

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



ĐỒ ÁN TỔNG HỢP - HUỐNG CÔNG NGHỆ PHẦN MỀM

Report for Project SMART PRINTING SYSTEM FOR SCHOOL

Advisor: Mai Đức Trung
Class: L06
Students: Nguyễn Minh Tú - 2213848
Đặng Văn Tân - 2213050
Trần Uy - 2213897
Hồ Minh Hưng - 2211361
Đỗ Quý - 2320010
Mai Hoàng Huỳnh - 2211314

HO CHI MINH CITY, JANUARY 2025

Mục lục

Danh mục viết tắt	2
Danh sách bảng	3
Danh sách hình vẽ	4
Tiến độ và phân công công việc	5
1 Xác định yêu cầu	6
1.1 Bối cảnh và các bên liên quan của dự án	6
1.1.1 Bối cảnh	6
1.1.2 Giới thiệu hệ thống	6
1.1.3 Xác định các bên liên quan và nhu cầu của họ	7
1.1.4 Lợi ích của hệ thống SPSS đối với mỗi bên liên quan	8
1.2 Yêu cầu chức năng và phi chức năng từ mô tả dự án	9
1.2.1 Yêu cầu chức năng	9
1.2.2 Yêu cầu phi chức năng	9
1.3 Những tính năng được hiện thực	12
2 Use-case Diagram	13
2.1 Use-case diagram cho toàn bộ hệ thống	13
2.2 Use-case diagram cho Module Xác Thực	14
2.3 Use-case diagram cho Module Quản lí in	16
2.3.1 Đăng ký in	16
2.3.2 Quản lí lịch sử in	18
2.4 Use-case diagram cho Module Quản lí máy in	22
2.5 Use-case diagram cho Module Quản lí hàng đợi	26
2.6 Use-case diagram cho Module Mua và Thanh Toán	29
3 Thiết kế giao diện	30
3.1 Liên kết	30
3.2 Mô tả	30
4 Lựa chọn công nghệ	36
4.1 Front-End	36
4.1.1 HTML (Hyper Text Markup Language)	36
4.1.2 CSS (Cascading Style Sheets)	36
4.1.3 JS (JavaScript)	37
4.2 Back-End	37
4.2.1 NodeJS	37
4.2.2 Express.js	38
4.2.3 PayOS	39
4.3 Cơ sở dữ liệu MySQL	39
4.4 Công cụ hỗ trợ	40
4.4.1 Docker	40
4.4.2 Postman	40
4.4.3 Figma	40
4.5 GitHub	41
5 Tổng kết	41
5.1 Link source của nhóm	41
5.2 Nhận xét	41
5.3 Hướng phát triển trong tương lai	42
6 Tài liệu tham khảo	42



Danh mục viết tắt

SPSS Student Smart Printing Service

SPSO Student Printing Service Officer

ID Identification

THPT Trung học phổ thông

NFR Non-Functional Requirements

QR Quick Response

JS JavaScript

HTML HyperText Markup Language

CSS Cascading Style Sheets

Node.js JavaScript runtime environment

Express.js Web application framework for Node.js

AI Artificial Intelligence

Danh sách bảng

1	Danh sách thành viên	5
2	Phần trăm đóng góp	5
3	Mô tả use case - Login	14
4	Mô tả use case - Forgot password	15
5	Mô tả use case - Log out	15
6	Mô tả use case - Tải file	16
7	Mô tả use case - Chọn máy in và cài đặt thông số in	17
8	Mô tả use case - In tài liệu	17
9	Mô tả use case - Xem lịch sử in cá nhân	18
10	Mô tả use case - Xem lịch sử in tổng	19
11	Mô tả use case - Xem lịch sử in của máy in	19
12	Mô tả use case - Xóa lịch sử in người dùng	20
13	Mô tả use case - Tìm kiếm lịch sử in	20
14	Mô tả use case - Xem tất cả lịch sử in của người dùng	21
15	Mô tả use case - Xem tất cả lịch sử in bất kì của người dùng	21
16	Mô tả use case - Xem máy in tổng	22
17	Mô tả use case - Tìm kiếm thông tin máy in	23
18	Mô tả use case - Xem thông tin chi tiết từng máy in	23
19	Mô tả use case - Xóa một máy in cụ thể	24
20	Mô tả use case - Thay đổi trạng thái máy in	24
21	Mô tả use case - Thêm máy in mới	25
22	Mô tả use case - Xem thông tin chi tiết từng máy in	25
23	Mô tả use case - Xem hàng đợi từng máy in	26
24	Mô tả use case - Quản lý hàng đợi	27
25	Mô tả use case - Xác nhận yêu cầu in	27
26	Mô tả use case - Xem lịch sử in của máy in	28
27	Mô tả use case - Xem hàng đợi tổng	28
28	Mô tả use case - Mua và thanh toán	29

Danh sách hình vẽ

2.1	Use-case diagram toàn bộ hệ thống	13
2.2	Use-case xác thực và phân quyền	14
2.3	Use-case đăng ký in	16
2.4	Use-case quản lí lịch sử in	18
2.5	Use-case Quản lí máy in	22
2.6	Use-case Quản lí hàng đợi	26
2.7	Use-case mua và thanh toán	29
3.8	Giao diện trang chủ	30
3.9	Giao diện chọn tư cách đăng nhập	31
3.10	Giao diện đăng nhập	31
3.11	Giao diện trang chủ	32
3.12	Giao diện tải tệp và điều chỉnh thông số in	32
3.13	Giao diện chọn máy in	33
3.14	Giao diện xem hàng đợi của máy in	33
3.15	Giao diện xem lịch sử in	34
3.16	Giao diện mua giấy in	34
3.17	Giao diện thanh toán	35
3.18	Thông báo thanh toán thành công	35

Tiến độ và phân công công việc

Bảng 1: *Danh sách thành viên*

Tính năng	Front-End	Back-End
Xác thực và phân quyền	Đặng Văn Tân, Trần Uy	Đặng Văn Tân
Đăng kí in	Đặng Văn Tân, Trần Uy	Đặng Văn Tân
Mua và Thanh toán	Trần Uy, Đỗ Quý	Đỗ Quý
Quản lí máy in và Trạng thái máy in	Nguyễn Minh Tú	Nguyễn Minh Tú
Quản lí lịch sử in	Nguyễn Minh Tú	Hồ Minh Hưng
Quản lí hàng đợi	Trần Uy, Nguyễn Minh Tú	Mai Hoàng Huỳnh
Thông báo	Trần Uy, Hồ Minh Hưng	Nguyễn Minh Tú, Hồ Minh Hưng
Tin nhắn	Mai Hoàng Huỳnh	Mai Hoàng Huỳnh

Bảng 2: *Phần trăm đóng góp*

Họ và Tên	Mã số sinh viên	Phần trăm đóng góp
Nguyễn Minh Tú	2213848	20%
Hồ Minh Hưng	2211361	14.5%
Trần Uy	2213897	19.0%
Mai Hoàng Huỳnh	2211314	12.5%
Đặng Văn Tân	2213050	20%
Đỗ Quý	2320010	14.0%

1 Xác định yêu cầu

1.1 Bối cảnh và các bên liên quan của dự án

1.1.1 Bối cảnh

Dưới sự phát triển của giáo dục, các trường học luôn không ngừng cải tiến quy trình giảng dạy, xây dựng các hệ thống hỗ trợ để góp phần nâng cao chất lượng dạy và học. Trong khi đó, việc học tập tại các trường học nói riêng và ngành giáo dục nói chung luôn đi kèm với nhu cầu in ấn các tài liệu liên quan phục vụ quá trình học tập, giảng dạy và nghiên cứu. Tuy nhiên, các hệ thống, địa điểm in ấn truyền thống thường không đáp ứng đủ các yêu cầu về tốc độ, quản lý tài nguyên, sự tiện lợi và tiện ích cho người dùng. Nhằm giải quyết vấn đề này, một hệ thống in ấn thông minh cần được xây dựng tại các trường học với mục tiêu tích hợp các công nghệ hiện đại để tự động hóa quy trình in ấn, hỗ trợ học sinh, sinh viên quản lý tài liệu, tối ưu hóa chi phí, và nâng cao hiệu quả sử dụng máy in trong khuôn viên nhà trường.

Trong môi trường giáo dục hiện đại, việc tối ưu hóa các dịch vụ hỗ trợ sinh viên là một yếu tố quan trọng để nâng cao trải nghiệm học tập. Hệ thống in ấn tại trường học là một trong những dịch vụ thiết yếu và cần thiết, đáp ứng nhu cầu in ấn các tài liệu liên quan của học sinh, sinh viên trong quá trình học tập. Tuy nhiên, việc quản lý và vận hành hệ thống in truyền thống thường gặp phải nhiều bất cập như thời gian chờ đợi quá lâu, khó khăn trong việc kiểm soát số lượng in và đôi khi còn thiếu minh bạch về chi phí. Để khắc phục những vấn đề này, một hệ thống in ấn thông minh cần được xây dựng tại các trường học với mục tiêu tích hợp các công nghệ hiện đại, tự động hóa quy trình in ấn, hỗ trợ học sinh, sinh viên quản lý tài liệu, tối ưu hóa chi phí và nâng cao hiệu quả sử dụng máy in trong khuôn viên nhà trường.

