỉnh sửa thông tin của môn học mới |

| | |
|---|---|
| thông tin môn học | |
| 5. Người dùng chỉnh sửa nội dung của môn học mới | |
| 6. Người dùng nhấn nút “Lưu” để tiến hành tạo mới môn học | 7. Hệ thống lưu thông tin môn học vào cơ sở dữ liệu và thông báo cho người dùng đã tạo thành công |
| Luồng sự kiện thay thế | |
| | |

3.5.2.7 UC007 – Usecase Xóa môn học

Bảng 3. 8 Đặc tả UC007 – Xóa môn học

| | |
|-----------------------------|--|
| ID | UC007 |
| Use case | Xóa môn học |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền xóa môn học trong hệ thống. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. Người dùng có quyền xóa môn học trong hệ thống. Môn học có tồn tại trong hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> Môn học đã được xóa khỏi hệ thống và không còn hiển thị trong danh sách các môn học. Người dùng hoàn thành quá trình xóa môn học. |
| Luồng sự kiện chính | |
| Actor | System |

| | |
|---|--|
| 1. Người dùng chọn chức năng quản lý môn học | 2. Hệ thống hiển thị giao diện quản lý môn học |
| 3. Người dùng chọn môn học từ danh sách các môn học | |
| 4. Người dùng bấm vào nút “Xóa môn học” ở giao diện thông tin môn học | 5. Hệ thống thông báo đã xóa thành công |
| Luồng sự kiện thay thế | |
| | |

3.5.2.8 UC008 – Usecase Tạo mới chương trình

Bảng 3. 9 Đặc tả UC008 - Tạo mới chương trình

| | |
|-----------------------------|---|
| ID | UC008 |
| Use case | Tạo mới chương trình |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền tạo mới chương trình học cho một khóa học. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. Người dùng có quyền tạo mới chương trình học |
| Điều kiện sau | <ul style="list-style-type: none"> Chương trình học mới đã được tạo và lưu vào trong hệ thống |
| Luồng sự kiện chính | |
| Actor | System |

| | |
|--|---|
| 1. Người dùng chọn chức năng quản lý chương trình | 2. Hệ thống hiển thị giao diện quản lý chương trình |
| 3. Người dùng nhấn vào biểu tượng dấu cộng trên giao diện quản lý chương trình | 4. Hệ thống hiển thị form để tạo mới chương trình. |
| 5. Người dùng nhập tên chương trình, số lượng học kì | 6. Hệ thống dựa vào số lượng học kì vừa nhập để tạo các lựa chọn trong thanh dropdown học kì. |
| 7. Người dùng chọn học kì | |
| 8. Người dùng chọn các môn học trong danh sách để thêm vào học kì vừa chọn, và tiếp tục với các học kì tiếp theo | |
| 9. Người dùng bấm nút “Tạo chương trình” | 10. Hệ thống lưu chương trình trong cơ sở dữ liệu và thông báo đã tạo thành công |
| Luồng sự kiện thay thế | |
| | |

3.5.2.9 UC009 – Usecase Chỉnh sửa chương trình

Bảng 3. 10 Đặc tả UC009 – Chỉnh sửa chương trình

| | |
|---|---|
| ID | UC009 |
| Use case | Chỉnh sửa chương trình |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền chỉnh sửa chương trình học cho một khóa học. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền chỉnh sửa chương trình học |
| Điều kiện sau | <ul style="list-style-type: none"> • Chương trình học mới đã được chỉnh sửa và lưu vào trong hệ thống |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý chương trình | 2. Hệ thống hiển thị giao diện quản lý chương trình |
| 3. Người dùng nhấn vào biểu tượng chỉnh sửa trên giao diện quản lý chương trình | 4. Hệ thống hiển thị form chỉnh sửa chương trình. |
| 5. Người dùng thực hiện sửa đổi trên form | |

| | |
|------------------------------------|--|
| 6. Người dùng nhấn nút “Chỉnh sửa” | 7. Hệ thống cập nhật thông tin chương trình vào hệ thống và thông báo cập nhật thành công. |
| Luồng sự kiện thay thế | |
| | |

3.5.2.10 UC010 – Usecase Sao chép chương trình

Bảng 3. 11 Đặc tả UC010 – Sao chép chương trình

| | |
|--|--|
| ID | UC010 |
| Use case | Sao chép chương trình |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền sao chép chương trình học cho một khóa học. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền sao chép chương trình học |
| Điều kiện sau | <ul style="list-style-type: none"> • Chương trình học mới đã được tạo và lưu vào trong hệ thống |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý chương trình | 2. Hệ thống hiển thị giao diện quản lý chương trình |
| 3. Người dùng nhấn vào biểu tượng “Sao chép” trên giao diện quản lý chương trình | 4. Hệ thống hiển thị một form cho người dùng chỉnh sửa thông tin của chương trình mới. |

| | |
|---|---|
| 5. Người dùng thực hiện sửa đổi trên form | |
| 6. Người dùng nhấn nút “Sao chép” | 7. Hệ thống tạo chương trình mới lưu vào hệ thống và thông báo cập nhật thành công. |
| Luồng sự kiện thay thế | |
| | |

3.5.2.11 UC011 – Usecase Xóa chương trình

Bảng 3. 12 Đặc tả UC011 – Xóa chương trình

| | |
|---|---|
| ID | UC011 |
| Use case | Xóa chương trình |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền xóa chương trình đã có trong hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. Người dùng có quyền xóa chương trình học. Chương trình đã tồn tại trong hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> Chương trình học đã bị xóa và không hiển thị trong danh sách chương trình học. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý chương trình | 2. Hệ thống hiển thị giao diện quản lý chương trình |
| 3. Người dùng nhấn vào biểu | 4. Hệ thống xóa chương trình khỏi cơ sở dữ liệu và thông báo đã xóa. |

| | |
|--|--|
| tượng thùng rác bên cạnh chương trình cần xóa | |
| Luồng sự kiện thay thế | |
| | |

3.5.2.12 UC012 – Usecase In chương trình

Bảng 3. 13 Đặc tả UC012 – In chương trình

| | |
|---|--|
| ID | UC012 |
| Use case | In chương trình |
| Tác nhân | Trưởng bộ môn/Giảng viên |
| Mô tả | Người dùng có quyền in chương trình đã có trong hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền in chương trình học. • Chương trình đã tồn tại trong hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> • Thông tin về chương trình học đã được xuất thành công và được tải về máy người dùng. • Người dùng đã hoàn thành quá trình xuất file PDF môn học |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý chương trình | 2. Hệ thống hiển thị giao diện quản lý chương trình |
| 3. Người dùng nhấn vào biểu | 4. Hệ thống hiển thị giao diện in chương trình cho người dùng xem trước |

| | |
|--|--|
| tượng in bên cạnh chương trình cần in | |
| 5. Người dùng nhấn nút In trên giao diện xem trước | 6. Hệ thống thực hiện export file PDF cho người dùng tải về. |
| Luồng sự kiện thay thế | |
| | |

3.5.2.13 UC013 – Thêm giảng viên

Bảng 3. 14 Đặc tả UC013 – Thêm giảng viên

| | |
|------------------------------|---|
| ID | UC013 |
| Use case | Thêm giảng viên |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền thêm giảng viên mới vào hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. Người dùng có quyền thêm giảng viên mới vào hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> Giảng viên mới đã được thêm và lưu vào cơ sở dữ liệu. Người dùng hoàn tất quá trình thêm giảng viên. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng | 2. Hệ thống hiển thị giao diện quản lý giảng viên |

| | |
|---|---|
| quản lý giảng viên | |
| 3. Người dùng nhấn vào biểu tượng dấu cộng trong giao diện quản lý giảng viên | 4. Hệ thống hiển thị giao diện thêm giảng viên |
| 5. Người dùng nhập thông tin giảng viên | |
| 6. Người dùng nhấn nút “Thêm” trên giao diện thêm giảng viên | 7. Hệ thống lưu dữ liệu giảng viên mới trong cơ sở dữ liệu và thông báo đã thêm thành công. |
| 8. | |
| Luồng sự kiện thay thế | |
| | |

3.5.2.14 UC014 – Reset mật khẩu giảng viên

Bảng 3. 15 Đặc tả UC014 – Reset mật khẩu giảng viên

| | |
|-----------------------------|---|
| ID | UC014 |
| Use case | Reset mật khẩu |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền reset mật khẩu giảng viên về mật khẩu mặc định. |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. |

| | |
|---|--|
| | <ul style="list-style-type: none"> Người dùng có quyền reset mật khẩu của giảng viên. |
| Điều kiện sau | <ul style="list-style-type: none"> Mật khẩu của giảng viên được reset về mặc định |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý giảng viên | 2. Hệ thống hiển thị giao diện quản lý giảng viên |
| 3. Người dùng nhấn vào biểu tượng mũi tên quay tròn bên cạnh giảng viên được chọn | 4. Hệ thống hiển thị giao diện xác nhận reset mật khẩu |
| 5. Người dùng nhấn nút trên giao diện xác nhận | 6. Hệ thống reset mật khẩu của giảng viên và thông báo đã reset thành công |
| Luồng sự kiện thay thế | |
| | 6.1 Hệ thống tắt giao diện xác nhận reset mật khẩu và không làm gì khác |

3.5.2.15 UC015 – Xóa giảng viên

Bảng 3. 16 Đặc tả UC015 – Xóa giảng viên

| | |
|-----------------------------|---|
| ID | UC015 |
| Use case | Xóa giảng viên |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền xóa giảng viên khỏi hệ thống |
| Điều kiện tiên quyết | <ul style="list-style-type: none"> Người dùng đã đăng nhập vào hệ thống. |

| | |
|---|---|
| | <ul style="list-style-type: none"> Người dùng có quyền xóa giảng viên khỏi hệ thống. |
| Điều kiện sau | <ul style="list-style-type: none"> Mọi thông tin của giảng viên được xóa và không xuất hiện trên danh sách giảng viên. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý giảng viên | 2. Hệ thống hiển thị giao diện quản lý giảng viên |
| 3. Người dùng nhấn vào biểu tượng thùng rác bên cạnh giảng viên được chọn | 4. Hệ thống hiển thị giao diện xác xóa giảng viên |
| 5. Người dùng nhấn nút trên giao diện xác nhận | 6. Hệ thống xóa thông tin giảng viên khỏi hệ thống và thông báo đã xóa thành công. |
| Luồng sự kiện thay thế | |
| | 6.1 Hệ thống tắt giao diện xác nhận xóa giảng viên và không làm gì khác |

3.5.2.16 UC016 – Tạo yêu cầu giảng viên

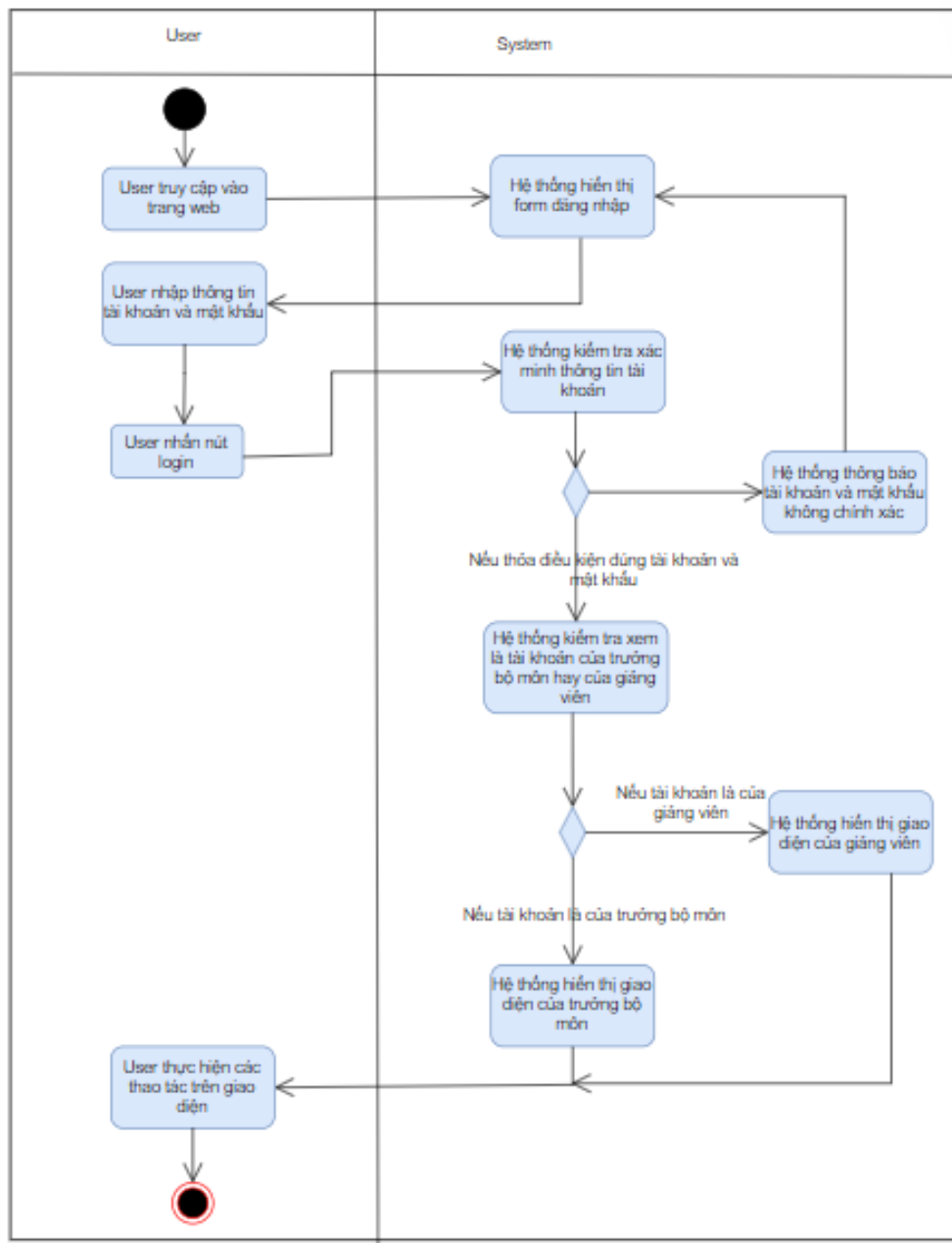
Bảng 3. 17 Đặc tả UC016 – Tạo yêu cầu giảng viên

| | |
|-----------------|--|
| ID | UC016 |
| Use case | Tạo yêu cầu giảng viên |
| Tác nhân | Trưởng bộ môn |
| Mô tả | Người dùng có quyền tạo yêu cầu cho giảng viên chỉnh sửa môn học |

| | |
|--|--|
| Điều kiện tiên quyết | <ul style="list-style-type: none"> • Người dùng đã đăng nhập vào hệ thống. • Người dùng có quyền tạo yêu cầu cho giảng viên chỉnh sửa môn học. |
| Điều kiện sau | <ul style="list-style-type: none"> • Trưởng bộ môn tạo xong yêu cầu và giảng viên nhận được yêu cầu chỉnh sửa môn học. |
| Luồng sự kiện chính | |
| Actor | System |
| 1. Người dùng chọn chức năng quản lý môn học | 2. Hệ thống hiển thị giao diện quản lý môn học. |
| 3. Người dùng nhấn nút “Tạo yêu cầu” trong giao diện quản lý môn học | 4. Hệ thống hiển thị giao tạo yêu cầu cho người dùng. |
| 5. Người dùng nhập tên môn học và tên giảng viên muốn gửi yêu cầu | |
| 6. Người dùng nhấn nút “Gửi yêu cầu” | 7. Hệ thống gửi yêu cầu cho giảng viên |
| Luồng sự kiện thay thế | |
| | |

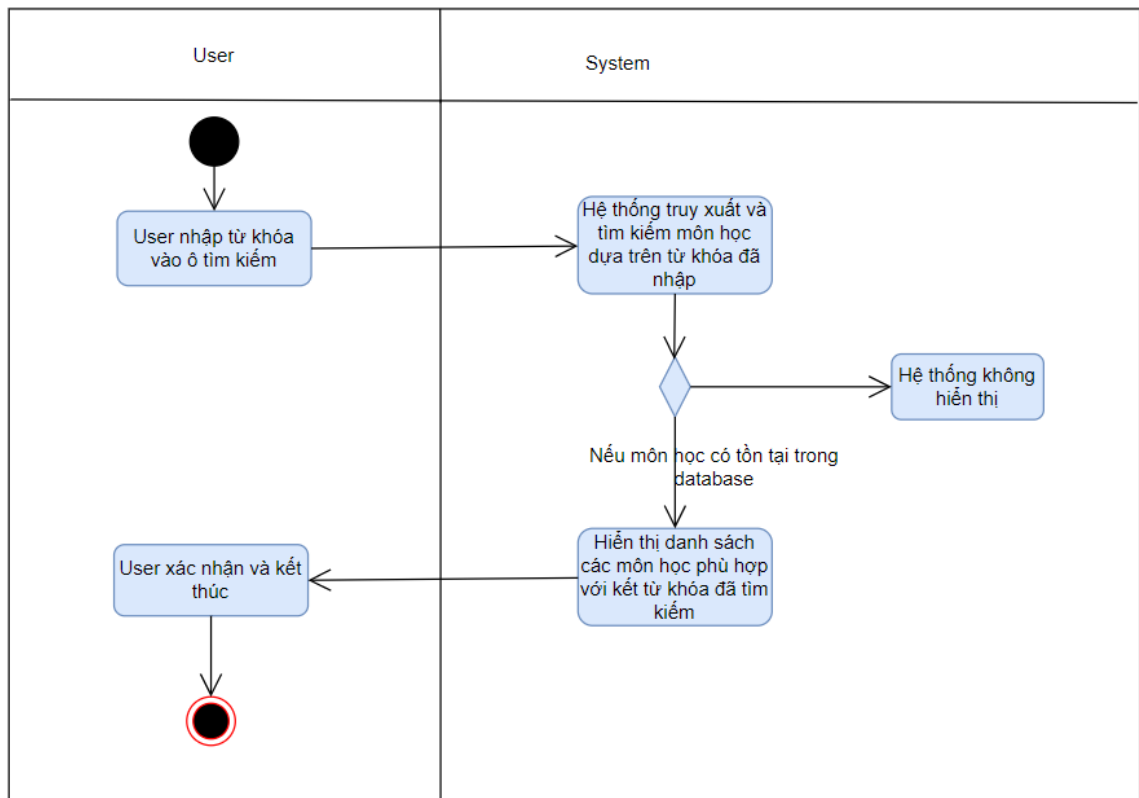
3.5.3 Sơ đồ activity

3.5.3.1 Sơ đồ activity use case Đăng nhập



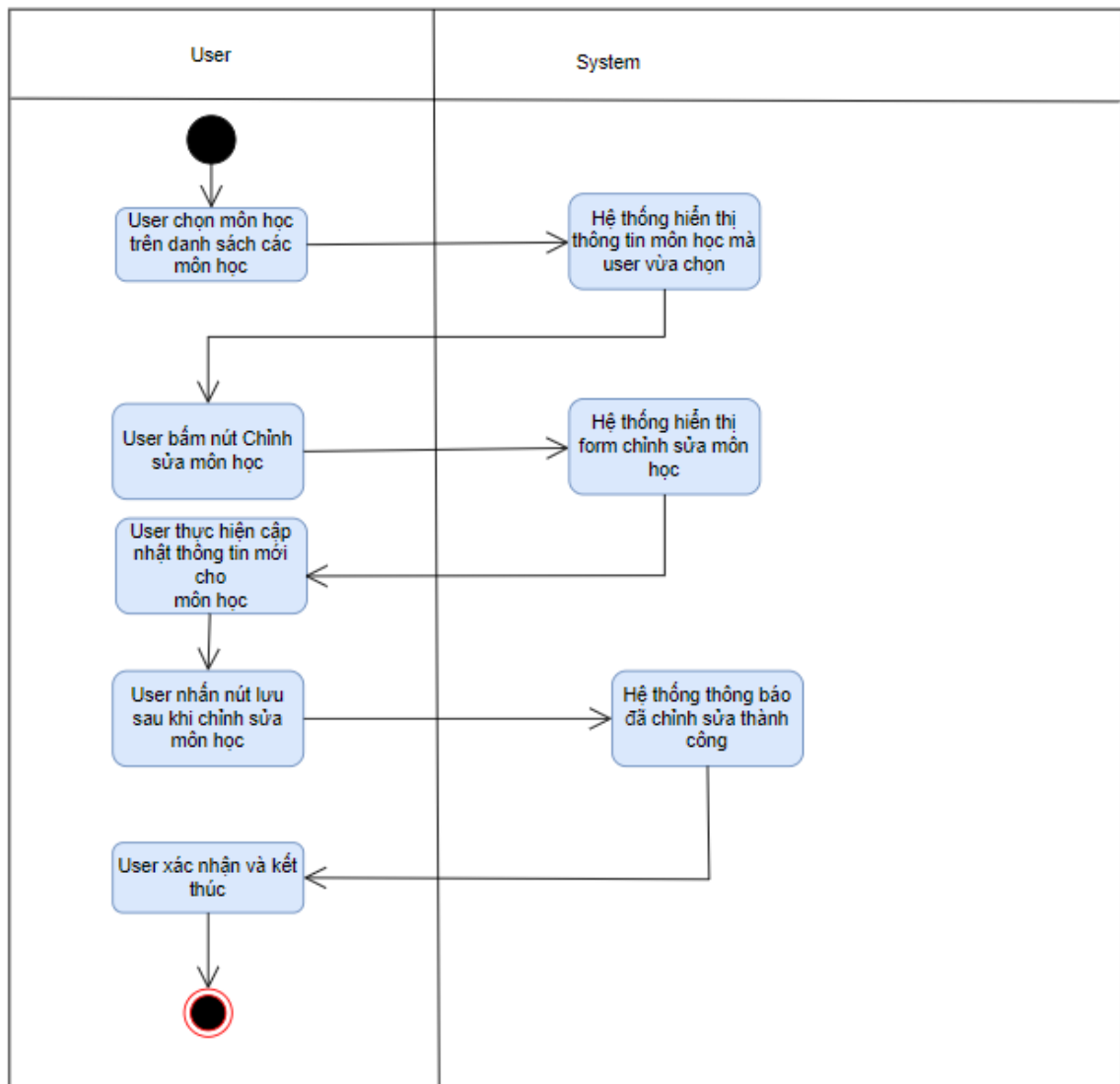
Hình 3. 6 Sơ đồ activity biểu diễn use case đăng nhập