1.1.2 Giới thiệu hệ thống

Hệ thống này bao gồm nhiều máy in được phân bố đồng đều xung quanh khuôn viên trường, mỗi máy in sẽ có các thông tin chi tiết như kiểu máy, hãng sản xuất,... và đặc biệt là 1 ID duy nhất cho từng máy để cán bộ tại trường học thuận tiện quản lý.

Để in tài liệu, học sinh cần tải tài liệu lên hệ thống, chỉ định các thuộc tính in và sau đó chọn 1 máy in nhất định và các hành động in ấn của học sinh đều được lưu lại để có thể kiểm tra khi cần thiết.

Các cán bộ quản lý có thể kiểm tra lịch sử in theo vùng, theo ID máy in để biết rõ những khu vực có nhu cầu in ấn nhiều từ học sinh để có thể thay đổi sự phân bố máy in trong trường khi cần thiết, giảm thiểu trường hợp lãng phí tài nguyên thiết bị không cần thiết.

Mỗi tháng, trường học sẽ cung cấp 1 số lượng trang in nhất định dành cho học, đối với những học sinh có nhu cầu sử dụng nhiều thì có thể mua thêm với đơn giá được quy định bởi cán bộ nhà trường. Học sinh có thể dễ dàng thanh toán trực tiếp trên hệ thống thông qua việc chuyển khoản ngân hàng, ví điện tử,... Nhằm đáp ứng nhu cầu in ấn tài liệu của học sinh, sinh viên tại các cơ sở giáo dục. Chúng tôi đã triển khai một hệ thống in ấn Thông minh dành cho học sinh, sinh viên. Hệ thống này được thiết kế để mang lại sự tiện lợi, hiệu quả và minh bạch trong việc in ấn tài liệu, đồng thời giúp tối ưu hóa việc quản lý tài nguyên in ấn trong nhà trường.

Hệ thống này bao gồm một mạng lưới các máy in được đặt tại các khu vực khác nhau trong trường. Mỗi máy in đều được quản lý thông qua các thông tin chi tiết như mã máy in, tên hãng sản xuất, model và vị trí cụ thể (cơ sở, tòa nhà, số phòng).

Hệ thống cho phép sinh viên thực hiện các thao tác thông qua ứng dụng web để tải tài liệu cần in, cài đặt các thông số và chọn máy in phù hợp sau đó tiến hành in tài liệu. Ngoài ra, hệ thống còn có thể lưu lại lịch sử in ấn, giúp sinh viên tra cứu các tài liệu đã in khi cần thiết, tránh trường hợp một tài liệu được in quá nhiều gây lãng phí.

Mỗi học kỳ, nhà trường sẽ cấp một số lượng trang in mặc định cho mỗi sinh viên (khổ A4). Sinh viên có thể mua thêm trang in qua tính năng Mua thêm trang in tích hợp trong hệ thống và thanh toán trực tuyến qua mã QR được tích hợp trên hệ thống. Quy định sử dụng trang in được áp dụng nghiêm ngặt: một trang A3 tương đương với hai trang A4, và số trang in không được vượt quá số dư hiện tại trong tài khoản in của sinh viên.

Đối với cán bộ quản lý dịch vụ in ấn, hệ thống cung cấp các tính năng như Quản lý máy in (thêm,

xóa, kích hoạt hoặc vô hiệu hóa máy in); Quản lý cấu hình hệ thống: thay đổi số lượng trang in mặc định, ngày cấp trang in, và các định dạng tệp được phép in; xem và tải xuống báo cáo sử dụng hệ thống của sinh viên trong một khoảng thời gian nhất định đã được hệ thống tự động tổng hợp trước đó.

Hệ thống này không chỉ giải quyết những bất cập hiện tại trong việc in ấn tại các trường học mà còn góp phần nâng cao trải nghiệm học tập và sự chuyên nghiệp trong quản lý tài nguyên nhà trường.

1.1.3 Xác định các bên liên quan và nhu cầu của họ

Hệ thống SPSS có 3 bên liên quan chính, bao gồm: *Người dùng cuối: Giảng viên, Sinh viên và Học sinh, Nhân viên quản lý dịch vụ in cho người dùng (SPSO) và Ban Quản trị kỹ thuật của nhà trường (Administrator)*. Các bên đều có yêu cầu và kỳ vọng cụ thể đối với hệ thống SPSS:

- **Giảng viên, Sinh viên và Học sinh** là nhóm người dùng trực tiếp và cũng là nhóm người dùng chủ yếu của hệ thống. Với nhu cầu lớn trong việc in ấn tài liệu, nhóm người dùng này cần một giao diện tiện lợi, trực quan và dễ sử dụng để tải lên và in các tài liệu, quản lý số trang in còn lại và xem lịch sử in. Học sinh, sinh viên mong muốn sử dụng ứng dụng đủ nhanh, với thời gian trễ không quá cao, đảm bảo tài liệu tải lên được toàn vẹn, không bị mất dữ liệu, bảo mật thông tin cá nhân và tài liệu. Ngoài ra, người dùng có thể gửi phản hồi, khiếu nại cho quản trị viên về các vấn đề gặp phải để được xem xét giải quyết và bồi thường (nếu có).
- **Nhân viên quản lý dịch vụ in cho người dùng (SPSO)** có nhiệm vụ quản lý và vận hành hệ thống in ấn một cách hiệu quả như quản lý hệ thống máy in bao gồm kiểm tra số lượng, tình trạng hoạt động...; cài đặt cấu hình hệ thống như số trang in mặc định, ngày hệ thống gửi số trang mặc định cho toàn bộ người dùng, định dạng file mà hệ thống cho phép người dùng tải lên; xem lịch sử in ấn của người dùng, xem báo cáo sử dụng in ấn, xác thực và quản lý người dùng.
- **Ban Quản trị kỹ thuật của nhà trường (Administrator)** cung cấp tài trợ và hỗ trợ cho dự án, đảm bảo tích hợp với các dịch vụ và hệ thống khác trong trường, đảm bảo tính ổn định và bảo mật của hệ thống, đảm bảo hệ thống hoạt động tốt cho tất cả người dùng, đặc biệt là trong khung giờ cao điểm. Hệ thống phải đảm bảo tuân thủ các quy định của trường và pháp luật.
- **Đội ngũ phát triển ứng dụng (Development Team):** thiết kế và phát triển hệ thống bao gồm việc quản lý cơ sở dữ liệu, lập trình giao diện, kiểm tra và bảo trì hệ thống, hướng dẫn sử dụng hệ thống.
- **Đội hỗ trợ kỹ thuật (Technical Support):** hỗ trợ vận hành cơ sở vật chất của hệ thống SPSS, bao gồm máy in, vật tư tiêu hao, nguồn điện,... cũng như công tác bảo trì thiết bị. Ngoài ra, đội cũng hỗ trợ các vấn đề kỹ thuật cho người dùng khi có sự cố in ấn.
- **Quản trị dịch vụ Xác thực (SSO Administrator):** cung cấp dịch vụ xác thực cho người dùng của trường và SPSO, đảm bảo tính bảo mật và quản lý truy cập vào hệ thống.
- **Quản trị hệ thống thanh toán trực tuyến (Payment Administrator):** cung cấp dịch vụ giao dịch trực tuyến cho người dùng một cách đáng tin cậy khi hệ thống SPSS được tích hợp với hệ thống thanh toán trực tuyến của trường.
- **Bộ phận pháp lý (Legality Department):** Bộ phận pháp lý đảm bảo rằng dự án SPSS tuân thủ hoàn toàn các quy định pháp luật và các quy tắc liên quan đến bảo vệ dữ liệu cá nhân của người dùng. Bộ phận này tham gia vào việc xây dựng các thỏa thuận và hợp đồng liên quan đến việc sử dụng dịch vụ in, đặc biệt là trong việc quản lý nội dung in phù hợp, mua thêm trang in và thanh toán trực tuyến, đảm bảo tính tuân thủ và bảo mật dữ liệu trong toàn bộ quá trình sử dụng dịch vụ.

1.1.4 Lợi ích của hệ thống SPSS đối với mỗi bên liên quan

- **Đối với End users (giảng viên, cán bộ và sinh viên):** được sử dụng dịch vụ thông minh, tiện lợi, tiết kiệm chi phí, thời gian và công sức, đáp ứng kịp thời nhu cầu in tài liệu, báo cáo,... so với trước đây dù là in trong khuôn viên nhà trường hay các tiệm in bên ngoài.
- **Đối với SPSO,** quản lý và giám sát hoạt động in ấn của sinh viên qua hệ thống, tổng hợp các số liệu về dịch vụ để kiểm soát tài nguyên, tài chính, và báo cáo hoạt động với nhà trường, từ đó tạo ra những chính sách phù hợp cho việc quản lý.
- **Đối với ban quản trị nhà trường:** mang đến dịch vụ tiện lợi cho sinh viên, cải thiện chất lượng học tập của sinh viên, tối ưu hóa nguồn cơ sở vật chất của nhà trường (các máy in), giảm thời gian xử lý, quản lý chi phí duy trì hệ thống một cách hiệu quả, cùng với SPSO đề ra những chính sách giúp nâng cao trải nghiệm của người dùng cuối.
- **Đối với đội ngũ phát triển ứng dụng:** xây dựng được một hệ thống có ích cho cộng đồng, có khả năng phát triển và duy trì phần mềm một cách hiệu quả và nhanh chóng, tăng thêm kinh nghiệm, kỹ năng.
- **Đối với đội hỗ trợ kỹ thuật:** thuận tiện trong việc kiểm soát tình trạng hoạt động của máy in, bảo trì thiết bị, và có thể hỗ trợ kịp thời cho người dùng.
- **Đối với quản trị hệ thống thanh toán trực tuyến:** Phát huy sự an toàn, bảo mật trong việc thực hiện giao dịch trực tuyến với người dùng, tích hợp, đồng bộ với các dịch vụ trong hệ sinh thái Trường Đại học lẫn các Trường THPT trên toàn quốc, mang đến quy trình nhanh chóng, tiện lợi, đáng tin cậy cho sinh viên nhà trường.
- **Đối với bộ phận pháp lý:** Đảm bảo tất cả hoạt động của hệ thống SPSS không vi phạm bất cứ quy định nào của pháp luật, đảm bảo quyền và lợi ích hợp pháp của cả bên người dùng và bên trường đại học

1.2 Yêu cầu chức năng và phi chức năng từ mô tả dự án

1.2.1 Yêu cầu chức năng

- **Yêu cầu chung:**

- Phân quyền rõ ràng: Ví dụ, chỉ nhân viên SPSO mới được truy cập lịch sử in của toàn trường, sinh viên chỉ được xem lịch sử cá nhân.

- **Sinh viên**

- Đăng nhập vào hệ thống.
- Đăng xuất khỏi hệ thống.
- Xem thông tin cá nhân của mình (Tên, email, số dư trang in, ...).
- Xem số lượng trang in kích thước A4 còn lại của mình.
- Mua thêm số trang in.
- Tải tài liệu lên hệ thống và chọn máy in ở khu vực có máy in được lắp đặt sẵn ở trường, và thiết lập các thuộc tính in (kích thước giấy, số lượng bản sao, in một mặt hoặc hai mặt, ...)
- Xem thông báo lỗi cụ thể: hết giấy, không kết nối được máy in.
- Xem thông báo nếu tài liệu tải lên không phù hợp (file bị lỗi, dung lượng quá lớn, nội dung không phù hợp).
- Xem thông báo trạng thái in: đang đợi in, in thành công, lỗi.
- Xem vị trí hàng đợi in của mình trên mỗi máy, xem lại lịch sử in của mình, bao gồm số lượng trang đã in, thời gian, máy in đã sử dụng thông qua hệ thống, xem lịch sử giao dịch cá nhân.
- thanh toán qua cổng thanh toán trực tuyến của trường hoặc các cổng thanh toán trực tuyến khác đã liên kết với trường.

- **Nhân viên SPSO**

- Đăng nhập vào hệ thống.
- Đăng xuất khỏi hệ thống.
- Quản lý máy in với các thao tác thêm, vô hiệu hóa trong hệ thống ở khu vực nhà trường.
- Quy định số lượng trang kích thước A4 cấp miễn phí mỗi học kỳ cho sinh viên.
- Xem lịch sử in của sinh viên.
- Xem lịch sử in của máy in.
- Thay đổi số lượng trang in miễn phí, ngày cấp cho mỗi sinh viên vào đầu mỗi học kỳ.
- Xem chi tiết báo cáo hàng tháng, hàng năm về các số liệu sử dụng dịch vụ in tại nhà trường.
- Quản lý các thông báo chung, thông báo lỗi.

1.2.2 Yêu cầu phi chức năng

Thang đo đánh giá độ phức tạp khi hiện thực yêu cầu phi chức năng (NFR) có 5 mức, từ 1 đến 5, trong đó mức 1 là thấp nhất và mức 5 là cao nhất. Thang đo được tính theo thời gian thực hiện chức năng đó:

- **Mức 1:** Khoảng 1-2 ngày
- **Mức 2:** Khoảng 2-4 ngày
- **Mức 3:** Khoảng 4-6 ngày
- **Mức 4:** Khoảng 6-7 ngày
- **Mức 5:** Trên một tuần và cần có sự hỗ trợ

- **Yêu cầu phi chức năng cho sinh viên:**

- **Security:** Không cho phép đăng nhập tài khoản trong 5 phút nếu người dùng nhập sai mật khẩu quá 5 lần. (Mức 1)
- **Performance:**
 - * Thời gian để đăng nhập thành công dưới 5 giây. (Mức 2)
 - * Hiện thị danh sách máy in cho người dùng lựa chọn phải được hiển thị trong vòng không quá 3 giây. (Mức 2)
 - * Lịch sử in ấn của tài khoản người dùng cùng các thông tin liên quan đến tài liệu cùng lần in đó phải hiển thị trong thời gian không quá 6 giây. (Mức 2)
 - * Thời gian tải file để in lên hệ thống phải dưới 10 giây. (Mức 2 hoặc 3 tùy độ lớn của dữ liệu)
- **Usability:** Giao diện đơn giản, có cả tiếng Việt và tiếng Anh (cho người dùng nước ngoài) (Mức 1)
- **Availability:** Bảo đảm hệ thống hoạt động 24/7, hạn chế tình trạng sập khi lượng người truy cập cao. (Mức 2)
- **Supportability:** Ứng dụng web có responsive, có thể sử dụng trên trình duyệt của các thiết bị có kích thước màn hình khác nhau. (Mức 2)
- **User Interface:** Có câu hỏi Y/N xác nhận trước khi thực hiện các bước thực hiện in. (Mức 1)

- **Yêu cầu phi chức năng cho quản lý hệ thống:**

- **Security:**
 - * Các thao tác liên quan đến bảo mật khách hàng phải yêu cầu nhập lại mật khẩu, nếu sai quá 3 lần hệ thống phải tự đăng xuất và không cho phép đăng nhập tiếp trong 5 phút. (Mức 2)
 - * Mật khẩu của tài khoản SPSO phải đảm bảo có ít nhất 6 ký tự, gồm cả chữ hoa, chữ thường, số. (Mức 1)
- **Recovery:**
 - * Hệ thống có khả năng backup dữ liệu. (Mức 2)
 - * Thường xuyên backup với tần suất 4 ngày/lần. (Mức 3)
 - * Có chức năng tìm lại mật khẩu nếu quên. (Mức 1)
- **Performance:**
 - * Hiện thị lịch sử in ấn của một tài khoản người dùng hoặc một máy in cụ thể dưới 7 giây. (Mức 1)
 - * Hiện thị toàn bộ lịch sử in ấn của toàn hệ thống trong một học kỳ trong dưới 10 giây. (Mức 2 hoặc 3)
- **Usability:** Giao diện đơn giản, không quá phức tạp, người mới có thể sử dụng website chỉ sau 5 phút tìm hiểu, giao diện có cả tiếng Việt và tiếng Anh. (Mức 1)
- **Online Manual:** Trang hướng dẫn phải ở vị trí trực quan như thanh Menu hoặc nút hỗ trợ nổi bật. (Mức 1)
- **Audit:** Các tác vụ, hành vi liên quan đến máy in phải được lưu trữ trên hệ thống để thuận tiện kiểm tra sau này. (Mức 2)

- **In ấn hiệu quả (Print Efficiency):**

- **Ưu tiên xếp hàng in (Print Queue Management):**
 - * Hệ thống phải hỗ trợ xếp hàng in theo thứ tự thời gian yêu cầu hoặc ưu tiên (ví dụ: giảng viên có thể được ưu tiên cao hơn sinh viên).
 - * Cho phép người dùng kiểm tra trạng thái của hàng chờ và ước tính thời gian hoàn thành.