3.5.2.2 Sơ đồ activity use case tìm kiếm môn học



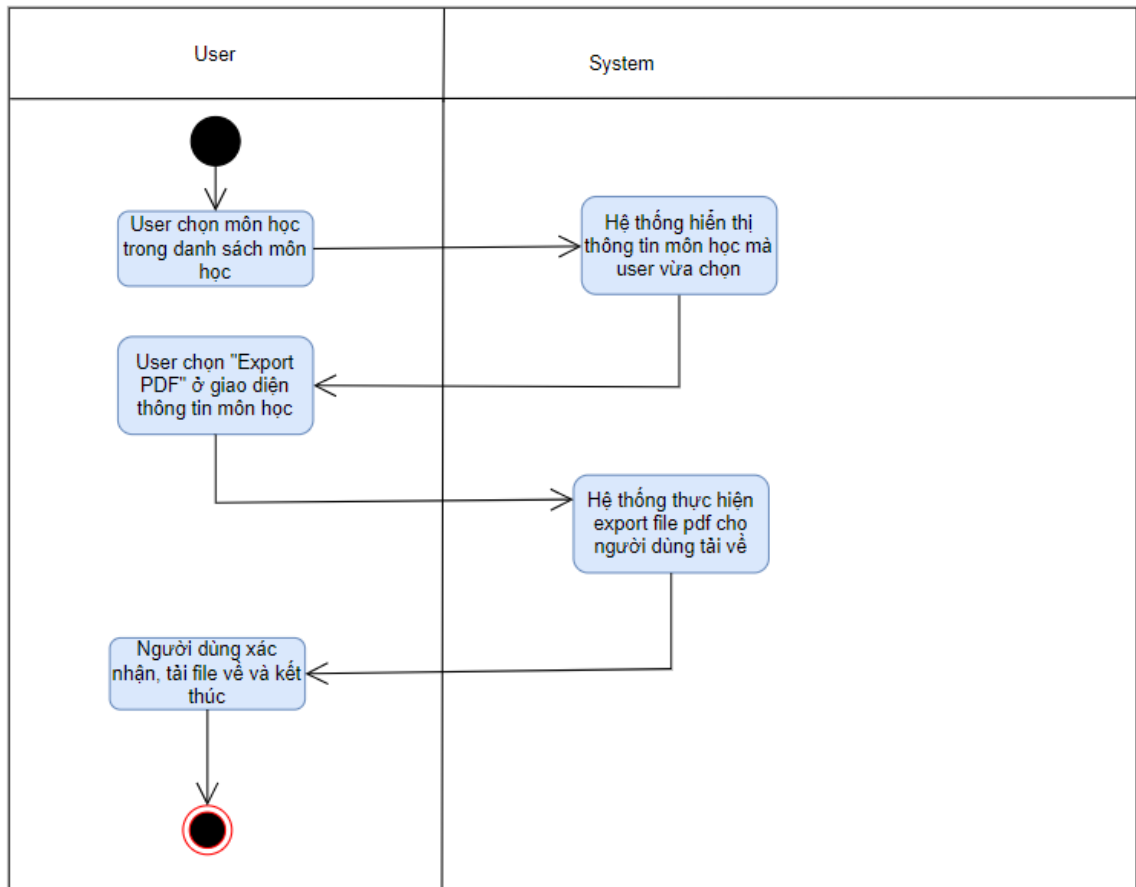
Hình 3. 7 Sơ đồ activity biểu diễn use case tìm kiếm môn học

3.5.2.3 Sơ đồ activity use case chỉnh sửa môn học



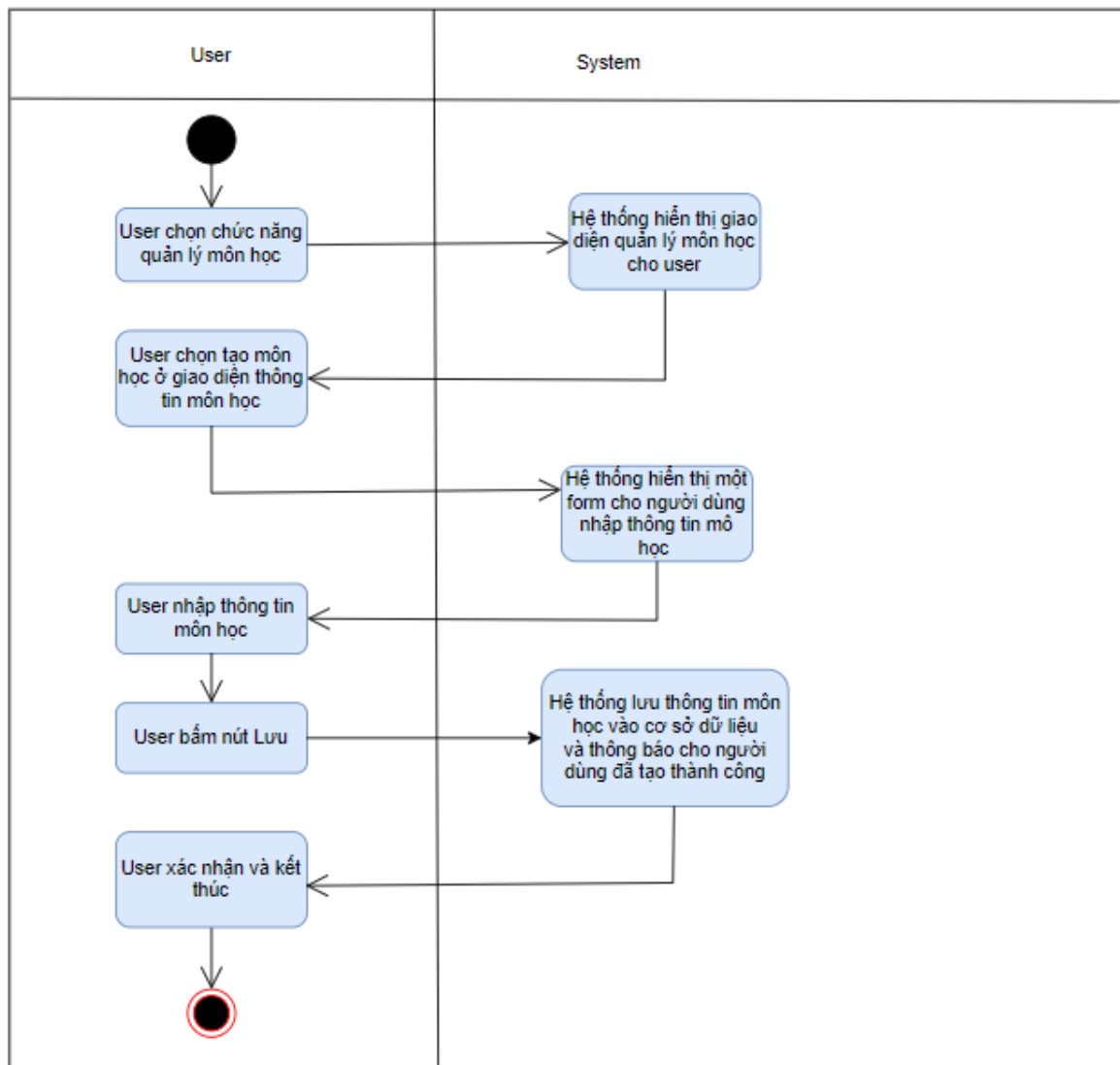
Hình 3. 8 Sơ đồ activity biểu diễn use case chỉnh sửa môn học

3.5.2.4 Sơ đồ activity use case in môn học



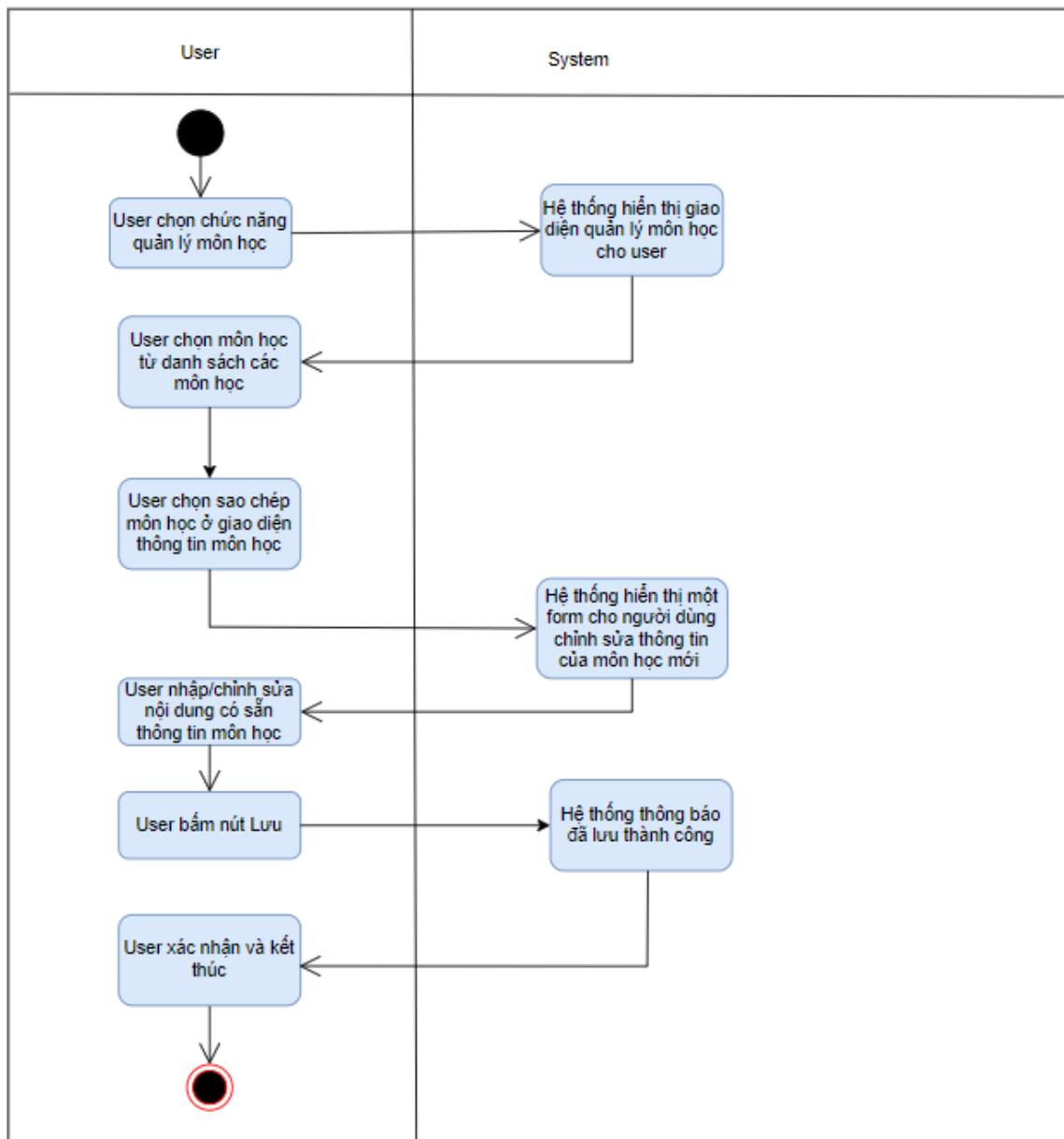
Hình 3. 9 Sơ đồ activity biểu diễn use case in môn học

3.5.2.5 Sơ đồ activity use case tạo môn học



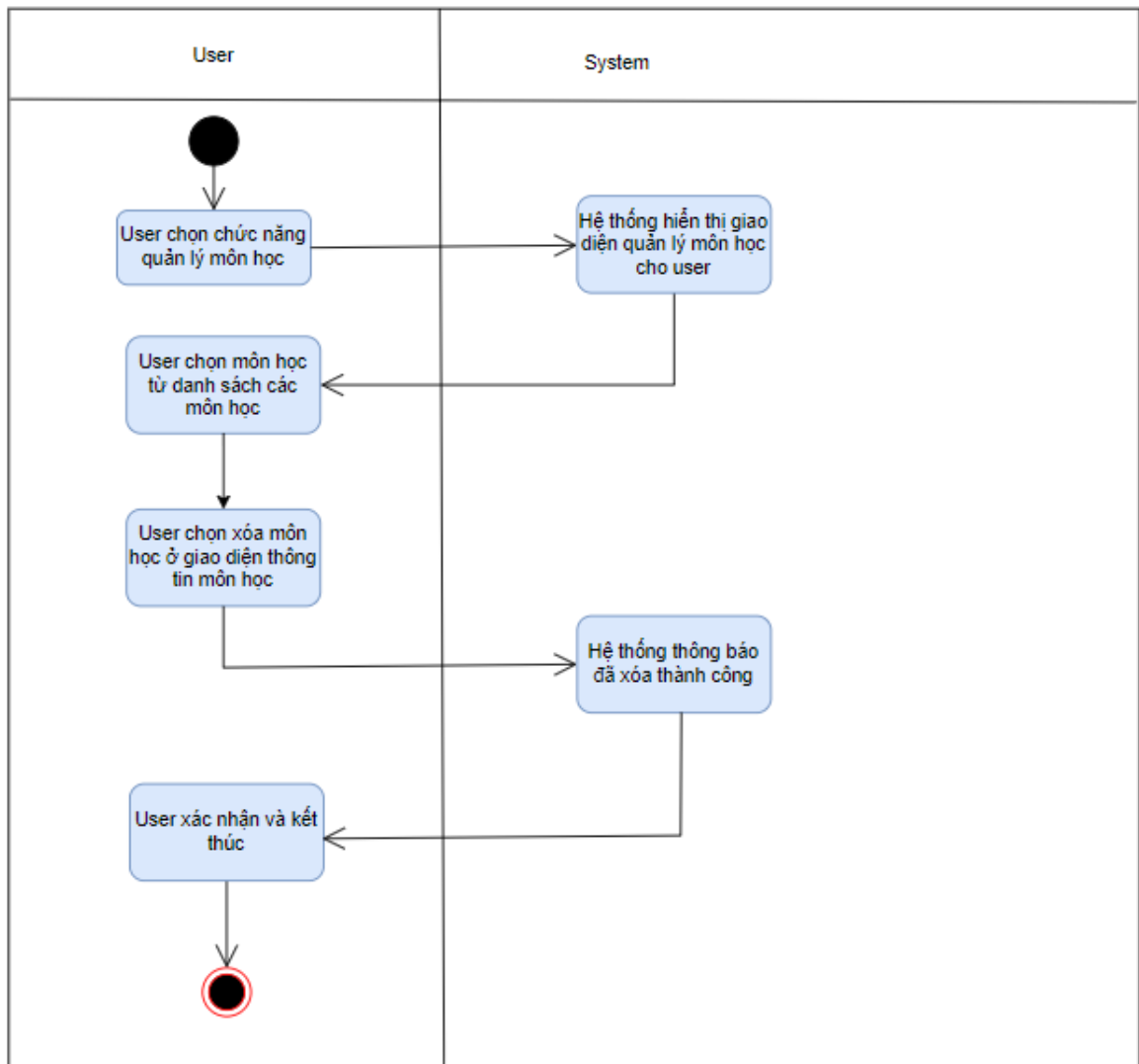
Hình 3. 10 Sơ đồ activity biểu diễn use case tạo môn học