- **Phân tích và tối ưu hóa tài nguyên in:** Hệ thống phải cảnh báo người dùng nếu tài liệu có thể tối ưu để tiết kiệm giấy (ví dụ: in hai mặt, in nhiều trang trên một tờ).
- **Khả năng truy cập (Availability):**
 - **Thời gian hoạt động (Uptime):** Hệ thống phải đảm bảo hoạt động **99.9% thời gian trong tháng**, tương đương với thời gian ngừng hoạt động không quá 43 phút/tháng.
 - **Thời gian phản hồi (Response Time):**
 - * Đăng nhập không quá **3 giây** trong 95% số lần truy cập. Thao tác tải lên tệp và chọn máy in không quá **5 giây** trong 95% số lần thao tác.
 - * **Khả năng mở rộng (Scalability):** Hệ thống phải hỗ trợ ít nhất **1.000 người dùng đồng thời** mà không làm giảm hiệu năng.
 - **Hỗ trợ đa nền tảng (Platform Support):** Ứng dụng phải chạy ổn định trên các trình duyệt phổ biến (Chrome, Edge, Firefox) và hệ điều hành di động (iOS, Android).
- **Bảo mật và xác thực (Security and Authentication):**
 - **Xác thực và ủy quyền (Authentication và Authorization):**
 - * Tài khoản sinh viên phải được bảo mật và xác thực qua Dịch vụ xác thực trước khi truy cập, với **thời gian xác thực không quá 10 giây**.
 - * Phân quyền rõ ràng: Chỉ những người dùng được ủy quyền mới có thể truy cập thông tin cá nhân và lịch sử in ấn của sinh viên.
 - **Bảo vệ dữ liệu (Data protection):**
 - * **Mã hóa dữ liệu:** Dữ liệu cá nhân và tài liệu in phải được mã hóa sử dụng **AES-256**.
 - * **Truyền tải an toàn:** Hệ thống phải sử dụng giao thức HTTPS để bảo mật thông tin trong quá trình truyền tải.
 - **Theo dõi và ghi nhật ký (Audit và Logging):**
 - * Hệ thống phải ghi lại 100% các thao tác quan trọng (đăng nhập, tải lên, in ấn) với thời gian lưu trữ nhật ký ít nhất 6 tháng.
 - * Phải có công cụ để truy vết và xử lý sự cố bảo mật trong vòng 24 giờ.
- **Bảo mật dữ liệu (Data security):**
 - **Quản lý quyền truy cập:** Chỉ những người dùng được ủy quyền mới được phép truy cập thông tin cá nhân và lịch sử in ấn.
 - **Sao lưu dữ liệu (Backup):** Dữ liệu phải được sao lưu tự động hàng ngày và có khả năng khôi phục trong vòng 4 giờ sau sự cố.
 - **An toàn khi xóa:** Khi xóa tài liệu, thông tin phải bị hủy vĩnh viễn bằng cách ghi đè dữ liệu (data overwriting).
- **Thân thiện với người dùng (Usability):**
 - **Giao diện người dùng (User Interface):** Thân thiện, dễ sử dụng, với thời gian làm quen cho người dùng mới không quá 10 phút.
 - **Định dạng tệp hỗ trợ:** Hệ thống phải hỗ trợ ít nhất 90% các định dạng tệp phổ biến cho việc in, bao gồm PDF, DOCX, PPTX, JPEG, PNG.
 - **Trợ giúp người dùng (Help and Support):** Tích hợp phần "Hướng dẫn sử dụng" và hỗ trợ trực tuyến qua chat hoặc email với thời gian phản hồi không quá 6 giờ làm việc.
- **Tính toàn vẹn dữ liệu (Data integrity):**
 - **Độ chính xác dữ liệu:** Tỷ lệ lỗi dữ liệu phải thấp hơn 0.01% trong tổng số giao dịch thực hiện.
 - **Đồng bộ dữ liệu:** Dữ liệu phải được cập nhật trong thời gian thực hoặc không quá 5 giây sau khi hoàn thành một thao tác.

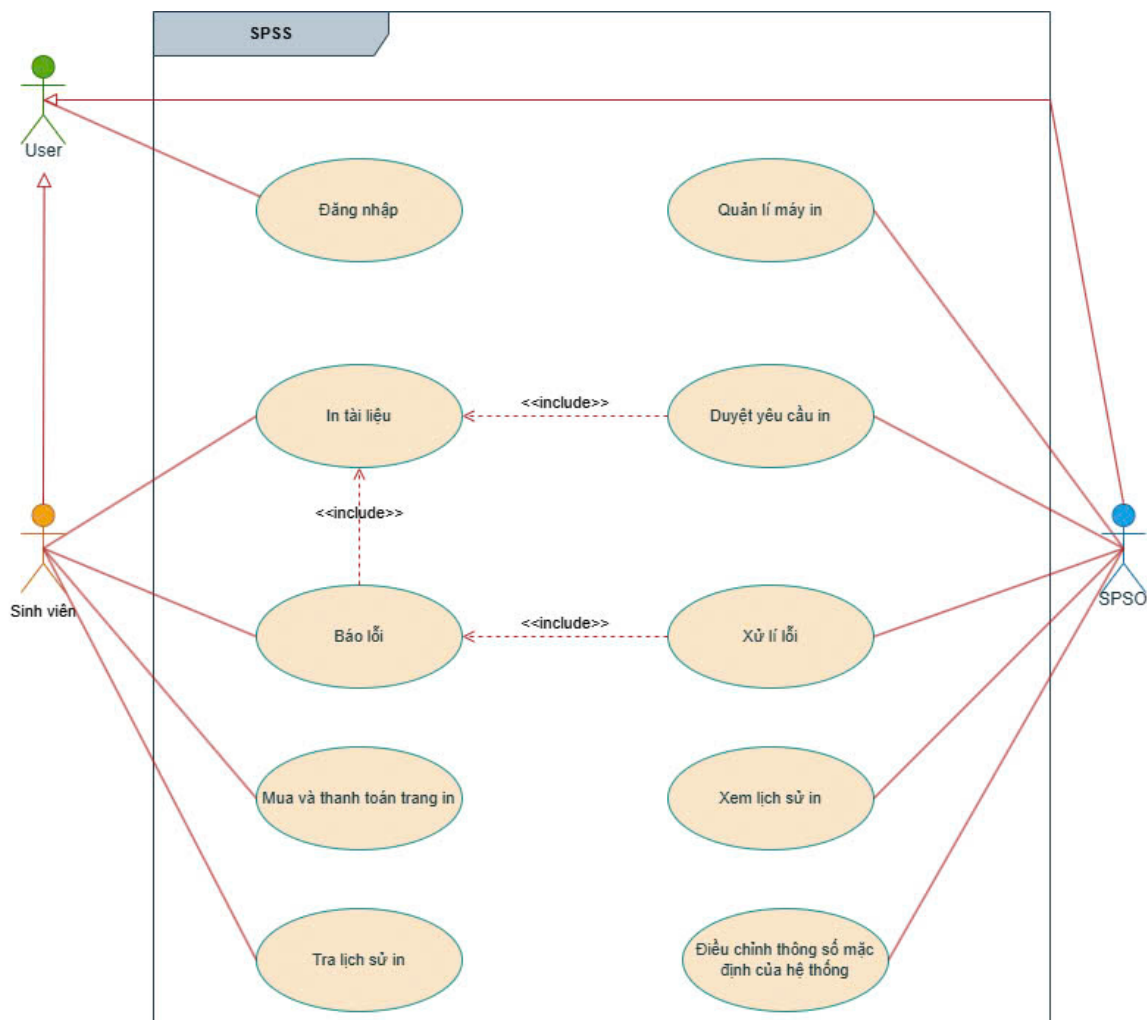
- **Kiểm tra dữ liệu:** Hệ thống phải tự động phát hiện và khắc phục các lỗi cơ bản (ví dụ: mất dữ liệu hoặc dữ liệu bị hỏng) trong vòng 10 phút.
- **Hiệu năng và hiệu quả (Performance và Efficiency):**
 - **Tài nguyên hệ thống:** Ứng dụng không được sử dụng quá 70% CPU hoặc RAM của máy chủ khi đạt tải tối đa.
 - **Tiết kiệm chi phí in:** Hệ thống phải tối ưu hóa lệnh in để giảm thiểu lượng mực và giấy thừa, thông báo lỗi in trước khi thực hiện.

1.3 Những tính năng được hiện thực

- **Quản lý Máy In:**
 - **Xem Thông Tin Máy In:** Cho phép quản trị viên xem và quản lý toàn bộ thông tin của máy in từ toàn cục đến chi tiết.
 - **Lọc Thông Tin Máy In:** Cho phép quản trị viên tìm kiếm thông tin máy in dựa trên Hãng, Model, ID máy in, Trọng lượng, Trạng thái máy in.
 - **Thêm Máy In Mới:** Cho phép quản trị viên thêm máy in mới vào hệ thống thông qua giao diện web. Nhập thông tin chi tiết như tên máy in, loại máy in và các thông số kỹ thuật khác.
 - **Sửa Thông Tin Máy In:** Cập nhật thông tin máy in hiện tại khi có sự thay đổi về cấu hình. Thay đổi thông tin như tên máy in, hoặc cập nhật phần mềm điều khiển (driver).
 - **Xóa Máy In:** Cho phép xóa máy in khỏi hệ thống khi không còn sử dụng. Xác nhận trước khi xóa để tránh việc xóa nhầm máy in quan trọng.
- **Quản lý Trạng Thái Máy In:**
 - **Kích hoạt/Vô hiệu hóa Máy In:** Tạm dừng hoặc kích hoạt máy in để ngăn người dùng in khi máy đang trong quá trình bảo trì hoặc gặp sự cố. Cung cấp lý do tạm dừng (ví dụ: bảo trì định kỳ, sửa chữa phần cứng, chưa thay mực,...) để người dùng hiểu rõ nguyên nhân.
 - **Thông Báo Trạng Thái:** Hiển thị thông báo về trạng thái máy in trên giao diện web. Gửi thông báo qua hệ thống khi trạng thái máy in thay đổi.
- Quản lý Hàng Đợi In:**
 - **Theo dõi Hàng đợi:** Hiển thị danh sách các máy in và số hàng đợi của từng máy in. Khi xem xét riêng từng máy in, có thể biết hàng đợi những yêu cầu in được gửi tới máy in đó.
 - **Quản lý Hàng đợi:** Xác nhận những yêu cầu in ở từng máy in và xác nhận in.
- **Quản lý Người Dùng và Quyền Truy Cập:**
 - **Phân Quyền Người Dùng:** Tạo và quản lý tài khoản người dùng với các quyền hạn khác nhau (ví dụ: quản trị viên, người dùng thường). Kiểm soát quyền truy cập vào các máy in cụ thể dựa trên vai trò hoặc bộ phận.
 - **Xác Thực Người Dùng:** Sử dụng các phương thức xác thực như tên người dùng/mật khẩu để đảm bảo an ninh.
- **Quản lý Lịch Sử In:**
 - **Xem Lịch Sử In:** Cho phép quản trị viên xem và quản lý toàn bộ lịch sử in.
 - **Lọc Thông Tin Lịch Sử In:** Cho phép quản trị viên tìm kiếm lịch sử in dựa trên tên người dùng, Model, ID người dùng, máy in,... Đồng thời cho phép quản trị viên tìm kiếm lịch sử in trong khoảng thời gian nhất định.
 - **Xem Lịch Sử In Của Máy In:** Cho phép quản trị viên xem thông tin về lịch sử in của máy in cụ thể.
 - **Xóa Lịch Sử In:** Cho phép xóa lịch sử in khi không cần thiết.

2 Use-case Diagram

2.1 Use-case diagram cho toàn bộ hệ thống



Hình 2.1: Use-case diagram toàn bộ hệ thống

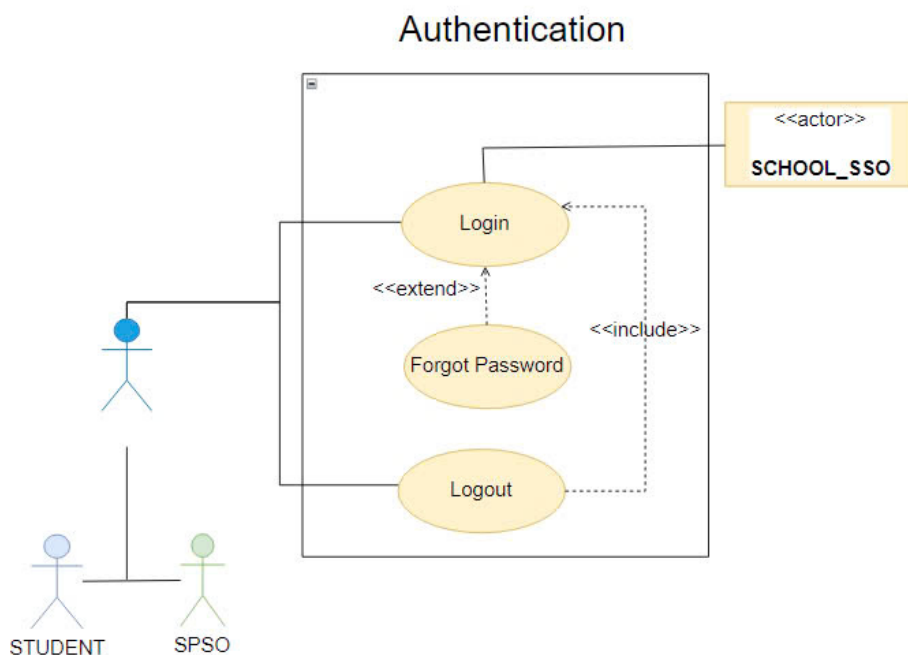
Use-case diagram tổng quát của toàn bộ hệ thống được mô tả trong hình 3.1. Diagram gồm 2 actor kế thừa từ actor User là Sinh viên và SPSO.

Sau khi đăng nhập thành công, sinh viên có thể tiến hành in tài liệu bằng cách tải tài liệu cần in lên hệ thống và tiến hành điều chỉnh các thông số cần thiết cho bản in (khổ giấy, số bản in,...), sau đó gửi yêu cầu in của mình lên hệ thống để SPSO xét duyệt, sau khi SPSO xét duyệt thì tài liệu sẽ được in. Trong quá trình in nếu xảy ra lỗi, sinh viên cũng có thể gửi báo lỗi để SPSO xử lý. Ngoài ra, nếu không đủ trang in, sinh viên có thể tiến hành mua 1 số lượng trang in nhất định. Sinh viên cũng có thể xem lại lịch sử các file mình đã in khi cần thiết.

SPSO: ngoài 2 use case là xử lý lỗi và duyệt yêu cầu in thì actor này còn có thể thêm hoặc xóa 1 vài máy in vào hệ thống hoặc bật tắt máy in khi cần thiết, máy in bị tắt là máy in vẫn nằm trên hệ thống nhưng sinh viên không thể đăng ký in tại máy in này. Ngoài ra, SPSO còn có thể xem lịch sử in cho từng sinh viên hoặc từng máy in cũng như điều chỉnh các thông số mặc định của hệ thống như: số trang in miễn phí cho sinh viên, định dạng tệp in.

2.2 Use-case diagram cho Module Xác Thực

- Use-case diagram



Hình 2.2: Use-case xác thực và phân quyền

- Bảng mô tả use-case

Bảng 3: Mô tả use case - Login

Usecase name	Login
Actor	Người dùng, Quản lý hệ thống
Preconditions	1. Người dùng có tài khoản hợp lệ. 2. Tài khoản người dùng đã được phân quyền. 3. Thiết bị của người dùng đã được kết nối internet.
Trigger	Người dùng nhấn vào nút " Đăng nhập "
Main flow	1. Người dùng nhập thông tin đăng nhập và thực hiện lệnh đăng nhập. 2. Hệ thống xác thực thông tin đăng nhập. 3. Sau khi xác thực thành công, hệ thống cho phép người dùng truy cập ứng dụng.
Postconditions	1. Người dùng đăng nhập hệ thống thành công. 2. Hệ thống ghi nhận hoạt động đăng nhập thành công.
Exception Flow	3E. Hệ thống xác thực thông tin không thành công: a. Hiện thị thông báo lỗi, yêu cầu người dùng thực hiện lại bước 1. b. Người dùng chọn lệnh lấy lại mật khẩu. .
Note	

Bảng 4: Mô tả use case - *Forgot password*

Usecase name	Forgot Password
Actor	Người dùng, Quản lí hệ thống
Preconditions	1. Người dùng đã đăng kí và có tài khoản hợp lệ. 2. Thiết bị của người dùng đã được kết nối internet.
Trigger	Người dùng nhấn vào nút " Quên mật khẩu "
Main flow	1. Người dùng thực hiện lệnh quên mật khẩu. 2. Hệ thống yêu cầu người dùng nhập email hoặc tên đăng nhập. 3. Sau khi kiểm tra email hoặc tên đăng nhập thành công, hệ thống gửi đường dẫn đặt lại mật khẩu cho người dùng thông qua email. 4. Người dùng nhấp vào đường dẫn trên để vào trang thay đổi mật khẩu. 5. Hệ thống kiểm tra mật khẩu mới hợp lệ, gửi thông tin thay đổi cho hệ thống.
Postconditions	1. Người dùng nhận được đường dẫn thay đổi mật khẩu. 2. Hệ thống ghi nhận hoạt động thay đổi đăng nhập thành công. 3. Người dùng đăng nhập hệ thống với mật khẩu mới thành công.
Exception Flow	3a. Hệ thống xác thực thông tin email hoặc tên đăng nhập không hợp lệ và hiển thị thông báo, yêu cầu người dùng thực hiện lại bước 1. 3a1. Người dùng thoát trang web. Use Case dừng lại.
Note	