3.5.2.6 Sơ đồ activity use case sao chép môn học



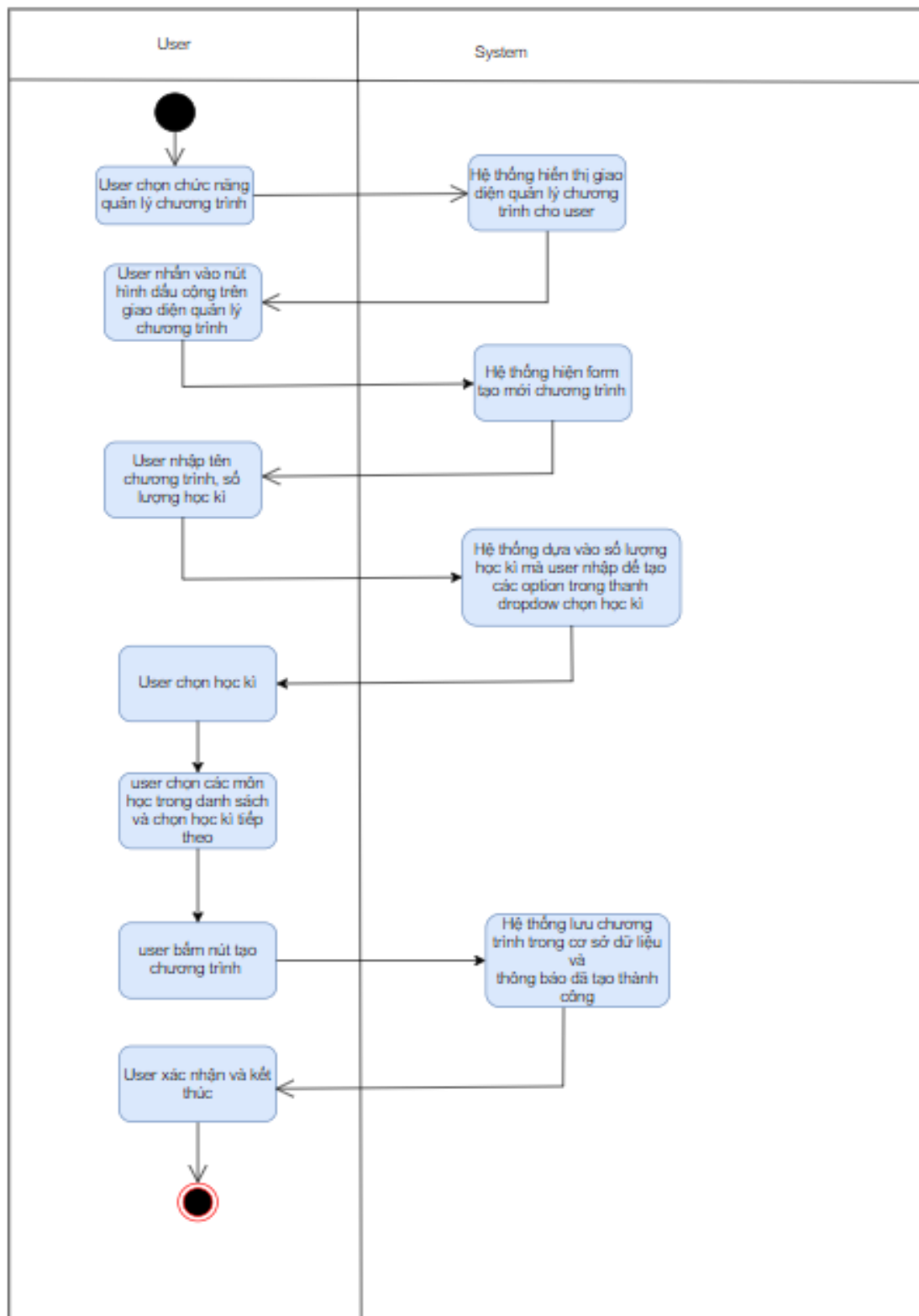
Hình 3. 11 Sơ đồ activity biểu diễn use case sao chép môn học

3.5.2.7 Sơ đồ activity use case xóa môn học



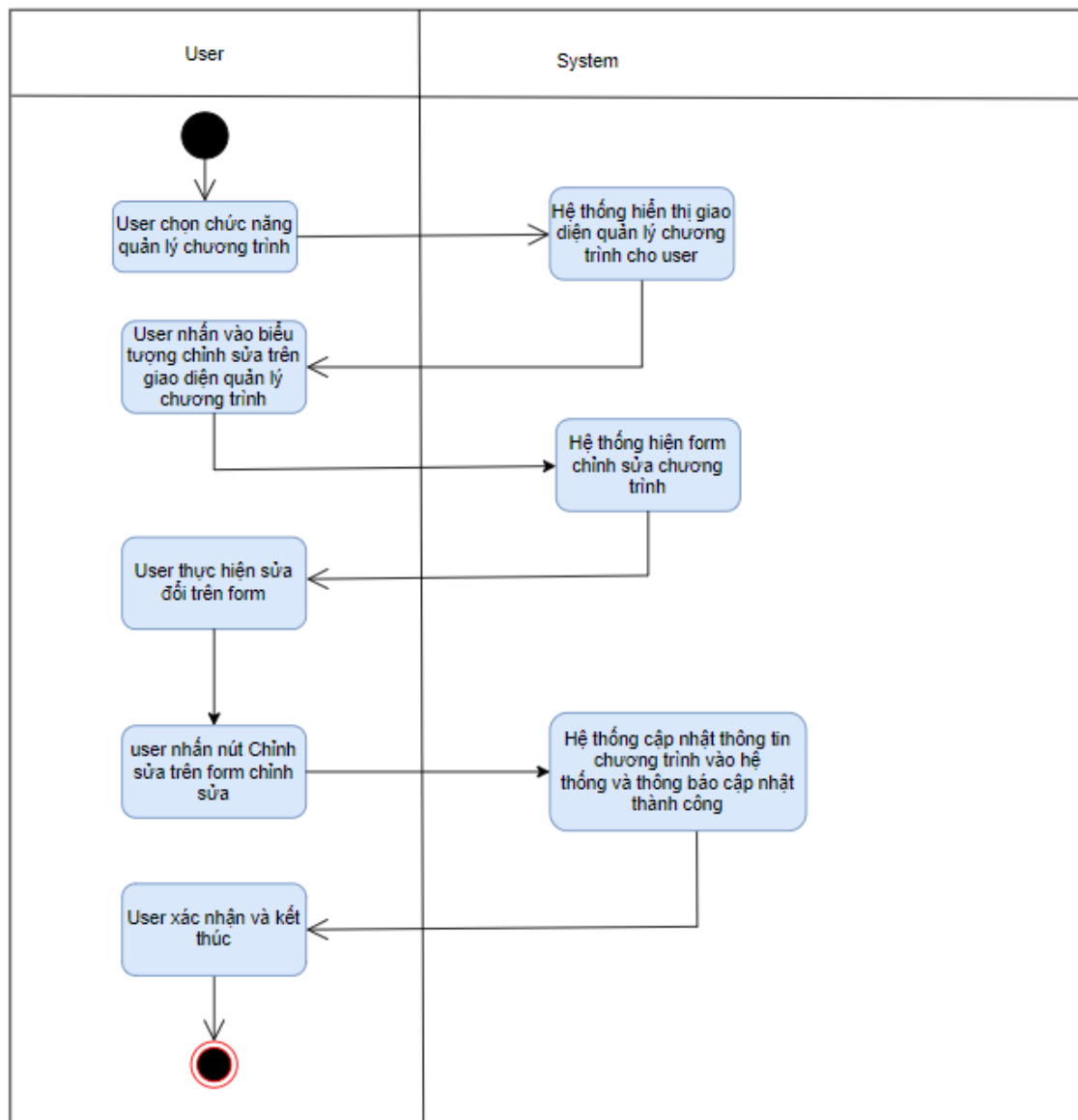
Hình 3. 12 Sơ đồ activity biểu diễn use case xóa môn học

3.5.2.8 Sơ đồ activity use case tạo mới chương trình



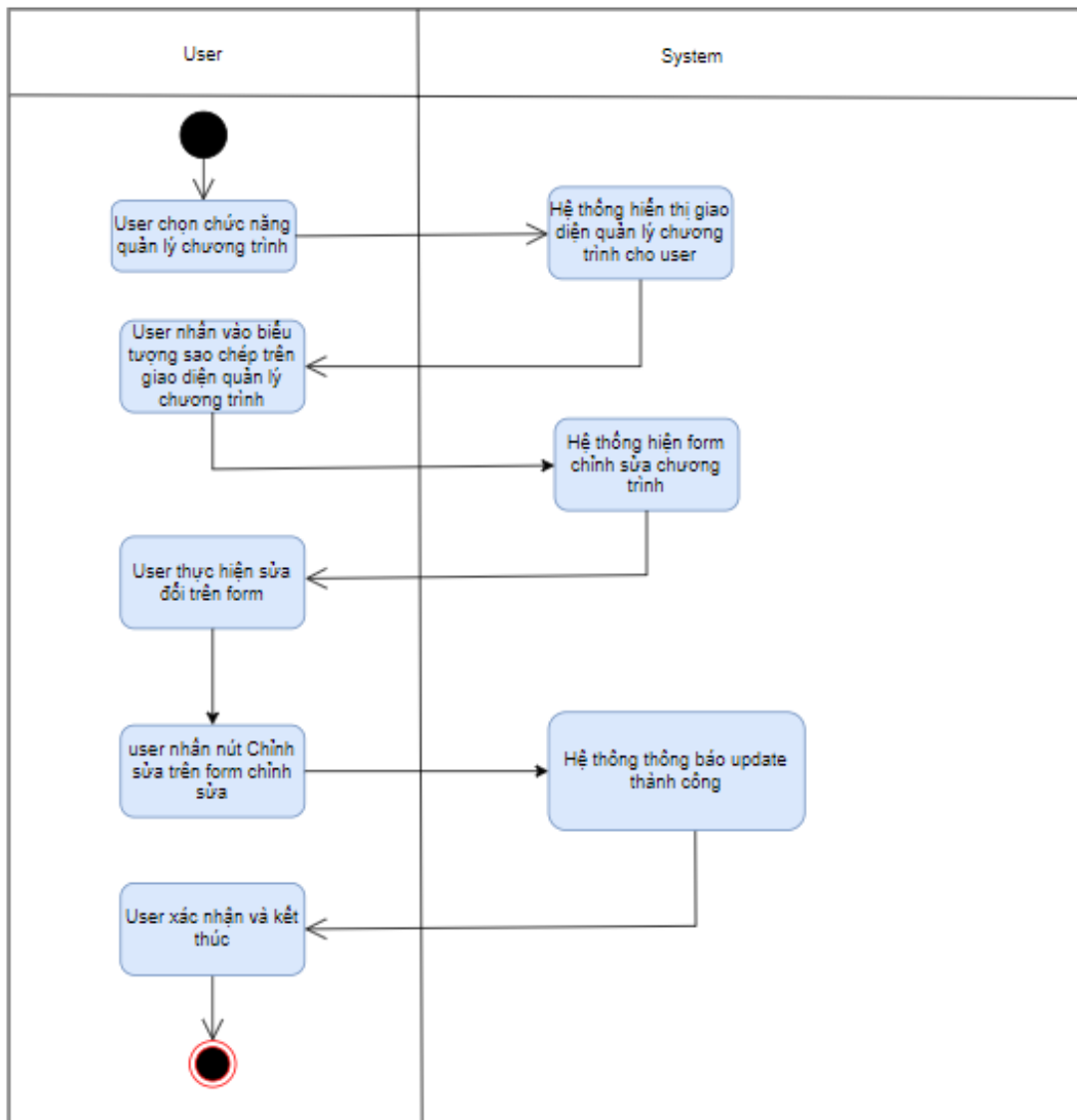
Hình 3. 13 Sơ đồ activity biểu diễn use case tạo mới chương trình

3.5.2.9 Sơ đồ activity use case chỉnh sửa chương trình



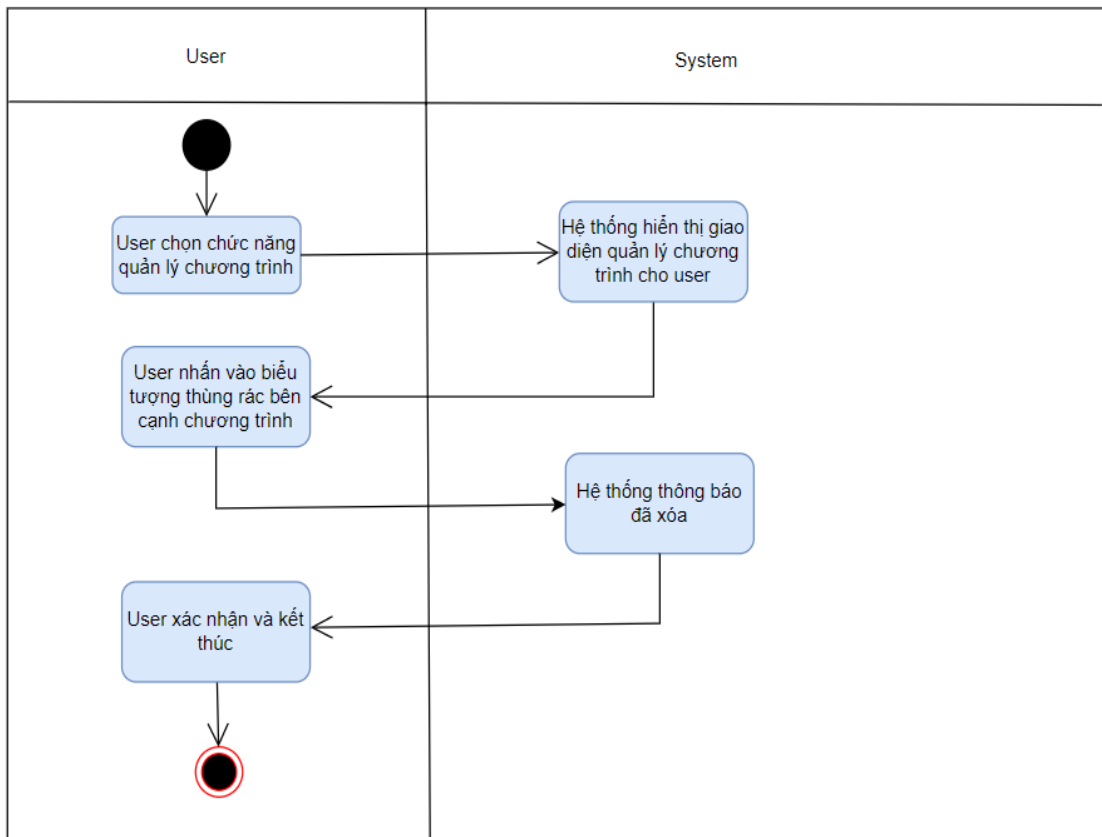
Hình 3. 14 Sơ đồ activity biểu diễn use case chỉnh sửa chương trình

3.5.2.10 Sơ đồ activity use case sao chép chương trình



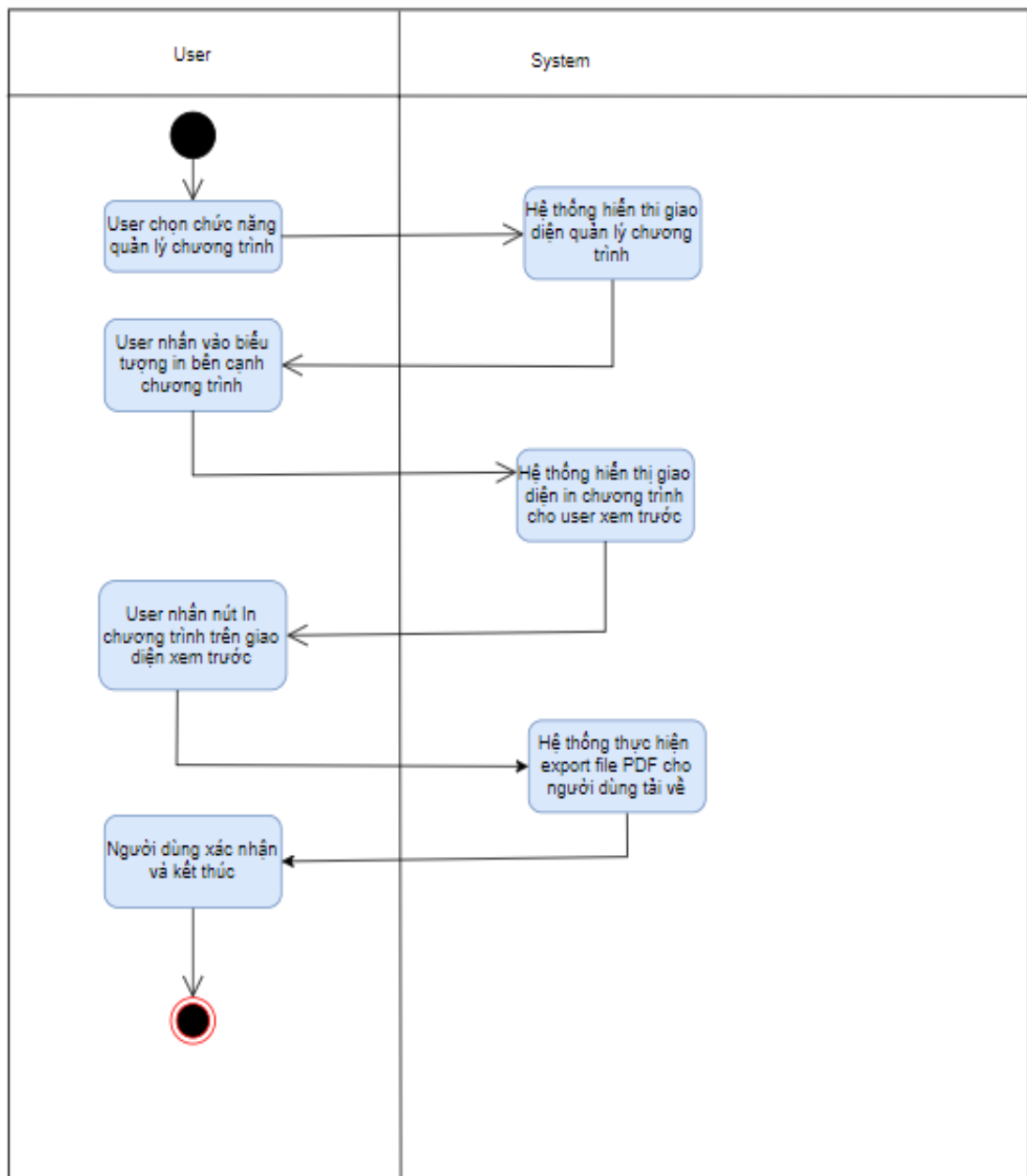
Hình 3. 15 Sơ đồ activity biểu diễn use case sao chép chương trình

3.5.2.11 Sơ đồ activity use case xóa chương trình



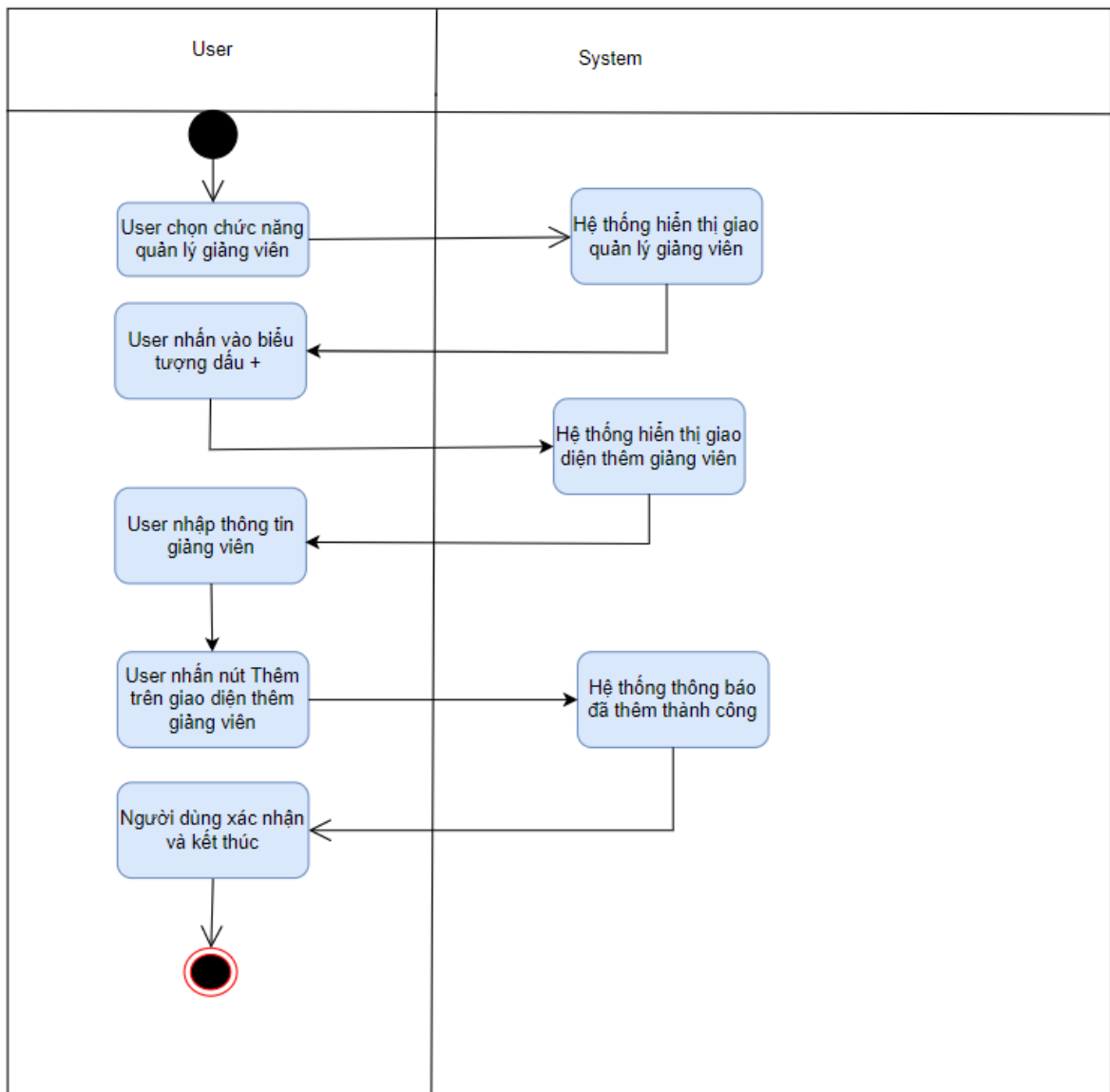
Hình 3. 16 Sơ đồ activity biểu diễn use case xóa chương trình

3.5.2.12 Sơ đồ activity use case in chương trình



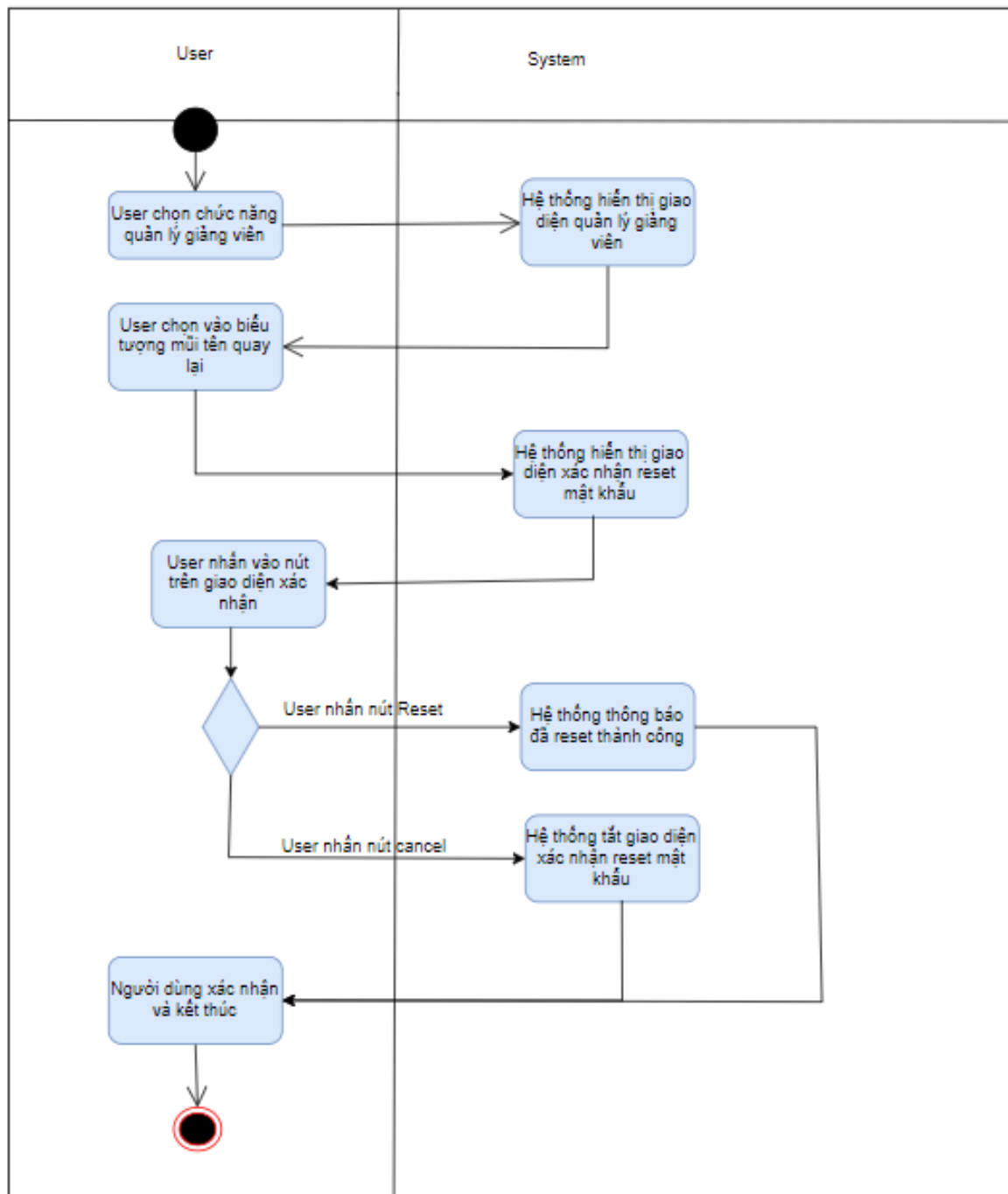
Hình 3. 17 Sơ đồ activity biểu diễn use case in chương trình

3.5.2.13 Sơ đồ activity use case thêm giảng viên



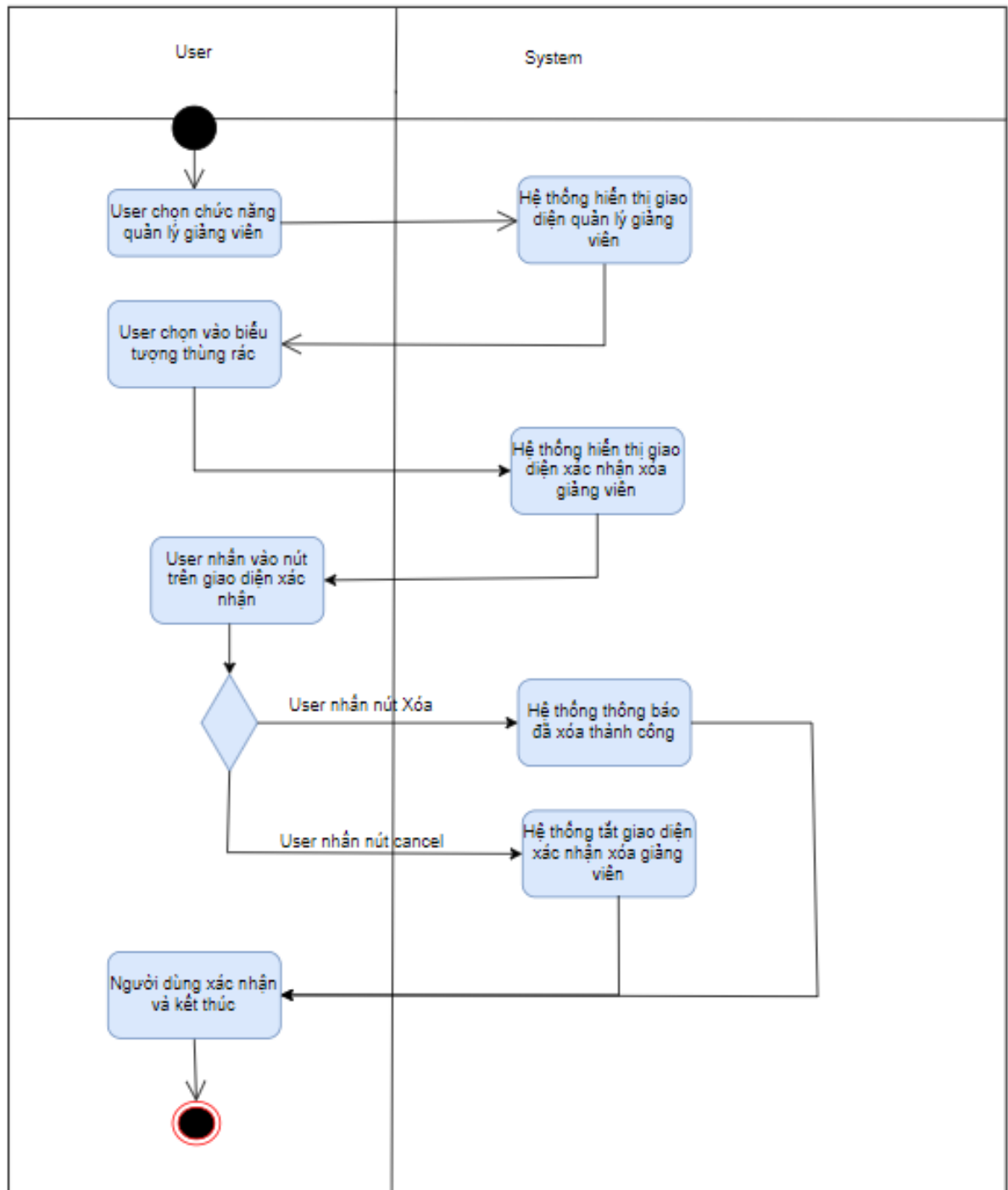
Hình 3. 18 Sơ đồ activity biểu diễn use case thêm giảng viên

3.5.2.14 Sơ đồ activity use case reset mật khẩu tài khoản giảng viên



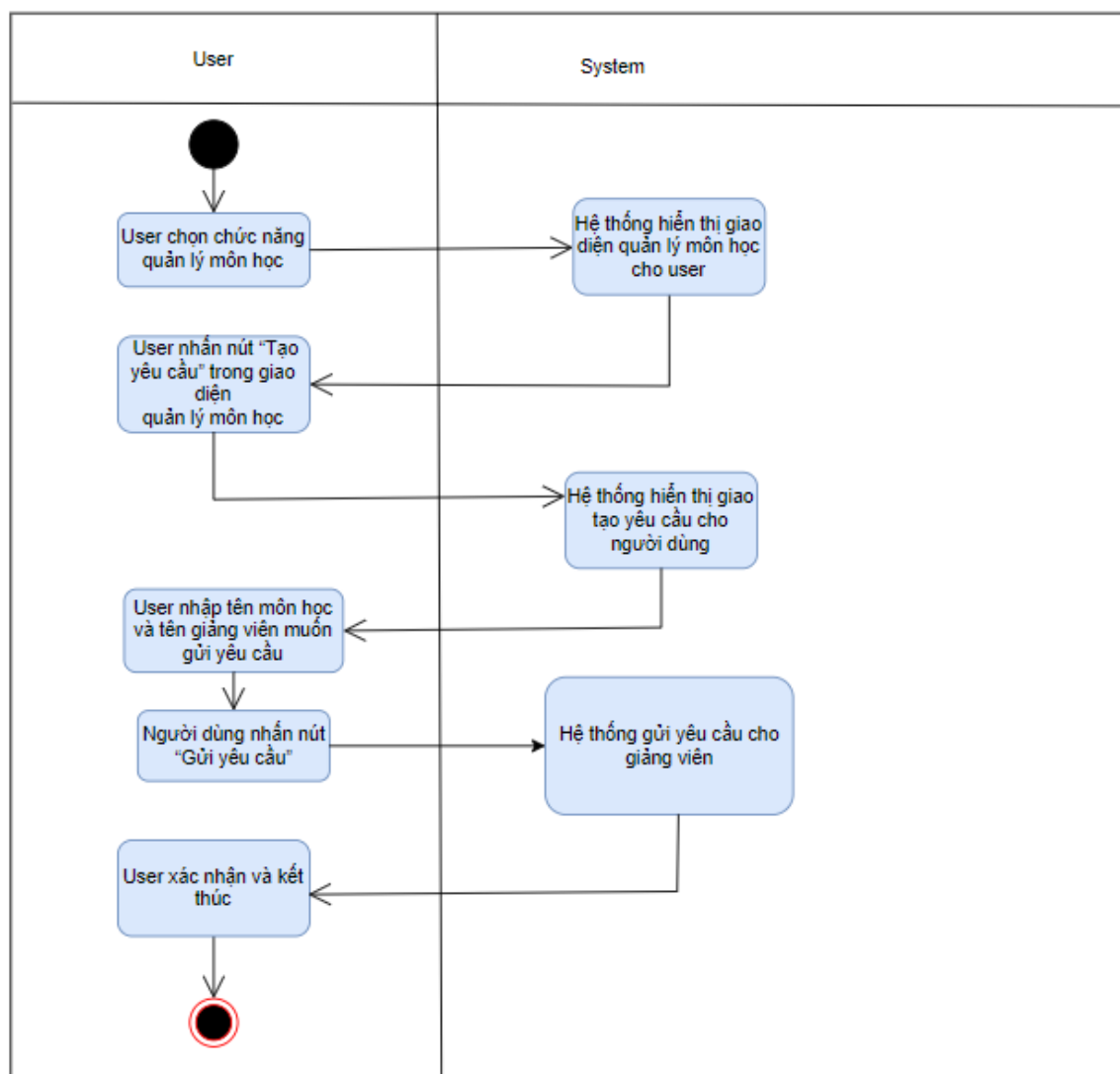
Hình 3. 19 Sơ đồ activity biểu diễn use case reset mật khẩu

3.5.2.15 Sơ đồ activity use case xóa giảng viên



Hình 3. 20 Sơ đồ activity biểu diễn use case xóa giảng viên

3.5.2.16 Sơ đồ activity use case tạo yêu cầu giảng viên



Hình 3. 21 Sơ đồ activity biểu diễn use case tạo yêu cầu giảng viên

3.6 Mô hình tổ chức dữ liệu

- curriculums(Id, Name)
- evalelements(Id, Order, Name, Method, Proportion, SubjectId)
- evaluates(Id, Order, Clo, Test, Method, Proportion, Target, SubjectId)
- semesters(Id ,Name)
- subjectcontents(Id, Order, Content, Nlessons, Clos, Method, Bonus, SubjectId)
- subjectcurriculums(Id, CurriculumId, SubjectId, SemesterId)
- subjectoutputstandards(Id, Clo, Content, SoPerPi, SubjectId)

- subjects(Id, Name, TheoryCredits, PracticeCredits, SelfLearningCredits, TotalCredits, Teachers, Documents, Goals, Abstract, A, B, C, Other, RequestUserMail)
- subjectusers(Id, UserId, SubjectId)
- users(Id, FullName, Faculty, DateOfBirth, Gender, PlaceOfBirth, UserName, NormalizedUserName, Email, NormalizedEmail, EmailConfirmed, PasswordHash, SecurityStamp, ConcurrencyStamp, PhoneNumber, PhoneNumberConfirmed, TwoFactorEnabled, LockoutEnd, LockoutEnabled, AccessFailedCount)

3.7 Mô tả chi tiết các bảng

Bảng 3. 18 Bảng curriculums

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|----------------------|--------------|-------------|
| Id | Id chương trình | Int | Primary key |
| Name | Tên chương trình học | Longtext | Not null |

Bảng 3. 19 Bảng evalelements

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|------------------------|--------------|-------------|
| Id | Id thành phần đánh giá | Int | Primary key |
| Order | Khác | Int | Not null |
| Name | Tên | Text | Not null |
| Method | Phương pháp đánh giá | Text | Not null |
| Proportion | Tỷ trọng | Int | Not null |
| SubjectId | Id môn học | Varchar(255) | Foreign key |

Bảng 3. 20 Bảng evaluates

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|--------------|--------------|-------------|
| Id | Id đánh giá | Int | Primary key |
| Order | Khác | Int | Not null |
| Clo | Clos | Int | Not null |
| Test | Bài kiểm tra | Longtext | Not null |
| Method | Phương pháp | Int | Not null |
| Proportion | Tỷ trọng | Int | Not null |
| Target | Chỉ tiêu | Int | Not null |
| SubjectId | Id môn học | Varchar(255) | Foreign key |

Bảng 3. 21 Bảng semesters

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|------------|--------------|-------------|
| Id | Id học kì | Int | Primary key |
| Name | Tên học kì | Longtext | Not null |

Bảng 3. 22 Bảng subjectcontents

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|---------------------------|--------------|-------------|
| Id | Id nội dung môn học | Int | Primary Key |
| Order | Khác | Int | Not null |
| Content | Nội dung các chương | Text | Not null |
| Nlessons | Số tiết học của chương đó | Int | Not null |
| Clos | Clos | Longtext | Not null |

| | | | |
|-----------|------------------------------|--------------|-------------|
| Method | Phương pháp giảng dạy | Longtext | Not null |
| Bonus | Nội dung và hướng dẫn tự học | Text | Null |
| SubjectId | Id môn học | Varchar(255) | Foreign key |

Bảng 3. 23 Bảng subjectcurriculums

| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|-----------------|--------------|-------------|
| Id | Id môn học | Int | Primary key |
| CurriculumId | Id chương trình | Int | Foreign key |
| SubjectId | Id môn học | Varchar(255) | Foreign key |
| SemesterId | Id học kì | Int | Foreign key |

Bảng 3. 24 Bảng subjectoutputstandards

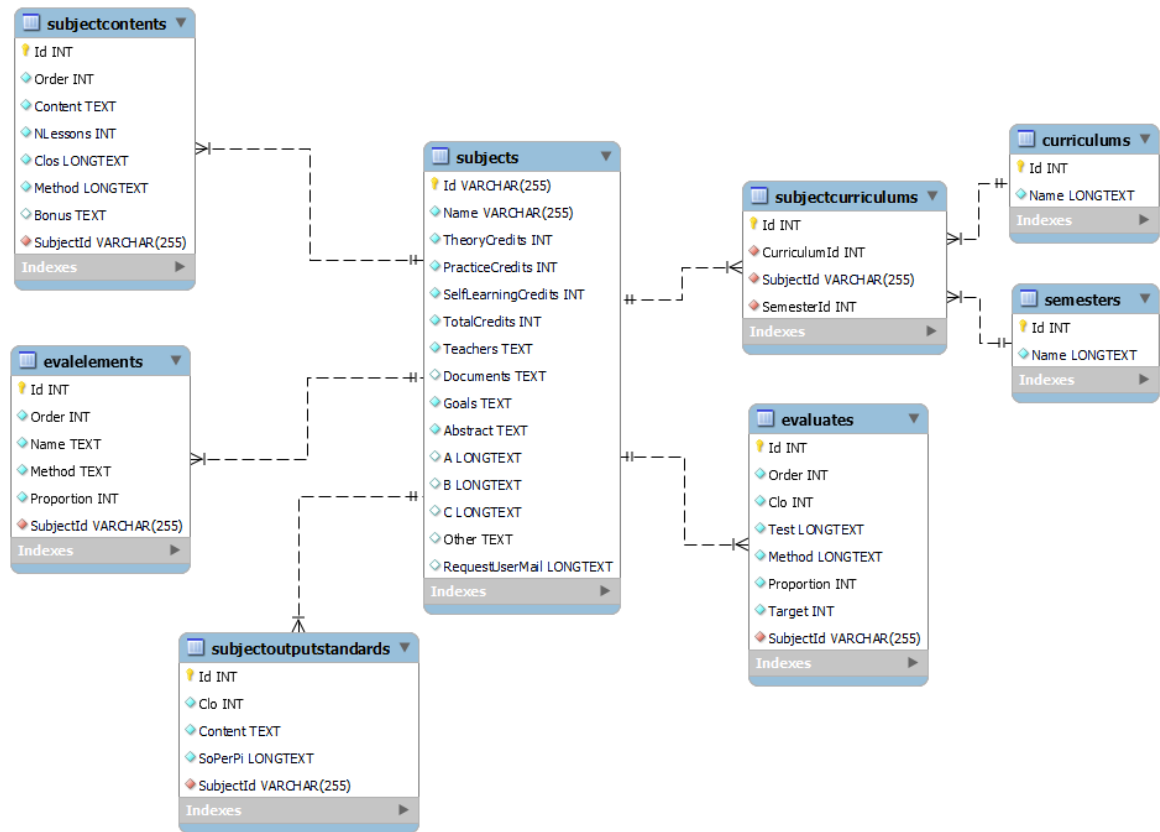
| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|-----------------------|--------------|-------------|
| Id | Id | Int | Primary key |
| Clo | CLos | Int | Not null |
| Content | Nội dung chuẩn đầu ra | Text | Not null |
| SoPerPi | SO/PI | Longtext | Not null |
| SubjectId | Id môn học | Varchar(255) | Foreign key |