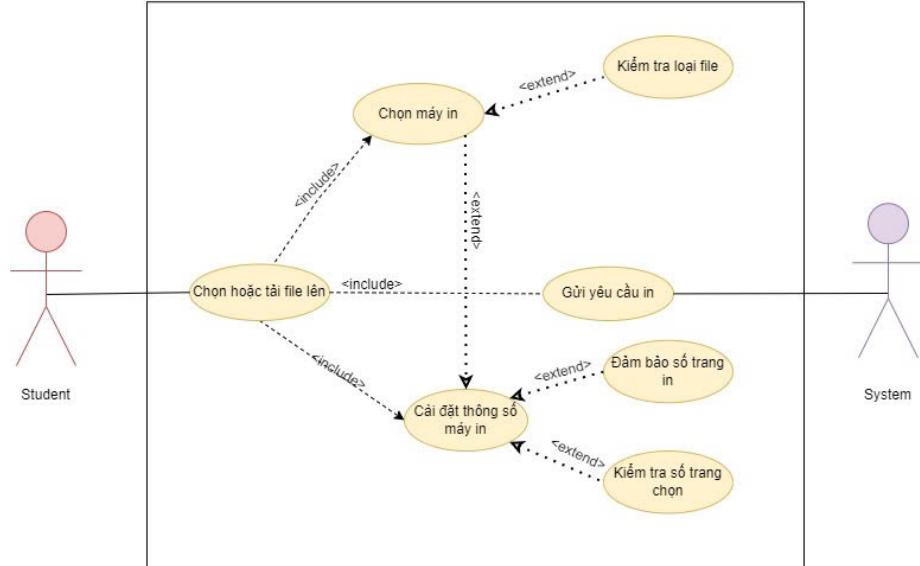
Bảng 5: Mô tả use case - *Log out*

Usecase name	Log Out
Actor	Người dùng, Quản lí hệ thống
Preconditions	1. Người dùng đã đăng kí và có tài khoản hợp lệ. 2. Thiết bị của người dùng đã được kết nối internet.
Trigger	Người dùng nhấn vào nút " Đăng xuất "
Main flow	1. Người dùng thực hiện lệnh đăng xuất. 2. Hệ thống xác thực thông tin đăng xuất thành công, và cho phép người dùng thoát khỏi hệ thống. 3. Người dùng được đưa về trang đăng nhập.
Postconditions	1. Người dùng đăng xuất hệ thống thành công. 2. Hệ thống ghi nhận hoạt động đăng xuất thành công.
Exception Flow	2a. Hệ thống xác thực thông tin đăng xuất không thành công và hiển thị thông báo, yêu cầu người dùng thực hiện lại bước 1. 2a1. Người dùng thoát trang web. Use Case dừng lại.
Note	

2.3 Use-case diagram cho Module Quản lý in

2.3.1 Đăng ký in

- Use-case diagram



Hình 2.3: Use-case đăng ký in

- Bảng mô tả use-case

Bảng 6: Mô tả use case - Tải file

Usecase name	Tải file
Actor	Sinh viên sử dụng dịch vụ in
Preconditions	1. Người dùng có tài khoản và đã đăng nhập vào dịch vụ in. 2. Thiết bị in đã được kết nối và sẵn sàng nhận tệp.
Trigger	Người dùng chọn "Browse file or upload".
Main flow	1. Người dùng nhấp vào nút "Browse file or upload". 2. Hệ thống hiển thị hộp thoại chọn tệp. 3. Người dùng chọn một hoặc nhiều tệp để tải lên. 4. Người dùng kiểm tra file đã tải lên thành công bằng nút "Xem file"
Postconditions	1. Tệp đã được tải lên thành công vào hệ thống. 2. Một thông báo xác nhận được hiển thị cho người dùng.
Exception Flow	2a. Nếu định dạng tệp không được hỗ trợ, hệ thống sẽ hiển thị thông báo lỗi yêu cầu người dùng tải lên định dạng tệp tương thích.
Note	

Bảng 7: Mô tả use case - Chọn máy in và cài đặt thông số in

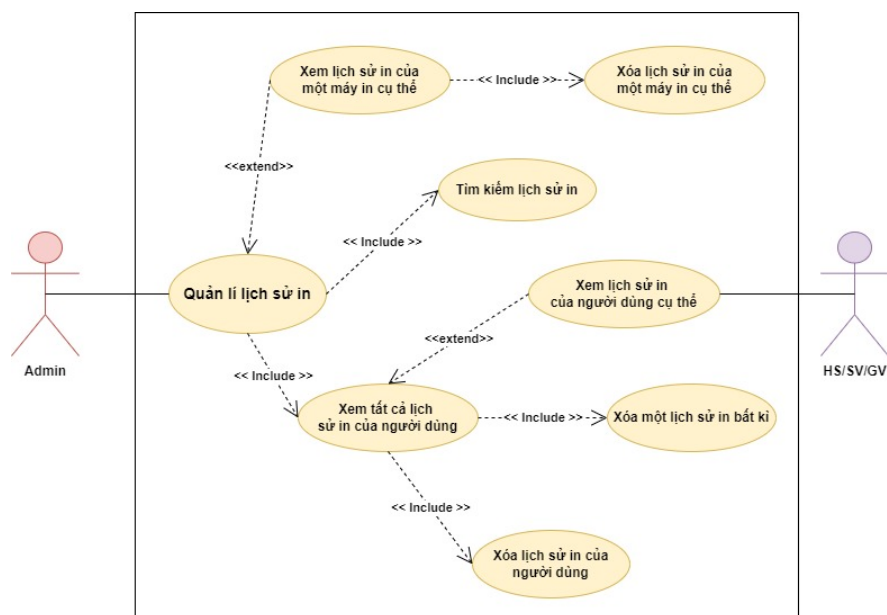
Usecase name	Chọn máy in và Cài đặt thông số in
Actor	Sinh viên sử dụng dịch vụ in
Preconditions	1. Người dùng đăng nhập và xác thực thành công bằng hệ thống quản lý tài khoản. 2. Người dùng đã upload file thành công. .
Trigger	Người dùng ở Tab “Tải file lên” tiến hành chọn các thông số in phù hợp sau đó Người dùng bấm vào Tab “Chọn máy in” rồi “Chọn” để chọn máy in
Main flow	1.Ở tab “Tải file lên” Người dùng chọn thông số in thích hợp 2. Người dùng bấm nút Tab “Chọn máy in”. 3. Hệ thống hiển thị danh sách các máy in. 4. Người dùng bấm chọn máy in phù hợp.
Postconditions	Người dùng chọn thành công máy in và cấu hình máy in phù hợp.
Exception Flow	3a, 4a. Tất cả các máy in trong hệ thống đang được bảo trì hoặc bị tắt bởi SPSO thì không có máy in nào sẵn sàng và không thể tiếp tục đăng kí in
Note	

Bảng 8: Mô tả use case - In tài liệu

Usecase name	In tài liệu
Actor	Sinh viên sử dụng dịch vụ in
Preconditions	1. Người dùng đăng nhập và xác thực thành công bằng hệ thống SCHOOL-SSO. 2. Người dùng đã upload file thành công. 3. Người dùng đã chỉnh sửa các thuộc tính tài liệu để chuẩn bị in. 4. Người dùng đã chọn máy in
Trigger	Người dùng nhấn vào “In tài liệu”
Main flow	1. Người dùng nhấn nút In tài liệu 2. Hệ thống xác nhận đăng kí và chuyển vào hàng đợi chờ admin duyệt.
Postconditions	Người dùng in thành công tài liệu
Exception Flow	1a. Nếu số trang in không đủ, hệ thống sẽ thông báo, in không thành công và hỏi mua thêm trang in 2a. Khi đó sẽ chuyển đến giao diện mua trang in và thanh toán 1b. Nếu cấu hình bị lỗi hoặc chưa chọn máy in, hệ thống sẽ hiển thị thông báo yêu cầu chọn lại cấu hình in đúng và máy in phù hợp
Note	

2.3.2 Quản lí lịch sử in

- Use-case diagram



Hình 2.4: Use-case quản lí lịch sử in

- Bảng mô tả use-case Lịch sử in của user - HS/SV/GV

Bảng 9: Mô tả use case - Xem lịch sử in cá nhân

Usecase name	Xem lịch sử in cá nhân
Actors	Sinh viên, Giảng viên
Precondition	Người dùng đã đăng nhập vào ứng dụng với vai trò là student
Trigger	Người dùng nhấn vào nút " Lịch Sử In " trên header trang.
Main flow	1. Hệ thống truy vấn cơ sở dữ liệu để lấy danh sách đơn in của người dùng. 2. Hệ thống hiển thị danh sách đơn in theo các thông tin: Máy in, Thời gian in, Trang/Khổ giấy, File In, Tình trạng thời gian in, hành động trên màn hình. Cùng thông tin user như tên, email, số dư trang in,..
Alternate Flow	2A. Người dùng lọc danh sách theo ngày hoặc tìm kiếm các thông tin theo cột. 2B. Hệ thống hiển thị kết quả lọc theo yêu cầu. 2C. Người dùng tiếp tục xem dữ liệu.
Postcondition	Người dùng có thể xem danh sách đơn in cá nhân
Exception Flow	1A. Hệ thống truy xuất dữ liệu không thành công. 1B. Hệ thống hiển thị thông báo lỗi "Không thể tải dữ liệu, vui lòng thử lại". 1C. Người dùng thử lại hoặc thoát khỏi giao diện.
Note	