Bảng 3. 25 Bảng subjects

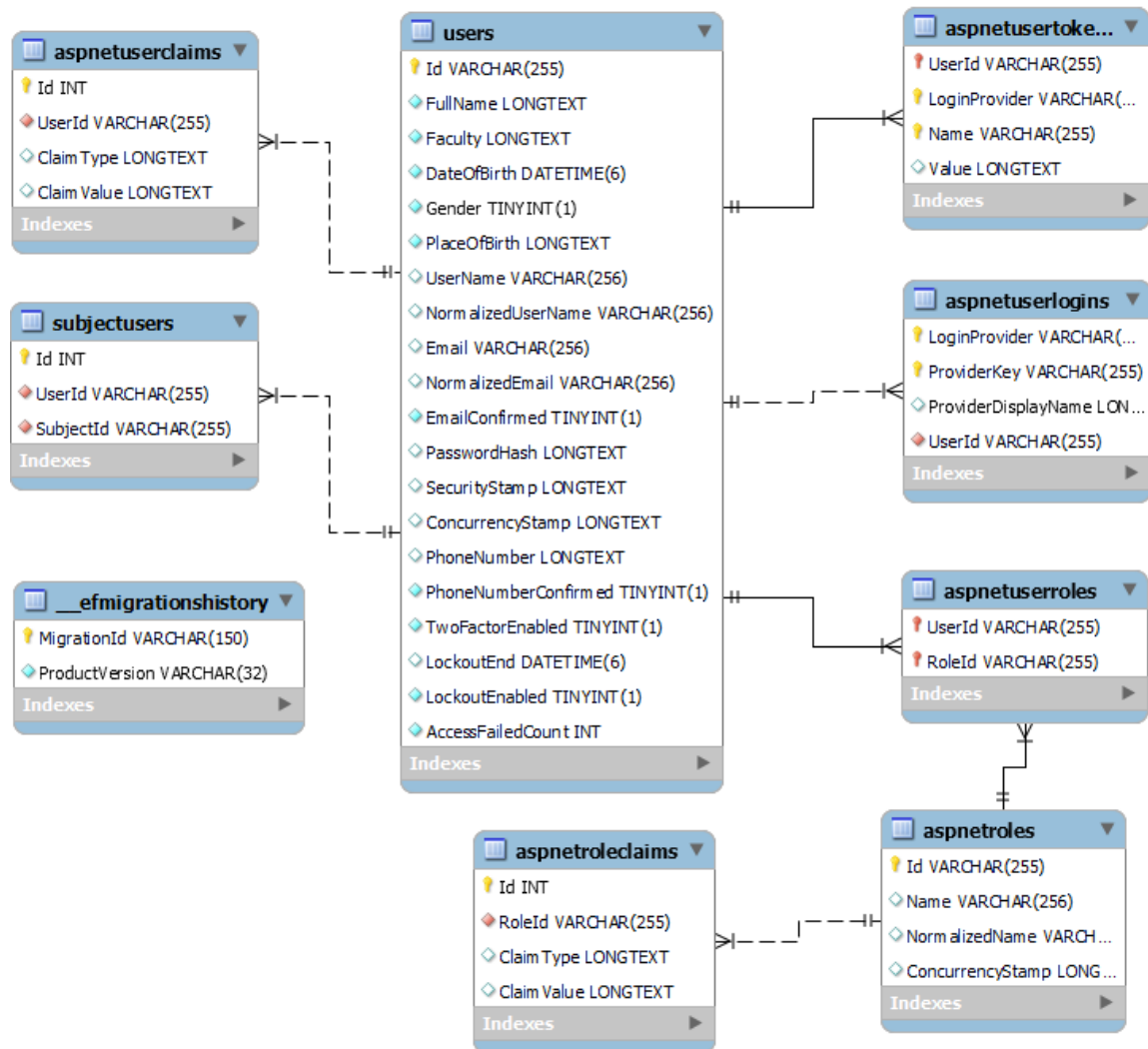
| Tên thuộc tính | Mô tả | Kiểu dữ liệu | Ràng buộc |
|----------------|----------------------|--------------|-------------|
| Id | Id môn học | Varchar(255) | Primary key |
| Name | Tên môn học | Varchar(255) | Not null |
| TheoryCredits | Số tín chỉ lý thuyết | Int | Not null |

| | | | |
|---------------------|------------------------|----------|----------|
| PracticeCredits | Số tín chỉ thực hành | Int | Not null |
| SelfLearningCredits | Số tín chỉ tự học | Int | Not null |
| TotalCredits | Tổng số tín chỉ | Int | Not null |
| Teachers | Giảng viên phụ trách | Text | Not null |
| Documents | Tài liệu giảng dạy | Text | Null |
| Goals | Mục tiêu môn học | Text | Not null |
| Abstract | Mô tả vắn tắt học phần | Text | Not null |
| A | Học phần học trước | Longtext | Null |
| B | Học phần tiên quyết | Longtext | Null |
| C | Học phần song hành | Longtext | Null |
| Other | Yêu cầu khác | Text | Null |
| RequestUserMail | Email của user | Longtext | Null |

3.8 Lược đồ quan hệ



Hình 3. 22 Lược đồ quan hệ môn học



Hình 3. 23 Lược đồ quan hệ hệ thống

CHƯƠNG 4: THIẾT KẾ VÀ HIỆN THỰC ỨNG DỤNG

4.1 Chuyển đổi từ file PDF, Docx sang database

4.1.1 Trích xuất tài liệu từ file Docx

Một trong những thách thức trong đồ án của nhóm là trích xuất các bảng dữ liệu từ các tệp tin đề cương ngành học. Để giải quyết vấn đề này, nhóm đã chọn sử dụng thư viện python-docx trong Python. Thư viện này cung cấp các công cụ và chức năng cho việc làm việc với tệp tin DOCX, cho phép nhóm truy cập và tương tác với nội dung của các tài liệu Word.

Bằng cách sử dụng python-docx, nhóm đã xây dựng một chương trình có khả năng tự động trích xuất các bảng từ các tệp tin đề cương ngành học. Các bảng được trích xuất được lưu trữ dưới dạng file csv sau đó được tổ chức quản lý bằng cách lưu trữ trong MySQL .

Việc sử dụng thư viện python-docx giúp nhóm giảm bớt công sức và thời gian trong việc trích xuất thông tin từ các tệp tin đề cương ngành học. Thư viện cung cấp các phương pháp và thuật toán tiện ích để tìm kiếm và truy cập vào các bảng dữ liệu trong tài liệu DOCX, đồng thời giúp tôi xử lý định dạng và cấu trúc của các bảng một cách linh hoạt.

Việc áp dụng python-docx để trích xuất các bảng cũng như thông tin từ các file đề cương môn học đem lại hiệu quả đáng kinh ngạc. Dưới đây là đoạn mã thực thi và kết quả trích xuất

Ở đây là file gốc đề cương học phần nhập môn tin học (ảnh dưới), trong trường hợp này nhóm muốn lấy bảng nội dung các chương môn học

ĐỀ CƯƠNG HỌC PHẦN

1. Tên và mã học phần: Nhập môn Tin học (2101539)

2. Số tín chỉ (2,0,4)

Tổng số tín chỉ: 2

Lý thuyết: 2

Thực hành: 0

Tự học: 4

3. Giảng viên phụ trách

ThS. Nguyễn Ngọc Dung

ThS. Trần Thị Kim Chi

TS. Lê Nhật Duy

TS. Ngô Hữu Dũng

4. Tài liệu học tập

Sách, giáo trình chính (thường chiếm từ 50% nội dung sử dụng trở lên)

[1] Computer Fundamentals (Sixth Edition) Pradeep K. Sinha, Priti Sinha

Tài liệu tham khảo (liệt kê tối đa 3 tài liệu tham khảo)

[1] Computer Literacy BASICS A Comprehensive Guide to IC3, 5th Edition [2]

Computer Concepts_2011 Jamrich Parsons and Dan Oja (2010)

5. Thông tin về học phần

a. Mục tiêu học phần

- Hiểu các khái niệm cơ bản về hệ thống máy tính, các thành phần cơ bản trong hệ thống máy tính và các hoạt động cơ bản trong hệ thống máy tính
- Phân biệt được các loại phần mềm
- Phân biệt được các hệ thống số, thực hiện được các phép tính trên hệ nhị phân, biết cách biểu diễn dữ liệu trên máy tính, phân biệt được các loại mã máy.
- Có khả năng tìm hiểu và sử dụng được những phần mềm ứng dụng thông dụng
- Tìm hiểu các môi trường làm việc và ứng dụng công nghệ thông tin - Nhận thức về các mối đe dọa đến an toàn dữ liệu và thông tin trên máy tính

b. Mô tả vắn tắt học phần

Môn học giới thiệu các khái niệm cơ bản về hệ thống máy tính, các loại phần mềm, các phép toán cơ bản trên các hệ cơ số, hệ nhị phân, các ứng dụng công nghệ thông tin, môi trường làm việc và các mối đe dọa đến an toàn thông tin trên máy tính.

c. Học phần học trước (A), tiên quyết (B), song hành (C)

Không

Hình 4. 1 Mẫu Nhập môn tin học 1

d. Yêu cầu khác Không

6. Chuẩn đầu ra của học phần

a. Chuẩn đầu ra của môn học.

Khi hoàn thành môn học, người học có khả năng:

| CLOs | Chuẩn đầu ra của học phần | SO/PI |
|------|---|-------|
| 1 | Trình bày cấu trúc và hoạt động cơ bản của hệ thống máy tính | |
| 2 | Thực hiện được các phép toán cơ bản trên hệ nhị phân và cách mã hóa dữ liệu cơ bản | A |
| 3 | Trình bày được các hoạt động cơ bản trong quy trình phát triển, hiện thực và bảo trì một hệ thống | |
| 4 | Trình bày được các khái niệm, cách tổ chức lưu trữ và xử lý dữ liệu trên máy tính | J |
| 5 | Trình bày được các mối đe dọa cơ bản đến hệ thống thông tin và cách phòng chống | |
| 6 | Trình bày được các môi trường làm việc và phương tiện truyền thông trong kỹ nguyên số | |

b. Ma trận tích hợp giữa chuẩn đầu ra của môn học và chuẩn đầu ra của chương trình đào tạo.

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | | | | |
| 2 | X | | | | | | | | | | |
| 3 | | | | | | | | | | | |
| 4 | | | | | | | | | | X | |
| 5 | | | | | | | | | | | |
| 6 | | | | | | | | | | | |

7. Nội dung cơ bản của môn học

| STT | Nội dung giảng dạy | Số tiết | CLOs | Phương pháp giảng dạy | Nội dung và hướng dẫn tự học |
|-----|--------------------|---------|------|-----------------------|------------------------------|
|-----|--------------------|---------|------|-----------------------|------------------------------|

Hình 4. 2 Mẫu Nhập môn tin học 2

| | | | | | |
|---|---|---|---|---------------|---|
| 1 | Chương 1. Tổng quan về hệ thống máy tính 1.1. Lịch sử phát triển của hệ thống máy tính 1.2. Khái niệm về hệ thống máy tính 1.2.1. Máy tính là gì? 1.2.2. Tầm quan trọng của máy tính 1.2.3. Hoạt động của một hệ thống máy tính 1.3. Thành phần cơ bản trong một máy tính 1.3.1. Thành phần của một hệ thống máy tính 1.3.1.1. Phần cứng 1.3.1.2. Phần mềm | 3 | 1 | L, D, I, P, H | Tìm hiểu tổng quan về máy tính và các hệ điều hành hiện nay |
| | 1.3.2. Hệ điều hành 1.3.3. Phân loại hệ điều hành 1.3.3.1. Hệ điều hành máy đơn 1.3.3.2. Hệ điều hành Mobile 1.3.3.3. Hệ điều hành nhúng 1.3.3.4. Hệ điều hành Server 1.3.4. Các hệ điều hành máy đơn thông dụng 1.3.4.1. Hệ điều hành windows 1.3.4.2. Hệ điều hành linux 1.3.4.3. Hệ điều hành Mac | | | | |

Hình 4. 3 Mẫu Nhập môn tin học 3

| | | | | | |
|---|--|---|---|---------------|--|
| 2 | Chương 2. Các phép toán liên quan đến máy tính – mã máy 2.1. Hệ đếm 2.1.1. Hệ đếm không theo vị trí 2.1.2. Hệ đếm theo vị trí 2.2. Cách chuyển đổi giữa các hệ thống số 2.3. Các phép toán trên số nhị phân 2.3.1. Phép cộng 2.3.2. Phép trừ, trừ bù 2.3.3. Phép nhân 2.3.4. Phép chia 2.4. Cách biểu diễn dữ liệu trên thiết bị số 2.4.1. Các loại mã máy 2.4.2. Cách biểu diễn dữ liệu Text, Number, Image, Sound trên thiết bị số | 6 | 2 | L, D, I, P, H | Tìm hiểu các loại phép toán liên quan đến máy tính Tìm hiểu loại mã biểu diễn ký tự |
| 3 | Chương 3. Ngôn ngữ máy tính 3.1. Phân loại ngôn ngữ máy tính 3.1.1. Ngôn ngữ máy 3.1.2. Hợp ngữ 3.1.3. Ngôn ngữ cấp cao 3.2. Ngôn ngữ máy – cấu trúc lệnh trong ngôn ngữ máy 3.2.1. Khái niệm ngôn ngữ máy 3.2.2. Ưu và nhược điểm của ngôn ngữ máy 3.3. Hợp ngữ - cấu trúc lệnh trong hợp ngữ 3.3.1. Khái niệm hợp ngữ 3.3.2. Cấu trúc lệnh trong hợp ngữ | 4 | | L, D, I, P, H | Tìm hiểu về các loại ngôn ngữ máy tính |

Hình 4. 4 Mẫu Nhập môn tin học 4

| | | | | | |
|---|---|---|---|---------------|--|
| | 3.3.3. Ưu và nhược điểm của hợp ngữ 3.4. Ngôn ngữ cấp cao 3.4.1. Khái niệm ngôn ngữ cấp cao 3.4.2. Một số ngôn ngữ cấp cao thông dụng | | | | |
| 4 | Chương 4. Hiện thực và vận hành hệ thống 4.1. Các hoạt động trong hiện thực hệ thống 4.1.1. Khái niệm kiểm thử và gỡ lỗi chương trình 4.1.2. Khái niệm về tài liệu phần mềm 4.2. Cách chuyển đổi thành hệ thống mới 4.2.1.1. Chuyển đổi trực tiếp 4.2.1.2. Chạy song song 4.2.1.3. Chuyển đổi từng giai đoạn 4.3. Đánh giá hệ thống 4.3.1. Khái niệm về quá trình đánh giá hệ thống 4.3.2. Những nội dung dùng để đánh giá hệ thống 4.4. Bảo trì hệ thống 4.4.1. Lý do bảo trì hệ thống 4.4.2. Quá trình bảo trì hệ thống | 4 | 3 | L, D, I, P, H | Tìm hiểu về quy trình vận hành và hiện thực hệ thống |

Hình 4. 5 Mẫu nhập môn tin học 5

| | | | | | |
|---|---|---|---|---------------|--|
| 5 | Chương 5. Khái niệm cơ bản về lưu trữ và xử lý dữ liệu 5.1. Các khái niệm cơ bản 5.1.1. Dữ liệu 5.1.2. Thông tin 5.1.3. Khái niệm về xử lý dữ liệu 5.2. Sự phân cấp trong lưu trữ dữ liệu 5.2.1. Các khái niệm trong lưu trữ dữ liệu 5.3. Các phương pháp tổ chức và lưu trữ dữ liệu 5.3.1. Hệ thống quản lý tập tin 5.3.2. Hệ quản trị cơ sở dữ liệu 5.3.3. Các mô hình dữ liệu | 4 | 4 | L, D, I, P, H | Tìm hiểu tổng quan về cơ sở dữ liệu |
| 6 | Chương 6. Cài đặt và sử dụng các phần mềm ứng dụng | 3 | | L, D, I, P, H | Tìm hiểu về các |
| | 1.1. Phân loại phần mềm 1.1.1. Phần mềm hệ thống 1.1.2. Phần mềm ứng dụng 1.2. Sơ lược về bản quyền phần mềm máy tính 1.2.1. Khái niệm phần mềm mã nguồn mở 1.2.2. Khái niệm về quyền của người dùng 1.3. Cách cài đặt và sử dụng phần mềm ứng dụng 1.3.1. Phần mềm văn phòng 1.3.2. Phần mềm đồ họa 1.3.3. Phần mềm công cụ | | | | phần mềm ứng dụng, cách khai thác và sử dụng |

Hình 4. 6 Mẫu Nhập môn tin học 6

| | | | | | |
|---|--|---|---|---------------|---|
| 7 | <p>Chương 7. Internet – đa phương tiện – ảnh hưởng của các giải pháp máy tính đến các cá nhân và tổ chức</p> <p>1.1. Internet</p> <p>1.1.1. Các khái niệm cơ bản</p> <p>1.1.1.1. IP address</p> <p>1.1.1.2. Domain name</p> <p>1.1.1.3. Tiện ích của Internet</p> <p>1.1.1.4. Kết nối Internet</p> <p>1.1.2. Khái niệm trình duyệt Internet</p> <p>1.1.2.1. Khái niệm trình duyệt</p> <p>1.1.2.2. Các trình duyệt thông dụng</p> <p>1.1.3. Các dịch vụ trên Internet</p> <p>1.1.3.1. Electronic –Mail</p> <p>1.1.3.2. File transfer protocol</p> <p>1.1.3.3. Telnet</p> <p>1.1.3.4. World Wide Web – WWW</p> <p>1.2. Multimedia</p> <p>1.2.1. Khái niệm đa phương tiện</p> <p>1.2.2. Các phương tiện truyền thông phổ biến</p> <p>1.2.3. ứng dụng của truyền thông đa phương tiện</p> <p>1.3. Ảnh hưởng các giải pháp máy tính đến các cá nhân và tổ chức</p> | 3 | 6 | L, D, I, P, H | Tìm hiểu về Internet và các ứng dụng đa phương tiện |
| | <p>1.3.1. Mô tả một số tác động chính của các giải pháp máy tính ảnh hưởng đến các cá nhân và tổ chức</p> <p>1.3.2. Thảo luận những tác động tích cực, tiêu cực liên quan đến những giải pháp công nghệ</p> <p>1.3.3. Những ảnh hưởng của giải pháp công nghệ đến chất lượng cuộc sống và mối quan hệ giữa các cá nhân</p> | | | | |

Hình 4. 7 Mẫu Nhập môn tin học 7

| | | | | | |
|---|---|---|---|---------------|--|
| 8 | Chương 8. Các mối đe dọa đến hệ thống thông tin 8.1. Vai trò của an toàn hệ thống thông tin 8.1.1. Khái niệm an toàn thông tin 8.1.2. Sự cần thiết của an toàn thông tin 8.1.3. Mục đích của an toàn thông tin 8.2. Các mối đe dọa đến hệ thống thông tin 8.2.1. Phần mềm gián điệp 8.2.2. Thư rác 8.2.3. Nguy cơ Virus máy tính 8.2.4. Phần mềm độc hại 8.2.5. Hacker và kẻ săn mồi 8.3. Cách phòng chống 8.3.1. Bảo vệ thông tin về mặt vật lý 8.3.2. Ngăn chặn nguy cơ mất thông tin 8.3.3. Ngăn chặn nguy cơ bị tấn công bởi các phần mềm độc hại 8.3.4. Ngăn chặn tấn công lỗ hổng bảo mật 8.3.5. kỹ thuật an toàn và bảo mật thông tin | 3 | 5 | L, D, I, P, H | Tìm hiểu các mối đe dọa ảnh hưởng đến an toàn thông tin của cá nhân và tổ chức |
|---|---|---|---|---------------|--|

L: Lecture **S:** Seminar **D:** Discussion **I:** Instructions in serving as model **Si:** Simulation **O:** Observation **P:** Practices **H:** Instruction for Homework **WA:** Work Assignment

Hình 4. 8 Mẫu Nhập môn tin học 8

Bên trên là nguyên mẫu của đề cương môn học, và có thể thấy bảng nội dung các chương không liền mạch. Nhóm đã dùng chương trình của mình tự làm ra để tiến hành trích xuất thông tin của bảng nội dung môn học.

Code

```

import docx
import pandas as pd
import re
    
```



```
def extract_content_table(docx_path:str)-> pd.DataFrame:
    data = []
    id = 0
    document = docx.Document(docx_path)
    len_table = len(document.tables)
    index_table = list(range(0,len_table))
    index_table = index_table[2:-2]
    for tb in index_table:
        table = document.tables[tb]
        if len(table.columns) == 6:
            for row in table.rows:
                row_data = []
                for cell in row.cells:
                    row_data.append(cell.text)
                data.append(row_data)
        elif len(table.columns) == 7:
            data_temp = []
            for row in table.rows:
                row_data = []
                for cell in row.cells:
                    row_data.append(cell.text)
                data_temp.append(row_data)
            df_temp = pd.DataFrame(data_temp)
            df_temp['D'] = df_temp[df_temp.columns[3]] +
df_temp[df_temp.columns[4]]
            df_temp = df_temp.drop([3,4], axis=1)
            df_temp =
df_temp[[df_temp.columns[0],df_temp.columns[1],df_temp.columns[2],'D',df_tem
p.columns[3],df_temp.columns[4]]]
```

```
df_temp = df_temp.rename(columns={"D":3})
for i , row in df_temp.iterrows():
    data.append(row)
df = pd.DataFrame(data)
for j in document.paragraphs[:4]:
    string = j.text
    string = string.split('\n')
    string = ".join(string)
    numbers = re.findall(r'\d+', string)
    if numbers == []:
        pass
    elif len(numbers) == 1 :
        numbers = ".join(numbers)
        id = numbers
    elif len(numbers) >1:
        numbers = numbers[-1]
        numbers = ".join(numbers)
        id = numbers

df.to_csv('C:\\Users\\Engineer\\Downloads\\Subjects\\Content\\content_tables_{}.csv'.format(id))

return df

if __name__ == "__main__":
    docx_path="C:\\Users\\Engineer\\Downloads\\Subjects\\subject\\result_0.docx"
    extract_content_table(docx_path)
```

Kết quả trích xuất:

Bảng 4. 1 Kết quả trích xuất nội dung giảng dạy môn Nhập môn tin học

| STT | Nội dung giảng dạy | Số tiết | CLOs | Phương pháp giảng dạy | Nội dung và hướng dẫn tự học |
|-----|--|---------|------|-----------------------|---|
| 1 | <p>Chương 1. Tổng quan về hệ thống máy tính</p> <p>1.1. Lịch sử phát triển của hệ thống máy tính</p> <p>1.2. Khái niệm về hệ thống máy tính</p> <p>1.2.1. Máy tính là gì?</p> <p>1.2.2. Tầm quan trọng của máy tính</p> <p>1.2.3. Hoạt động của một hệ thống máy tính</p> <p>1.3. Thành phần cơ bản trong một máy tính</p> <p>1.3.1. Thành phần của một hệ thống máy tính</p> <p>1.3.1.1. Phần cứng</p> <p>1.3.1.2. Phần mềm</p> | 3 | 1 | L, D, I, P, H | Tìm hiểu tổng quan về máy tính và các hệ điều hành hiện nay |
| | <p>1.3.2. Hệ điều hành</p> <p>1.3.3. Phân loại hệ điều hành</p> <p>1.3.3.1. Hệ điều hành máy đơn</p> <p>1.3.3.2. Hệ điều hành Mobile</p> <p>1.3.3.3. Hệ điều hành nhúng</p> <p>1.3.3.4. Hệ điều hành Server</p> <p>1.3.4. Các hệ điều hành máy đơn thông dụng</p> | | | | |

| | | | | | |
|---|--|---|---|---------------|---|
| | <p>1.3.4.1. Hệ điều hành windows</p> <p>1.3.4.2. Hệ điều hành linux</p> <p>1.3.4.3. Hệ điều hành Mac</p> | | | | |
| 2 | <p>Chương 2. Các phép toán liên quan đến máy tính – mã máy</p> <p>2.1. Hệ đếm</p> <p>2.1.1. Hệ đếm không theo vị trí</p> <p>2.1.2. Hệ đếm theo vị trí</p> <p>2.2. Cách chuyển đổi giữa các hệ thống số</p> <p>2.3. Các phép toán trên số nhị phân</p> <p>2.3.1. Phép cộng 2.3.2. Phép trừ, trừ bù</p> <p>2.3.3. Phép nhân</p> <p>2.3.4. Phép chia</p> <p>2.4. Cách biểu diễn dữ liệu trên thiết bị số</p> <p>2.4.1. Các loại mã máy</p> <p>2.4.2. Cách biểu diễn dữ liệu Text, Number, Image, Sound trên thiết bị số</p> | 6 | 2 | L, D, I, P, H | <p>Tìm hiểu các loại phép toán liên quan đến máy tính</p> <p>Tìm hiểu loại mã biểu diễn ký tự</p> |

| | | | | | |
|---|--|---|--|---------------|---|
| 3 | <p>Chương 3. Ngôn ngữ máy tính</p> <p>3.1. Phân loại ngôn ngữ máy tính</p> <p>3.1.1. Ngôn ngữ máy</p> <p>3.1.2. Hợp ngữ</p> <p>3.1.3. Ngôn ngữ cấp cao</p> <p>3.2. Ngôn ngữ máy – cấu trúc lệnh trong ngôn ngữ máy 3.2.1. Khái niệm ngôn ngữ máy</p> <p>3.2.2. Ưu và nhược điểm của ngôn ngữ máy</p> <p>3.3. Hợp ngữ - cấu trúc lệnh trong hợp ngữ</p> <p>3.3.1. Khái niệm hợp ngữ</p> <p>3.3.2. Cấu trúc lệnh trong hợp ngữ</p> | 4 | | L, D, I, P, H | <p>Tìm hiểu về các loại ngôn ngữ máy tính</p> |
| | <p>3.3.3. Ưu và nhược điểm của hợp ngữ</p> <p>3.4. Ngôn ngữ cấp cao</p> <p>3.4.1. Khái niệm ngôn ngữ cấp cao</p> <p>3.4.2. Một số ngôn ngữ cấp cao thông dụng</p> | | | | |

| | | | | | |
|---|---|---|---|---------------|--|
| 4 | <p>Chương 4. Hiện thực và vận hành hệ thống</p> <p>4.1. Các hoạt động trong hiện thực hệ thống</p> <p>4.1.1. Khái niệm kiểm thử và gỡ lỗi chương trình</p> <p>4.1.2. Khái niệm về tài liệu phần mềm</p> <p>4.2. Cách chuyển đổi thành hệ thống mới</p> <p>4.2.1.1. Chuyển đổi trực tiếp</p> <p>4.2.1.2. Chạy song song</p> <p>4.2.1.3. Chuyển đổi từng giai đoạn</p> <p>4.3. Đánh giá hệ thống</p> <p>4.3.1. Khái niệm về quá trình đánh giá hệ thống</p> <p>4.3.2. Những nội dung dùng để đánh giá hệ thống</p> <p>4.4. Bảo trì hệ thống</p> <p>4.4.1. Lý do bảo trì hệ thống</p> <p>4.4.2. Quá trình bảo trì hệ thống</p> | 4 | 3 | L, D, I, P, H | Tìm hiểu về quy trình vận hành và hiện thực hệ thống |
|---|---|---|---|---------------|--|

| | | | | | |
|---|---|---|---|---------------|--|
| 5 | <p>Chương 5. Khái niệm cơ bản về lưu trữ và xử lý dữ liệu</p> <p>5.1. Các khái niệm cơ bản</p> <p>5.1.1. Dữ liệu</p> <p>5.1.2. Thông tin</p> <p>5.1.3. Khái niệm về xử lý dữ liệu</p> <p>5.2. Sự phân cấp trong lưu trữ dữ liệu</p> <p>5.2.1. Các khái niệm trong lưu trữ dữ liệu</p> <p>5.3. Các phương pháp tổ chức và lưu trữ dữ liệu</p> <p>5.3.1. Hệ thống quản lý tập tin</p> <p>5.3.2. Hệ quản trị cơ sở dữ liệu</p> <p>5.3.3. Các mô hình dữ liệu</p> | 4 | 4 | L, D, I, P, H | Tìm hiểu tổng quan về cơ sở dữ liệu |
| 6 | <p>Chương 6. Cài đặt và sử dụng các phần mềm ứng dụng</p> | 3 | | L, D, I, P, H | Tìm hiểu về các |
| | <p>1.1. Phân loại phần mềm</p> <p>1.1.1. Phần mềm hệ thống</p> <p>1.1.2. Phần mềm ứng dụng</p> <p>1.2. Sơ lược về bản quyền phần mềm máy tính</p> <p>1.2.1. Khái niệm phần mềm mã nguồn mã nguồn mở phần mềm thương mại</p> <p>1.2.2. Khái niệm về quyền của người dùng</p> <p>1.3. Cách cài đặt và sử dụng phần mềm ứng dụng</p> | | | | phần mềm ứng dụng, cách khai thác và sử dụng |

| | | | | | |
|---|---|---|---|---------------|--|
| | <p>1.3.1. Phần mềm văn phòng</p> <p>1.3.2. Phần mềm đồ họa</p> <p>1.3.3. Phần mềm công cụ</p> | | | | |
| 7 | <p>Chương 7. Internet – đa phương tiện – ảnh hưởng của các giải pháp máy tính đến các cá nhân và tổ chức</p> <p>1.1. Internet</p> <p>1.1.1. Các khái niệm cơ bản</p> <p>1.1.1.1. IP address</p> <p>1.1.1.2. Domain name</p> <p>1.1.1.3. Tiện ích của Internet</p> <p>1.1.1.4. Kết nối Internet</p> <p>1.1.2. Khái niệm trình duyệt Internet</p> <p>1.1.2.1. Khái niệm trình duyệt</p> <p>1.1.2.2. Các trình duyệt thông dụng</p> <p>1.1.3. Các dịch vụ trên Internet</p> <p>1.1.3.1. Electronic –Mail</p> <p>1.1.3.2. File transfer protocol</p> <p>1.1.3.3. Telnet</p> <p>1.1.3.4. World Wide Web – WWW</p> <p>1.2. Multimedia</p> <p>1.2.1. Khái niệm đa phương</p> | 3 | 6 | L, D, I, P, H | <p>Tìm hiểu về Internet và các ứng dụng đa phương tiện</p> |

| | | | | | |
|--|--|--|--|--|--|
| | <p>tiện 1.2.2. Các phương tiện truyền thông phổ biến</p> <p>1.2.3. ứng dụng của truyền thông đa phương tiện</p> <p>1.3. Ảnh hưởng các giải pháp máy tính đến các cá nhân và tổ chức</p> | | | | |
| | <p>1.3.1. Mô tả một số tác động chính của các giải pháp máy tính ảnh hưởng đến các cá nhân và tổ chức</p> <p>1.3.2. Thảo luận những tác động tích cực, tiêu cực liên quan đến những giải pháp công nghệ</p> <p>1.3.3. Những ảnh hưởng của giải pháp công nghệ đến chất lượng cuộc sống và mối quan hệ giữa các cá nhân</p> | | | | |

| | | | | | |
|---|---|---|---|---------------|--|
| 8 | <p>Chương 8. Các mối đe dọa đến hệ thống thông tin</p> <p>8.1. Vai trò của an toàn hệ thống thông tin</p> <p>8.1.1. Khái niệm an toàn thông tin</p> <p>8.1.2. Sự cần thiết của an toàn thông tin 8.1.3. Mục đích của an toàn thông tin</p> <p>8.2. Các mối đe dọa đến hệ thống thông tin</p> <p>8.2.1. Phần mềm gián điệp</p> <p>8.2.2. Thư rác</p> <p>8.2.3. Nguy cơ Virus máy tính</p> <p>8.2.4. Phần mềm độc hại</p> <p>8.2.5. Hacker và kẻ săn mồi</p> <p>8.3. Cách phòng chống</p> <p>8.3.1. Bảo vệ thông tin về mặt vật lý</p> <p>8.3.2. Ngăn chặn nguy cơ mất thông tin</p> <p>8.3.3. Ngăn chặn nguy cơ bị tấn công bởi các phần mềm độc hại 8.3.4. Ngăn chặn tấn công lỗ hổng bảo mật</p> <p>8.3.5. kỹ thuật an toàn và bảo mật thông tin</p> | 3 | 5 | L, D, I, P, H | Tìm hiểu các mối đe dọa ảnh hưởng đến an toàn thông tin của cá nhân và tổ chức |
|---|---|---|---|---------------|--|

4.1.2 Trích xuất tài liệu từ file PDF

Ngoài việc sử dụng các file tài liệu docx, nhóm còn trích xuất dữ liệu từ file PDF với thư viện pypdf2. Dưới đây là kết quả trích xuất:

6. Chuẩn đầu ra của môn học

a. Chuẩn đầu ra của môn học.

| CLOs | Chuẩn đầu ra của học phần | SO/PI |
|------|---|-------|
| 1 | Sinh viên thiết kế được cơ sở dữ liệu mức logic. | |
| 2 | Sinh viên thiết kế được CSDL mức vật lý. | E1 |
| 3 | Sinh viên sử dụng được đại số quan hệ để tạo và truy xuất dữ liệu của một CSDL quan hệ. | A1 |
| 4 | Sinh viên sử dụng được ngôn ngữ SQL để tạo và truy xuất dữ liệu của một CSDL quan hệ. | K2 |
| 5 | Sinh viên xác định được dạng chuẩn của một cơ sở dữ liệu quan hệ. | |

Hình 4. 9 Mẫu cần trích xuất

Code

```
def transform_clos_table(path):
    import pandas as pd
    df = pd.DataFrame(path)
    list_column = list(df.columns)
    if 'CLOs' not in list_column :
        df = df.rename(columns={'Unnamed: 0.1' :str(df['Unnamed: 0.1'])[0]),
        'Unnamed: 1': str(df['Unnamed: 1'])[0]})
        df = df.loc[2: ]
        df = df.reset_index()
        return df
    else:
        return df
def set_clos_corrext(dataframe):
    clos_temp = 0
    for i in range(len(dataframe['Chuẩn đầu ra của học phần'])):
        try:
            if str(dataframe['Chuẩn đầu ra của học phần'][i]) != 'nan':
```

```
        if "Transaction logs backup) bằng lệnh hoặc công" in
str(dataframe['Chuẩn đầu ra của học phần'][i]):
            dataframe['CLOs'][i] = clos_temp
        elif str(dataframe['Chuẩn đầu ra của học phần'][i])[0].isupper() == True:
            clos_temp += 1
            dataframe['CLOs'][i] = clos_temp
        else:
            dataframe['CLOs'][i] = clos_temp
    else:
        if str(dataframe['Chuẩn đầu ra của học phần'][i])[0].isupper() == True:
            dataframe['CLOs'][i] = clos_temp
    except:
        pass
    return dataframe

def get_SO_PI(dataframe):
    list_key = []
    for i in range(len(dataframe)):
        text = str(dataframe['CLOs'][i]) + "-" + str(dataframe['SO/PI'][i])
        if text not in list_key:
            list_key.append(text)
        else:
            pass
    return list_key

def out_put_SOPI(string_1):
    curent = 0
    list_temp = []
    for i in range(len(string_1)):
        curent += 1
```

```
try:
    split_string_current = string_1[curent].split('-')
    split_string_i = string_1[i].split('-')
    if split_string_current[0] == split_string_i[0]:
        if split_string_current[1] == 'nan':
            list_temp.append(string_1[i])
        else:
            list_temp.append(string_1[curent])
    else:
        pass

except:
    continue

return list_temp

def convert_form_standar(dataframe):
    import pandas as pd
    dataframe = dataframe.dropna(subset=['Chuẩn đầu ra của học phần'])
    df_temp = dataframe.groupby('CLOs').agg(lambda x: x.tolist())
    df_temp = df_temp.reset_index()
    try:
        df_temp = df_temp.drop(['Unnamed: 0'], axis=1)
        df_temp = df_temp.drop(['index'], axis=1)
    except:
        print("Delete Faild")
    return df_temp

def convert_list_to_string(list):
    listToStr = '-'.join(map(str, list))
    return listToStr
```

```
def processing_SOPI_standar(listt):
    list_temp = list(set(listt))
    if len(list_temp) == 1:
        string = convert_list_to_string(list_temp)
        return string
    elif len(list_temp) >= 2:
        list_temp = [x for x in list_temp if str(x) != 'nan']
        string = convert_list_to_string(list_temp)
        return string

import PyPDF2
import tabula
from collections import OrderedDict
from collections import defaultdict
import pandas as pd
import numpy as np

def extract_table(pdf_path):
    dict = {'CLOs':[1], 'Chuẩn đầu ra của học phần':[np.nan],"SO/PI":[np.nan]}
    data = pd.DataFrame(dict)
    pdfFileObj = open(pdf_path, 'rb')
    pdfReader = PyPDF2.PdfFileReader(pdfFileObj)
    """get number page in pdf file"""
    number_page = pdfReader.numPages
    """create dict to set page correct"""
    d = defaultdict()
    for page in range(number_page):
        pageObj = pdfReader.getPage(page)
        text = pageObj.extractText()
        split = text.split('\n')
```

```
list_temp = [i for i in split if "Trang " in i]
list_to_string = ''.join([str(elem) for elem in list_temp])
list_to_string = list_to_string.strip()
number_real_page = list_to_string.split()[1]
d[number_real_page] = page
dict1 = OrderedDict(sorted(d.items()))
"extract CLOS table"
for i in dict1.values():
    page = i + 1
    dfs = tabula.read_pdf(pdf_path, pages=page, lattice=True)
    if len(dfs) == 0:
        pass
    else:
        for table in dfs:
            table_temp = pd.DataFrame(table)
            list_column = table_temp.columns
            if 'Chuẩn đầu ra của học phần' in list_column:
                # if 'Chuẩn đầu ra của học phần' in list_column:
                dataframe_clos = transform_clos_table(table)
                datafame_1 = set_clos_corrext(dataframe_clos)
                list_key_1 = get_SO_PI(datafame_1)
                out_1 = out_put_SOPI(list_key_1)
                df_temp_1 = convert_form_standar(datafame_1)
                df_temp_1['Chuẩn đầu ra của học phần'] = df_temp_1['Chuẩn đầu ra
của học phần'].apply(
                    lambda x: convert_list_to_string(x))
                df_temp_1['SO/PI'] = df_temp_1['SO/PI'].apply(lambda x:
processing_SOPI_standar(x))
```

```

print("Transfrom success!")

data = pd.concat([data, df_temp_1]).reset_index(drop=True)

elif (len(list_column) == 3 or "Phương pháp đánh giá" in list_column or
      "Tỉ trọng %" in list_column) and 'Chuẩn đầu ra của học phần' not in list_column:

    if table_temp.columns[2] == 'Unnamed: 0':

        try:

            new_row = pd.DataFrame({'CLOs': table_temp.columns[0],
            'Chuẩn đầu ra của học phần': table_temp.columns[1], 'SO/PI': np.nan}, index=[0])

            df = table_temp.rename(columns={table_temp.columns[0]:
            "CLOs", table_temp.columns[1]: 'Chuẩn đầu ra của học phần',

            table_temp.columns[2]: "SO/PI"})

            print('rename success')

            df = pd.concat([new_row, df]).reset_index(drop=True)

            data = pd.concat([data, df]).reset_index(drop=True)

        except:

            print('rename fail')

    else:

        new_row = pd.DataFrame(

            {'CLOs': table_temp.columns[0], 'Chuẩn đầu ra của học phần':
            table_temp.columns[1], 'SO/PI': table_temp.columns[2]},

            index=[0])

        df = table_temp.rename(

            columns={table.columns[0]: "CLOs", table_temp.columns[1]:
            'Chuẩn đầu ra của học phần',

            table_temp.columns[2]: "SO/PI"})

        print('rename success')

        df = pd.concat([new_row, df]).reset_index(drop=True)

        data = pd.concat([data, df]).reset_index(drop=True)

    else:

```



```
        pass
    index = 0
    for id in range(len(data.CLOs)):
        try:
            int(data.CLOs[id])
        except:
            print(id)
            index = id
            break
    data = data.loc[1:index-1]
    data = data[['CLOs', 'Chuẩn đầu ra của học phần', 'SO/PI']]
    try:
        df2 = data.drop_duplicates(keep='first')
        df2 = df2[['CLOs', 'Chuẩn đầu ra của học phần', 'SO/PI']]
        print("df2")
        return df2
    except:
        print("data")
        return data
    path = 'C:\\Users\\Engineer\\CurrTex\\subject\\result_1.pdf'
    dataframe = extract_table(path)
    dataframe.to_csv('result.csv', index=False, encoding='utf-8-sig')
    print(dataframe)
```

Kết quả

Bảng 4. 2 Kết quả trích xuất chuẩn đầu ra môn học

| CLOs | Chuẩn đầu ra của học phần | SO/PI |
|------|---|-------|
| 1 | Sinh viên thiết kế được cơ sở dữ liệu mức logic. | Nan |
| 2 | Sinh viên thiết kế được CSDL mức vật lý. | E1 |
| 3 | Sinh viên sử dụng được đại số quan hệ để tạo và truy xuất dữ liệu của một CSDL quan hệ. | A1 |
| 4 | Sinh viên sử dụng được ngôn ngữ SQL để tạo và truy xuất dữ liệu của một CSDL quan hệ. | K2 |
| 5 | Sinh viên xác định được dạng chuẩn của một cơ sở dữ liệu quan hệ. | nan |

4.2 Tổng quan về ứng dụng


Trong đồ án chuyên ngành này, hệ thống quản lý đề cương ngành học đã ứng dụng công nghệ thông tin trong việc:

- Hỗ trợ giảng viên soạn thảo đề cương môn học dễ dàng hơn, dựa trên những môn học đã được lưu trữ lại
- Hỗ trợ trưởng bộ môn xây dựng đề cương ngành học dễ dàng hơn.
- Hỗ trợ trưởng bộ môn quản lý chương trình học, nội dung đề cương các môn học
- Hỗ trợ trưởng bộ môn quản lý giảng viên phụ trách các môn học.
- Thống nhất được khuôn mẫu cho các đề cương môn học, ngành học.

4.3 Một số chức năng của ứng dụng

4.3.1 Đăng nhập

WELLCOME



Your email

Password

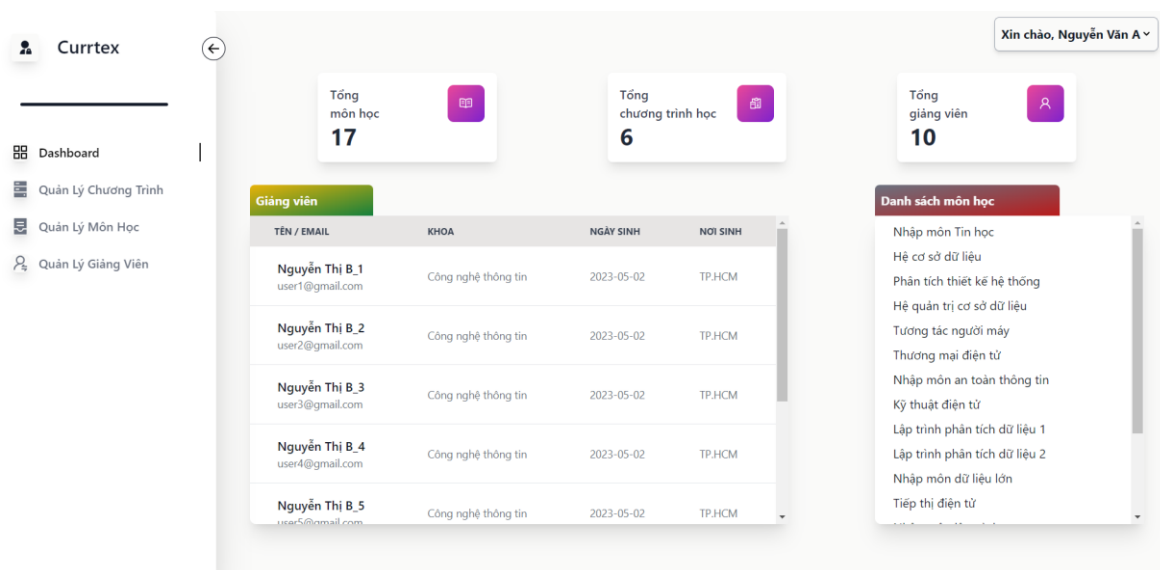
Sign in

Hình 4. 10 Giao diện đăng nhập của hệ thống

Đây là giao diện đầu tiên khi người dùng truy cập vào hệ thống.

Người dùng có thể đăng nhập bằng email và password. Trong trường hợp người dùng không có tài khoản, cần yêu cầu cấp tài khoản từ admin để đăng nhập. Trong trường hợp người dùng là admin, nếu quên tài khoản và mật khẩu cần liên hệ với hệ thống để được cấp lại. Nếu người dùng là giảng viên, trong trường hợp quên tài khoản hoặc mật khẩu, cần liên hệ với admin để được cấp hoặc reset lại mật khẩu.

4.3.2 Giao diện Dashboard của trưởng bộ môn



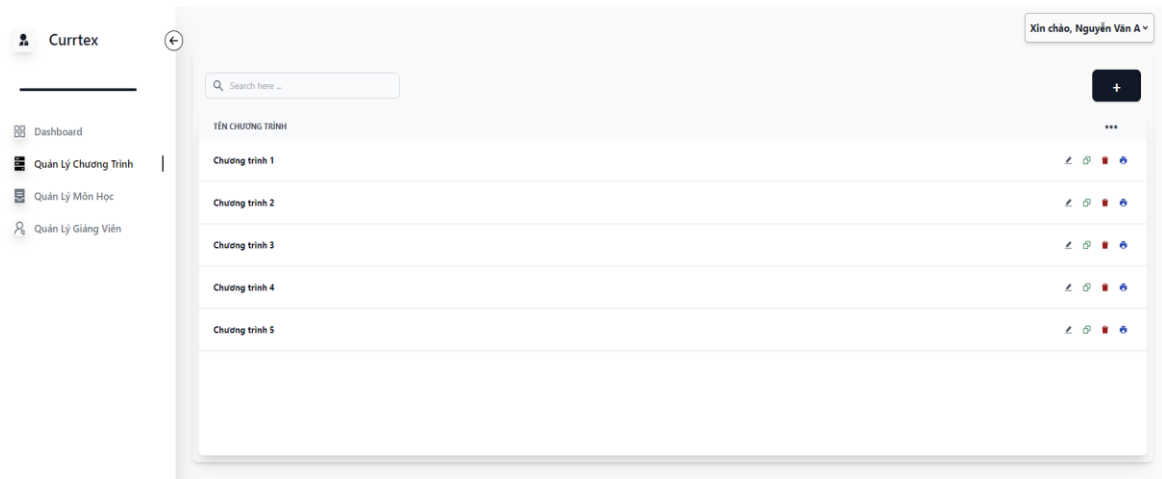
Hình 4. 11 Giao diện Dashboard khi trưởng bộ môn đăng nhập

Giao diện này sẽ hiển thị các thông tin như:

- Tổng số môn học hiện đang có trong hệ thống.
- Tổng số chương trình học hiện có.
- Tổng số giảng viên hiện đang có tài khoản trong hệ thống.
- Danh sách các môn học hiện có.
- Thông tin giảng viên cùng khoa. Ở đây, trưởng bộ môn đang thuộc khoa công nghệ thông tin nên chỉ có thể xem thông tin các giảng viên thuộc khoa công nghệ thông tin.

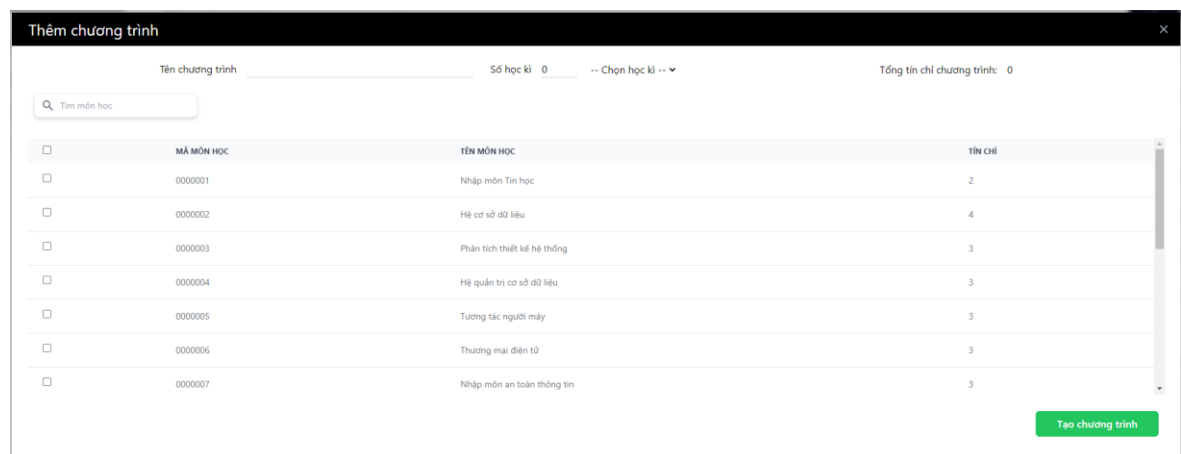
4.3.3 Giao diện quản lý chương trình

- Giao diện quản lý chương trình



Hình 4. 12: Giao diện quản lý chương trình

- Giao diện thêm chương trình



Hình 4. 13 Giao diện thêm chương trình

Ở giao diện này, trưởng bộ môn sẽ nhập tên chương trình, nhập số lượng học kỳ. Sau khi nhập số lượng học kỳ, bên dropdown sẽ hiển thị các lựa chọn với số lượng học kỳ tương ứng như hình bên dưới.

Số học kì 08

-- Chọn học kì --

- Học kì 1
- Học kì 2
- Học kì 3
- Học kì 4
- Học kì 5
- Học kì 6
- Học kì 7
- Học kì 8

TÊN MÔN HỌC

Nhập môn Tin học

Hệ cơ sở dữ liệu

Hình 4. 14 Dropdown chọn học kì tương ứng với số lượng học kì đã nhập

Sau khi chọn học kì ở dropdown, tiếp theo tiến hành học những môn học cho học kì đó bằng cách tích vào các ô checkbox trước mỗi môn học.

- Giao diện sao chép chương trình

Sao chép chương trình

Tên chương trình: Số học kì: 8 -- Chọn học kì -- Tổng tín chỉ chương trình: 90

Q Tìm môn học

| <input type="checkbox"/> | MÃ MÔN HỌC | TÊN MÔN HỌC | TÍN CHỈ |
|--------------------------|------------|-----------------------------|---------|
| <input type="checkbox"/> | 0000001 | Nhập môn Tin học | 2 |
| <input type="checkbox"/> | 0000002 | Hệ cơ sở dữ liệu | 4 |
| <input type="checkbox"/> | 0000003 | Phân tích thiết kế hệ thống | 3 |
| <input type="checkbox"/> | 0000004 | Hệ quản trị cơ sở dữ liệu | 3 |
| <input type="checkbox"/> | 0000005 | Tương tác người máy | 3 |
| <input type="checkbox"/> | 0000006 | Thương mại điện tử | 3 |
| <input type="checkbox"/> | 0000007 | Nhập môn an toàn thông tin | 3 |

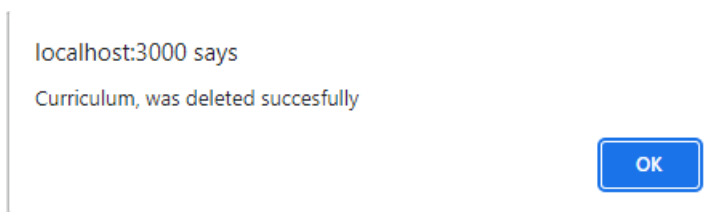
Sao chép

Hình 4. 15 Giao diện sao chép chương trình

Giao diện này có các chức năng tương tự tạo chương trình. Mục đích là để tạo chương trình mới một cách nhanh chóng dựa trên các chương trình học đã có trong hệ thống. Khi nhấn vào nút sao chép chương trình, giao diện này sẽ hiện lên nội dung chương

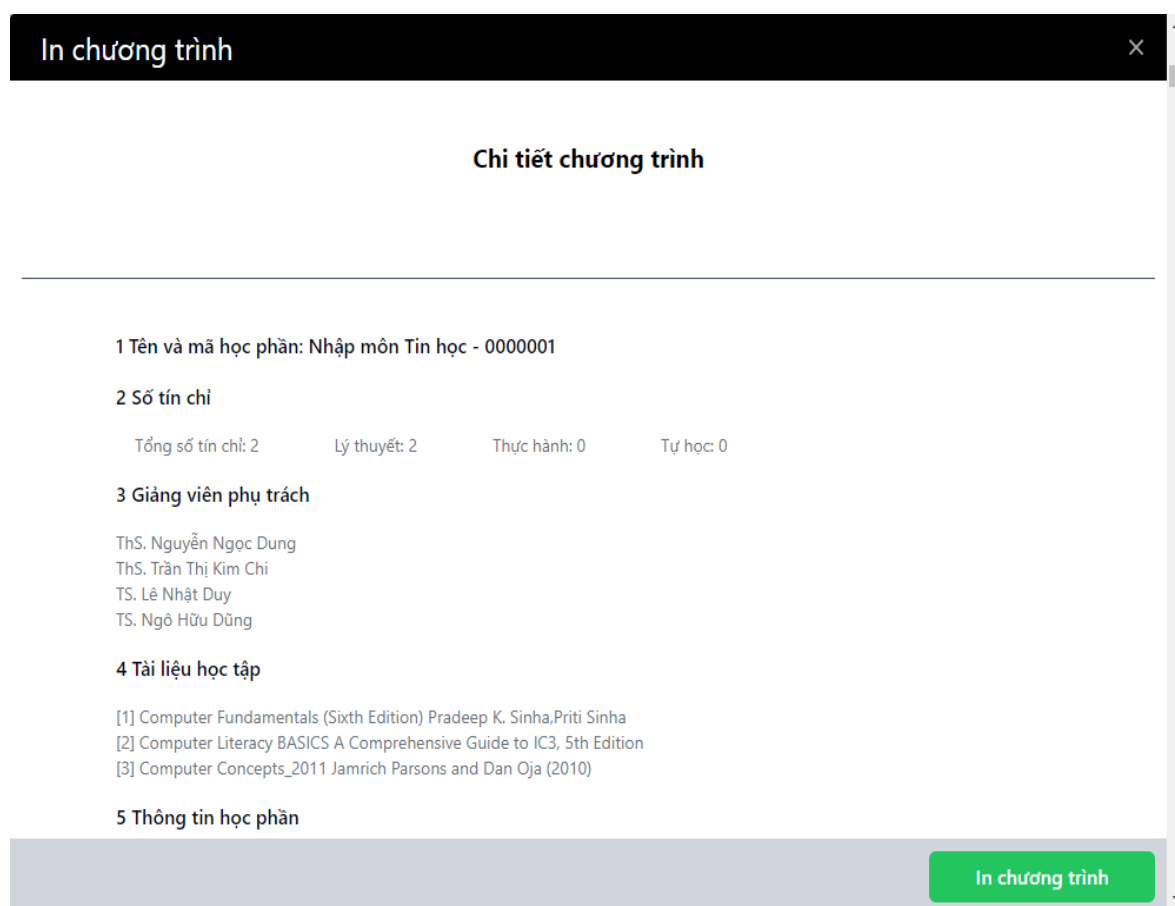
trình vừa sao chép. Sau khi chỉnh sửa chương trình và bấm vào nút “Sao chép”, một chương trình mới sẽ được tạo.

- Giao diện xóa chương trình



Hình 4. 16 Thông báo khi xóa chương trình thành công

- Giao diện in chương trình

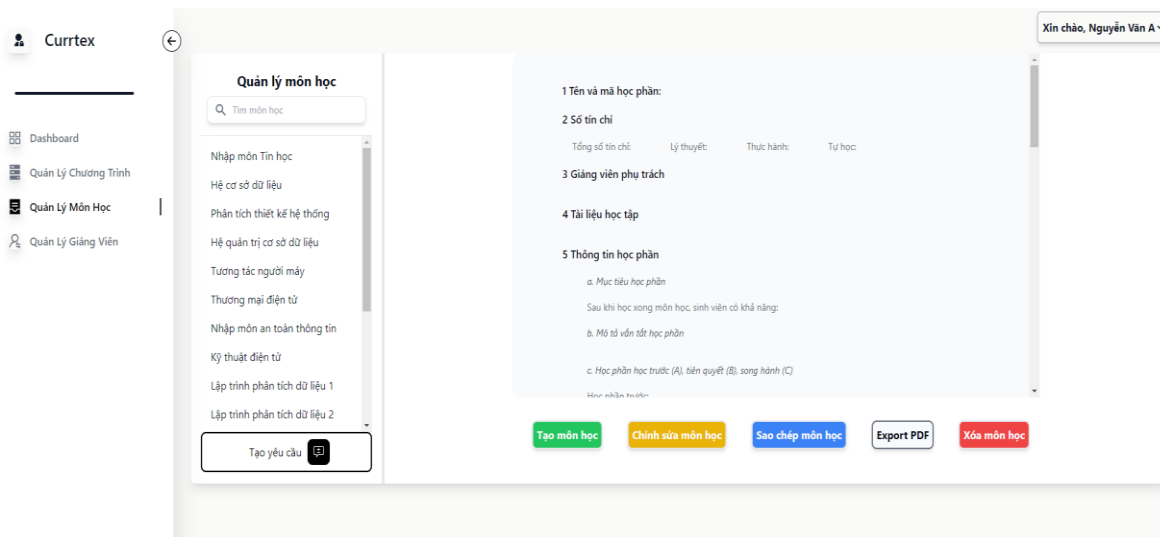


Hình 4. 17 Giao diện in chương trình học

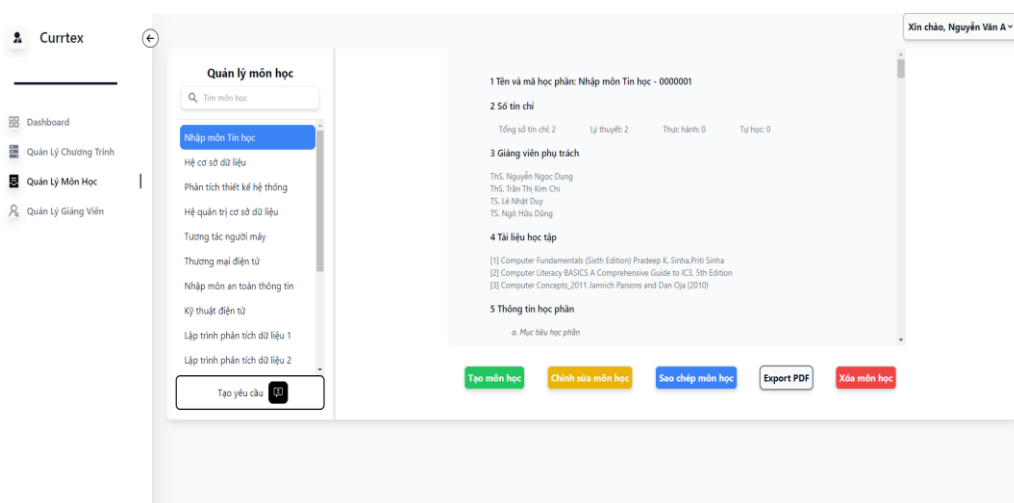
Giao diện in chương trình cho phép người dùng có thể nhìn lại chương trình một cách tổng quát như trên file dạng pdf. Sau khi nhấn vào nút “In chương

trình”, hệ thống sẽ cho phép người dùng tải về file pdf với nội dung là chương trình vừa chọn.