- Bảng mô tả use-case Lịch sử in - SPSO

Bảng 10: Mô tả use case - Xem lịch sử in tổng

Usecase name	Xem lịch sử in tổng
Actors	SPSO
Precondition	Người dùng đã đăng nhập vào ứng dụng với vai trò là SV/GV
Postcondition	Admin có thể xem danh sách đơn in hệ thống
Trigger	Người dùng nhấn vào nút " Lịch Sử In " trên header trang.
Main flow	<ol style="list-style-type: none"> Hệ thống truy vấn cơ sở dữ liệu để lấy danh sách đơn in của cả hệ thống. Hệ thống hiển thị danh sách đơn in theo các thông tin: tên người dùng, ID user, số trang/khổ giấy, máy in, thời gian in, hành động trên màn hình. SPSO có thể sử dụng chức năng lọc để tìm kiếm lịch sử đơn in theo tiêu chí như ngày tháng, máy in, người dùng,... Hệ thống hiển thị kết quả lọc và SPSO có thể chọn xem chi tiết các đơn in từ kết quả.
Alternate Flow	<ol style="list-style-type: none"> Người dùng lọc danh sách theo ngày tháng, người dùng, máy in. Hệ thống hiển thị kết quả lọc theo yêu cầu. Người dùng tiếp tục xem dữ liệu.
Exception Flow	<ol style="list-style-type: none"> Hệ thống truy xuất dữ liệu không thành công. Hệ thống hiển thị thông báo lỗi "Không thể tải dữ liệu, vui lòng thử lại".
Note	

Bảng 11: Mô tả use case - Xem lịch sử in của máy in

Tên use-case	Xem lịch sử in của máy in
Actor	Admin
Description	Admin xem lịch sử in của máy in
Precondition	<ol style="list-style-type: none"> Đã đăng nhập với role là admin, spso. Admin đang ở trang "Thông tin chi tiết của máy in".
Postcondition	Hệ thống hiển thị toàn bộ lịch sử in của máy in.
Trigger	Admin nhấn nút " Chi tiết – ! " của Lịch sử in.
Basic Flows	<ol style="list-style-type: none"> Hệ thống gửi thông báo "Bạn có chắc muốn xem lịch sử in của máy in này không?". Admin nhấn nút "Xác nhận" và hệ thống điều hướng sang trang xem lịch sử máy in. Hệ thống hiển thị thông tin cơ sở của máy in và lịch sử in của máy in.
Alternative Flows	<ol style="list-style-type: none"> Admin nhấn nút "Hủy" và không có gì xảy ra. Admin nhấn nút "Xóa lịch sử in". Admin nhấn nút "Chi tiết – !" của Printer's Info.
Exception Flows	<ol style="list-style-type: none"> Hệ thống báo lỗi "Bạn không thể xem thông tin máy in lúc này!" nếu xảy ra lỗi. Hệ thống hiển thị "N/A" cho các thông tin bị lỗi.

Bảng 12: Mô tả use case - Xóa lịch sử in người dùng

Tên use-case	Xóa lịch sử in của người dùng
Actor	Admin
Description	Admin xóa lịch sử in của người dùng
Precondition	1. Đã đăng nhập với role là admin. 2. Admin đang ở trang xem lịch sử in toàn hệ thống. 3. Hệ thống chỉ hiển thị các yêu cầu in chưa bị xóa.
Postcondition	Hệ thống xóa toàn bộ lịch sử in của người dùng và hiển thị danh sách trống các yêu cầu in.
Trigger	Admin nhấn nút " Xóa lịch sử in " ở dưới danh sách các đơn in của người dùng.
Basic Flows	1. Admin nhấn vào nút "Lịch sử in" ở phần thanh điều hướng. 2. Tại cột Thao tác của mỗi bản ghi, nhấn vào icon "Thông tin" để hiển thị tất cả các yêu cầu in của người dùng. 3. Nhấn nút "Xóa lịch sử in" để xóa tất cả các yêu cầu in. 4. Hệ thống hiển thị thông báo "Xóa lịch sử in thành công".
Alternative Flows	2a. Admin có thể xóa từng yêu cầu in bằng cách nhấn nút X ở cột Thao tác. 2b. Xác nhận để thực hiện xóa và quay lại bước 2.
Exception Flows	2.1. Nếu chưa có lịch sử in, hệ thống hiển thị thông báo "Chưa có lịch sử in". 3E. Hệ thống báo lỗi "Không thể xóa lịch sử in lúc này!" nếu xảy ra lỗi.

Bảng 13: Mô tả use case - Tìm kiếm lịch sử in

Tên use-case	Tìm kiếm lịch sử in
Actor	Admin
Description	Admin tìm kiếm lịch sử in người dùng trong khoảng thời gian cụ thể.
Precondition	1. Đã đăng nhập với role là admin. 2. Admin đang ở trang xem lịch sử in toàn hệ thống.
Postcondition	Hệ thống chỉ hiển thị danh sách các yêu cầu in của người dùng trong khoảng thời gian đã chọn.
Trigger	Admin nhấn vào nút " Lọc " ở trên danh sách các yêu cầu in.
Basic Flows	1. Admin nhấn vào nút "Lịch sử in" ở phần thanh điều hướng. 2. Admin chọn ngày bắt đầu và ngày kết thúc tại 2 ô nhập thời gian tương ứng. 3. Nhấn vào nút "Lọc" để hiển thị kết quả. 4. Hệ thống hiển thị danh sách các yêu cầu in trong khoảng thời gian đã chọn.
Alternative Flows	None
Exception Flows	3.1. Ở bước 3, nếu không có kết quả, hệ thống để trống danh sách lịch sử in và hiển thị thông báo "Không tìm thấy lịch sử in phù hợp".

Bảng 14: Mô tả use case - Xem tất cả lịch sử in của người dùng

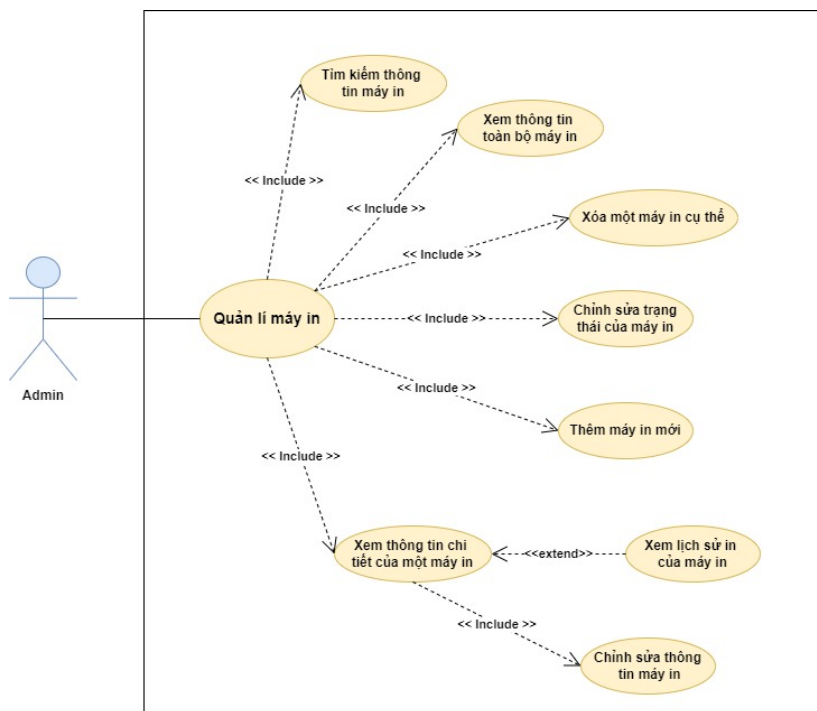
Tên use-case	Xem tất cả lịch sử in của người dùng
Actor	Admin
Description	Cho phép Admin xem toàn bộ lịch sử in của người dùng.
Precondition	1. Admin đã đăng nhập vào hệ thống quản lý.
Postcondition	Admin có thể xem danh sách toàn bộ lịch sử in của người dùng.
Trigger	Admin nhấn vào nút " Lịch sử in " ở trên Thanh điều hướng.
Basic Flows	1. Admin chọn "Lịch sử in" ở thanh điều hướng. 2. Hệ thống hiển thị danh sách toàn bộ lịch sử in của người dùng. 3. Admin có thể tìm kiếm và lọc các lịch sử in theo các tiêu chí khác nhau.
Alternative Flows	None
Exception Flows	2.1. Nếu không có lịch sử in nào trong hệ thống, hệ thống sẽ hiển thị thông báo "Không có lịch sử in nào".