4.3.4 Giao diện quản lý môn học



Hình 4. 18 Giao diện khi chưa chọn môn học để xem thông tin



Hình 4. 19 Giao diện khi chọn vào môn học để xem thông tin chi tiết

Khi chưa chọn môn học, thông tin các trường môn học được để trống. Sau khi chọn môn học, thông tin môn đó sẽ được đổ vào các trường dữ liệu cho người dùng.

- Giao diện tạo môn học

Thêm môn học

1

Normal

÷

B

I

U

↺

≡

≡

0

Normal

÷

B

I

U

↺

≡

≡

+

Phương pháp đánh giá

a. Phương pháp đánh giá các chuẩn đầu ra của học phần

| CLOs | Bài kiểm tra | Phương pháp đánh giá | Tỉ trọng (%) | Chỉ tiêu (%) |
|------|--------------|----------------------|--------------|--------------|
| 0 | | | 0 | 0 |

+

b. Các thành phần đánh giá

| Phương pháp | Phương pháp đánh giá | Tỉ trọng (%) |
|-------------|---|--------------|
| | <div>Normal</div> <div>÷</div> <div>B</div> <div>I</div> <div>U</div> <div>↺</div> <div>≡</div> <div>≡</div> <div>↵</div> | |

+

c. Thang điểm đánh giá: Theo học chế tín chỉ

Lưu

Hình 4. 20 Giao diện thêm môn học

Ở giao diện này, người dùng có thể điền các thông tin môn học sau đó nhấn nút “Lưu”. Hệ thống sẽ lưu thông tin môn học mới trong hệ thống và thông báo đã tạo thành công môn học mới.

- Giao diện chỉnh sửa môn học

Cập nhật môn học

b. Các thành phần đánh giá

| Phương pháp | Phương pháp đánh giá | Tỉ trọng (%) |
|-------------|--|--------------|
| Lý thuyết | <div>Normal</div> <div>÷</div> <div>B</div> <div>I</div> <div>U</div> <div>↺</div> <div>≡</div> <div>≡</div> <div>↵</div> <div>Đánh giá thường xuyên (cách 1) (20)</div> <ul style="list-style-type: none"> Bài kiểm tra (5) Bài tập ở nhà (5) Project (10) <div>Kiểm tra giữa kỳ (30)</div> <div>Kiểm tra cuối kỳ (50)</div> | 100 |
| Thực hành | <div>Normal</div> <div>÷</div> <div>B</div> <div>I</div> <div>U</div> <div>↺</div> <div>≡</div> <div>≡</div> <div>↵</div> <div>Chuẩn bị bài (10)</div> <div>Kỹ năng thực hành (40)</div> <div>Báo cáo thực hành (10)</div> <div>Kiểm tra thực hành cuối khóa (40)</div> | 100 |

+

-

c. Thang điểm đánh giá: Theo học chế tín chỉ

Lưu

Hình 4. 21 Giao diện chỉnh sửa môn học

Cho phép người dùng chỉnh sửa thông tin môn học đã có trong hệ thống, sau đó nhấn nút “Lưu” để cập nhật thông tin môn học.

- Giao diện sao chép môn học

Sao chép môn học

| STT | Tên môn học | Loại hình | Thời gian | Điểm | Trạng thái |
|-----|-----------------------|-----------------------|-----------|------|------------|
| 1 | Kiểm tra thường kỳ | Trắc nghiệm / tự luận | 40 | 40 | + |
| 2 | Thi giữa kỳ (Tự luận) | Tự luận | 60 | 60 | - |
| 3 | Kiểm tra thường kỳ | Trắc nghiệm / tự luận | 40 | 40 | - |
| 4 | Thi cuối kỳ (Tự luận) | Tự luận | 60 | 60 | - |

b. Các thành phần đánh giá

| Phương pháp | Phương pháp đánh giá | Tỉ trọng (%) |
|-------------|---|--------------|
| Lý thuyết | Normal B I U ☞ ☞ ☞ Đánh giá thường xuyên (20) • Bài kiểm tra thường xuyên (5) • Bài tập ở nhà (5) • Báo cáo trên lớp (5) • Hoạt động khác (5) Kiểm tra giữa kỳ (30) Kiểm tra cuối kỳ (50) | 100 + |

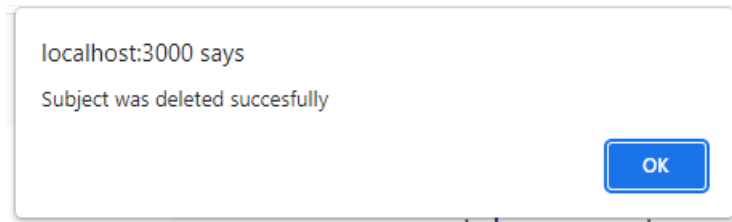
c. Thang điểm đánh giá: Theo học chế tín chỉ

Lưu

Hình 4. 22 Giao diện sao chép môn học

Cho phép người dùng dựa vào môn học có sẵn để tạo môn học mới, tránh mất nhiều thời gian nếu như môn học mới có nhiều nội dung tương đồng môn học đã có trong hệ thống.

- In môn học
 Khi nhấn vào biểu tượng in, hệ thống sẽ xuất 1 file dạng pdf cho người dùng tải về máy.
- Giao diện xóa môn học



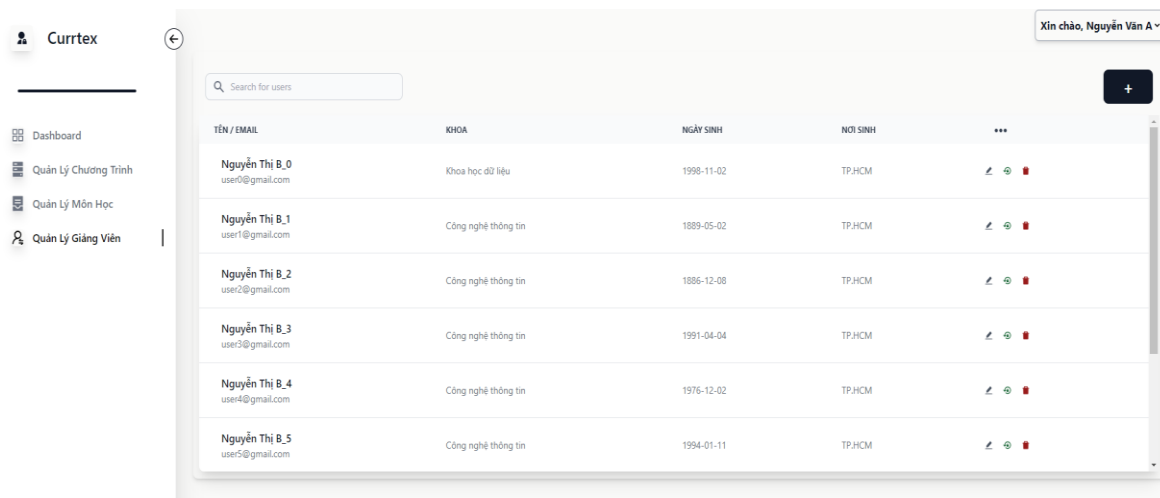
Hình 4. 23 Thông báo đã xóa môn học thành công

- Giao diện Tạo yêu cầu giảng viên

A screenshot of a web application form titled "Gửi yêu cầu" (Send request). The form has a black header bar with the title and a close button (X). Below the header, there are two input fields: "Tên học phần" (Course name) and "Chọn giảng viên được phân công" (Select assigned lecturer). To the right of the second field is a search bar with the placeholder text "Tìm giảng viên" (Find lecturer) and a magnifying glass icon. At the bottom right of the form is a green button labeled "Gửi yêu cầu" (Send request).

Hình 4. 24 Giao diện tạo yêu cầu cho giảng viên

4.3.5 Giao diện quản lý giảng viên



Hình 4. 25 Giao diện quản lý giảng viên

Ở giao diện này, admin có thể sử dụng các chức năng như:

- Thêm giảng viên mới
- Chỉnh sửa thông tin giảng viên
- Reset mật khẩu cho tài khoản giảng viên
- Xóa giảng viên khỏi hệ thống
 - Giao diện thêm giảng viên mới

Thêm giảng viên

Tên

Tên

Giới tính

☒ Nam ☐ Nữ

Email

example@gmail.com

Số điện thoại

Số điện thoại

Ngày sinh

mm/dd/yyyy

Nơi sinh

Nơi sinh

Khoa

-- Chọn khoa --

Thêm

Hình 4. 26 Giao diện thêm giảng viên

- Giao diện chỉnh sửa thông tin giảng viên

Cập nhật thông tin giảng viên

Tên

Nguyễn Thị B_0

Giới tính

☐ Nam

☒ Nữ

Email

user0@gmail.com

Số điện thoại

0900000000

Ngày sinh

11/02/1998

Nơi sinh

TP.HCM

Khoa

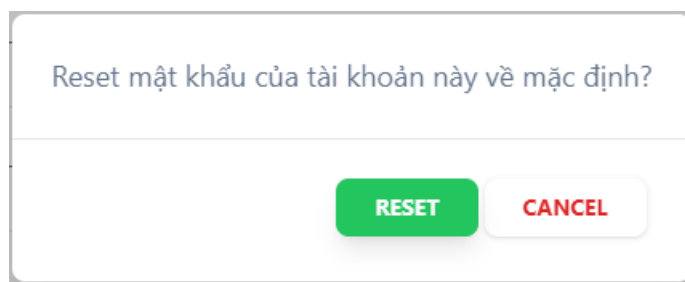
Khoa học dữ liệu

Cập nhật thông tin

Hình 4. 27 Giao diện chỉnh sửa thông tin giảng viên

Ở giao diện này, trưởng bộ môn có thể chỉnh sửa các thông tin như tên, giới tính, email, số điện thoại, ngày sinh, nơi sinh, khoa của giảng viên. Phần cập nhật thông tin ngày sinh, có thể nhập ngày sinh vào khung nhập hoặc nhấn vào biểu tượng cuốn lịch để chọn ngày sinh. Sau khi chỉnh sửa hoàn tất, admin nhấn nút “Cập nhật thông tin” để cập nhật lại thông tin giảng viên trong hệ thống.

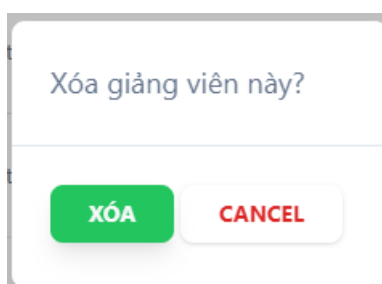
- Giao diện reset mật khẩu cho giảng viên



Hình 4. 28 Giao diện xác nhận khi reset mật khẩu giảng viên

Khi nhấn Reset, mật khẩu sẽ được trở về mặc định. Khi giảng viên quên mật khẩu và yêu cầu admin cấp lại mật khẩu. Admin sẽ sử dụng chức năng này. Nhấn cancel, giao diện reset mật khẩu sẽ đóng lại.

- Giao diện xóa giảng viên khỏi hệ thống



Hình 4. 29 Giao diện xác nhận khi xóa giảng viên khỏi hệ thống

Khi nhấn xóa, mọi thông tin của giảng viên được chọn sẽ được xóa khỏi hệ thống, màn hình sẽ trở về giao diện quản lý giảng viên ban đầu. Nếu nhấn vào nút Cancel, giao diện này cũng sẽ đóng lại mà không thực hiện bất kì hành động nào khác.

4.4 Ưu điểm

Hệ thống đảm bảo được các chức năng cơ bản mà yêu cầu đặt ra.

Việc sử dụng các API giúp cho hệ thống có thể chạy độc lập giữa front-end và back-end mà không phụ thuộc vào bất cứ điều gì.

Việc sử dụng Reactjs và Tailwindcss trong việc xây dựng front-end giúp cho người dùng có những trải nghiệm cũng như hiển thị một cách tốt hơn so với các framework có cùng chức năng.

Việc sử dụng ASP.NET cũng như EntityFramework giúp cho việc xây dựng back-end dễ dàng tích hợp các module của front-end.

4.5 Hạn chế

- Khuôn mẫu in đề cương ngành học còn đơn giản, chưa thực sự đầy đủ thông tin.
- Chức năng tương tác giữa trưởng bộ môn và giảng viên chưa thực sự hoàn thiện.

CHƯƠNG 5: KẾT LUẬN

5.1 Những điều đạt được

Về phần lý thuyết thì chúng em đã tìm hiểu được những vấn đề sau:

- Tìm hiểu tổng quan về Reactjs, Tailwindcss, ASP.NET, Web API
- Tìm hiểu được về Docker, MySQL
- Nắm được về phân tích thiết kế hệ thống
- Nắm được nghiệp vụ của hệ thống

Ngoài ra trong hệ thống, chúng em đã xây dựng được các chức năng sau:

- Tạo được một hệ thống quản lý về đề cương ngành học.
- Hỗ trợ cho trưởng bộ môn và giảng viên trong việc xây dựng, quản lý đề cương ngành/môn học.

5.2 Những vấn đề thuận lợi

Được tạo điều kiện để tìm hiểu cũng như làm việc và tiếp thu công nghệ mới. Tạo tiền đề để tiếp tục tìm hiểu chuyên sâu hơn những công nghệ mới giúp bản thân bắt kịp với tiến độ phát triển của công nghệ.

5.3 Hướng phát triển của đề tài

- Nâng cấp khuôn mẫu in đề cương theo chuẩn theo yêu cầu của khoa: Cụ thể thay vì dùng khuôn tạo từ html css thay bằng latex để có thể điều chỉnh, nâng cấp dễ dàng hơn.
- Hoàn thiện các chức năng tương tác giữa trưởng bộ môn và giảng viên: Cụ thể hoàn thiện chức năng tương tác có thông báo đã hoàn thành yêu cầu và bổ sung thêm việc tương tác qua email.
- Nâng cấp giao diện hệ thống dễ dàng sử dụng, trực quan hơn với người dùng.
- Tạo thêm tài khoản quyền admin phòng trường hợp hệ thống gặp sự cố, không để trưởng bộ môn vừa quyền admin vừa quyền trưởng bộ môn

- Chính sửa lại chức năng của trưởng bộ môn: Cụ thể loại bỏ quyền xóa giảng viên của trưởng bộ môn, thay vào đó là chức năng trạng thái (hoạt động hoặc không hoạt động).
- Nâng cấp lại hệ thống lưu trữ: Cụ thể tách tạo riêng bảng giảng viên và bổ sung thêm bảng giảng viên phụ trách môn học.

TÀI LIỆU THAM KHẢO

- [1] H. Nguyễn, "Quantrimang," 15 5 2022. [Online]. Available: <https://quantrimang.com/cong-nghe/ly-do-nen-su-dung-phan-mem-ao-hoa-docker-189844>.
- [2] T. N. Minh, "Viblo," 20 10 2020. [Online]. Available: <https://viblo.asia/p/components-va-props-trong-reactjs-Ljy5VmDGla>.
- [3] P. X. Nam, "Viblo," 17 7 2019. [Online]. Available: <https://viblo.asia/p/reactjs-component-lifecycle-methods-3Q75wdQ2KWb>.
- [4] N. Đ. A. Tuấn, "Viblo," 14 7 2018. [Online]. Available: <https://viblo.asia/p/hieu-sao-ve-virtual-dom-trong-reactjs-bWrZngDblxw>.
- [5] N. V. Thịnh, "Viblo," 9 2 2021. [Online]. Available: <https://viblo.asia/p/vi-sao-nhieu-du-an-su-dung-tailwind-css-gDVK2Bv0KLj>.
- [6] P. Đ. Anh, "Viblo," 20 06 2019. [Online]. Available: <https://viblo.asia/p/tailwind-css-lieu-co-thay-the-duoc-bootstrap-4P8564g9ZY3>.
- [7] V. V. Phong, "Viblo," 22 2 2021. [Online]. Available: <https://viblo.asia/p/utility-first-css-xay-dung-giao-dien-tu-nhung-vien-gach-nho-nhat-eW65GmeLZDO>.

Hệ thống quản lý đề cương ngành học

GVHD: ThS. Trương Vĩnh Linh

NHẬT KÝ LÀM VIỆC

| Tuần | Từ ngày | Đến ngày | Tóm tắt công việc đã thực hiện |
|------|------------|------------|--|
| 1 | 09/01/2023 | 15/01/2023 | Tìm hiểu, nghiên cứu các yêu cầu của hệ thống |
| 2 | 16/01/2023 | 22/01/2023 | Tìm hiểu về các file đề cương môn học/ngành học có sẵn |
| 3 | 23/01/2023 | 29/01/2023 | Từ các file đề cương môn học/ngành học có sẵn, sử dụng các thư viện Python để trích xuất dữ liệu |
| 4 | 30/01/2023 | 05/02/2023 | Tiếp tục trích xuất dữ liệu từ các file tài liệu có sẵn |
| 5 | 06/02/2023 | 12/02/2023 | Xây dựng database để lưu trữ dữ liệu được trích xuất từ file tài liệu |
| 6 | 13/02/2023 | 19/02/2023 | Thiết kế giao diện Figma |

| | | | |
|----|------------|------------|---|
| 7 | 20/02/2023 | 26/02/2023 | Xây dựng Back-end: Thiết kế các entity theo cơ sở dữ liệu. |
| 8 | 27/02/2023 | 05/03/2023 | Xây dựng chức năng Back-end: Đăng nhập, xác thực. |
| 9 | 06/03/2023 | 12/03/2023 | Xây dựng chức năng Back-end: Đăng nhập, xác thực. |
| 10 | 13/03/2023 | 19/03/2023 | Xây dựng chức năng Backend: Quản lý Môn học. |
| 11 | 20/03/2023 | 26/03/2023 | Xây dựng chức năng Backend: Quản lý Chương trình. |
| 12 | 27/03/2023 | 02/04/2023 | Xây dựng Front-end: Thiết lập dự án, cài đặt các thư viện cơ bản, cài đặt các router. |
| 13 | 03/04/2023 | 09/04/2023 | Xây dựng Front-end: Xây dựng trang Login, Dashboard, Profile |
| 14 | 10/04/2023 | 16/04/2023 | Xây dựng Front-end: Xây dựng trang Quản lý giảng viên, Quản lý môn học |

| | | | |
|----|------------|------------|--|
| 15 | 17/04/2023 | 23/04/2023 | Xây dựng Front-end: Xây dựng trang Quản lý chương trình |
| 16 | 24/04/2023 | 30/04/2023 | Xây dựng Front-end cho Giảng viên: Xây dựng trang Quản lý chương trình, Quản lý môn học và Dashboarh |
| 17 | 01/05/2023 | 07/05/2023 | Báo cáo trước GVHD, chuẩn bị file báo cáo |
| 18 | 08/05/2023 | 14/05/2023 | Hoàn thành file báo cáo |