Bảng 15: Mô tả use case - Xem tất cả lịch sử in bất kỳ của người dùng

Tên use-case	Xóa lịch sử in bất kỳ của người dùng
Actor	Admin
Description	Cho phép Admin xóa lịch sử in cụ thể của người dùng.
Precondition	1. Admin đã đăng nhập vào hệ thống quản lý.
Postcondition	Lịch sử in cụ thể của người dùng bị xóa khỏi hệ thống.
Trigger	Admin nhấn vào nút " x " ở cột Thao tác với mỗi bản ghi.
Basic Flows	1. Admin chọn chức năng quản lý lịch sử in. 2. Hệ thống hiển thị danh sách lịch sử in của người dùng. 3. Admin xác định lịch sử in cần xóa và nhấn nút "x" ở cột Thao tác. 4. Hệ thống hiển thị thông báo xác nhận việc xóa. 5. Admin xác nhận xóa. 6. Hệ thống xóa lịch sử in khỏi cơ sở dữ liệu và hiển thị thông báo thành công.
Alternative Flows	None
Exception Flows	2.1. Nếu không có lịch sử in nào trong hệ thống, hệ thống sẽ hiển thị thông báo "Không có lịch sử in nào". 5.1. Nếu có lỗi khi xóa, hệ thống sẽ thông báo lỗi "Không thể xóa lịch sử in" cho Admin.

2.4 Use-case diagram cho Module Quản lý máy in

- Use-case diagram



Hình 2.5: Use-case Quản lý máy in

- Bảng mô tả use-case

Bảng 16: Mô tả use case - Xem máy in tổng

Tên use-case	Xem máy in tổng
Actor	Admin
Description	Admin xem thông tin của máy in theo yêu cầu
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin tìm kiếm thông tin máy in bằng SearchBar.
Postcondition	Hệ thống hiển thị thông tin chung của tất cả máy in
Trigger	Admin nhấn vào nút “Quản lý máy in” trên thanh điều hướng
Basic Flows	1. Admin tìm kiếm thông tin máy in theo Hãng, Model, ID, Vị trí, Trọng lượng, Trạng thái. 2. Hệ thống hiển thị danh sách các máy in ứng với yêu cầu tìm kiếm.
Alternative Flows	None
Exception Flows	2E. Hiển thị “Không có dữ liệu” nếu không có máy in đáp ứng yêu cầu tìm kiếm.

Bảng 17: Mô tả use case - Tìm kiếm thông tin máy in

Tên use-case	Tìm kiếm thông tin máy in
Actor	Admin
Description	Admin xem thông tin của máy in theo yêu cầu
Precondition	<ul style="list-style-type: none"> – Đã đăng nhập với role là admin, spso. – Admin tìm kiếm thông tin máy in bằng SearchBar.
Postcondition	Hệ thống hiển thị thông tin chung của tất cả máy in
Trigger	Admin nhấn vào nút “Quản lý máy in” trên thanh điều hướng
Basic Flows	<ul style="list-style-type: none"> – Admin tìm kiếm thông tin máy in theo Hãng, Model, ID, Vị trí, Trọng lượng, Trạng thái. – Hệ thống hiển thị danh sách các máy in ứng với yêu cầu tìm kiếm.
Alternative Flows	None
Exception Flows	<ul style="list-style-type: none"> – 2E. Hiển thị “Không có dữ liệu” nếu không có máy in đáp ứng yêu cầu tìm kiếm.

Bảng 18: Mô tả use case - Xem thông tin chi tiết từng máy in

Tên use-case	Xem thông tin chi tiết từng máy in
Actor	Admin
Description	Admin xem thông tin chi tiết của từng máy in
Precondition	<ol style="list-style-type: none"> 1. Admin đang ở trang quản lý máy in tổng. 2. Admin nhấn nút “Thông tin - !” của máy in cụ thể.
Postcondition	Hệ thống hiển thị toàn bộ thông số kỹ thuật và các dữ liệu vận hành khác của máy in
Trigger	Admin nhấn vào nút “Thông tin - !” ở máy in muốn xem
Basic Flows	<ol style="list-style-type: none"> 1. Admin nhấn nút “Thông tin - !” của máy in cụ thể. 2. Hệ thống hiển thị thông tin chi tiết của máy in bao gồm các thông số kỹ thuật và các dữ liệu vận hành khác như hàng đợi, số lượt in trong ngày,...
Alternative Flows	<ol style="list-style-type: none"> 1. Admin nhấn nút “Chỉnh sửa”. 2. Admin nhấn nút “Chi tiết - !” của Lịch sử in.
Exception Flows	<ol style="list-style-type: none"> 1. Ở bước 2, nếu không có máy in nào, hệ thống hiển thị “Không có kết quả!”.

Bảng 19: Mô tả use case - Xóa một máy in cụ thể

Tên use-case	Xóa một máy in cụ thể
Actor	Admin
Description	Admin xóa một máy in cụ thể
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin đang ở trang Quản lý máy in tổng.
Postcondition	Hệ thống xóa máy in ra khỏi hệ thống
Trigger	Admin nhấn nút “Xóa – x” của máy in cụ thể
Basic Flows	1. Hệ thống hiển thị thông báo “Bạn có chắc muốn xóa máy in này?”. 2. Admin nhấn nút “Xác nhận”. 3. Hệ thống xóa máy in ra khỏi hệ thống và gửi thông báo “Xóa máy in thành công”.
Alternative Flows	2.1. Admin nhấn nút “Hủy” và không có chuyện gì xảy ra.
Exception Flows	2E. Hệ thống hiển thị “Bạn không thể xóa máy in lúc này” nếu xảy ra lỗi không thể xóa máy in.

Bảng 20: Mô tả use case - Thay đổi trạng thái máy in

Tên use-case	Thay đổi trạng thái máy in
Actor	Admin
Description	Admin thay đổi trạng thái của máy in
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin đang ở trang Quản lý máy in tổng.
Postcondition	Hệ thống cập nhật trạng thái mới cho máy in
Trigger	Admin nhấn nút “Bật/Tắt trạng thái máy in”
Basic Flows	1. Hệ thống gửi thông báo “Bạn có chắc thay đổi trạng thái máy in này không?”. 2. Admin nhấn nút “Xác nhận”. 3. Hệ thống cập nhật trạng thái mới cho máy in.
Alternative Flows	Admin nhấn nút “Hủy” và không có gì thay đổi.
Exception Flows	3E. Hệ thống báo lỗi “Bạn không thể thay đổi trạng thái máy in lúc này” nếu xảy ra lỗi không thể cập nhật trạng thái mới.

Bảng 21: Mô tả use case - Thêm máy in mới

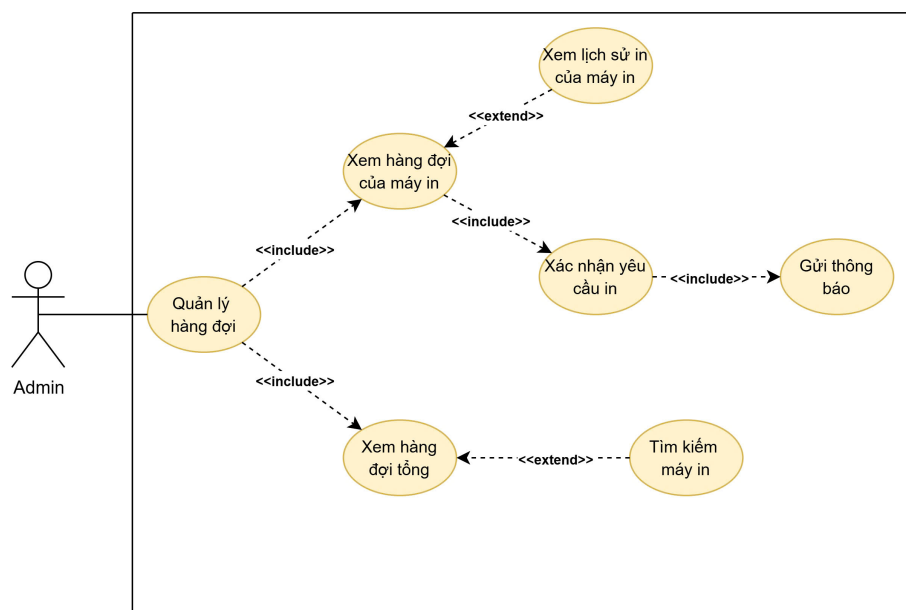
Tên use-case	Thêm máy in mới
Actor	Admin
Description	Admin thêm một máy in mới vào hệ thống
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin đang ở trang Quản lý máy in tổng.
Postcondition	Hệ thống thêm một máy in mới vào hệ thống thành công
Trigger	Admin nhấn nút “Thêm máy in”
Basic Flows	1. Hệ thống hiển thị form mới để admin nhập thông tin máy in mới. 2. Admin điền thông tin và nhấn nút “Thêm”. 3. Hệ thống hiển thị thông báo “Thêm máy in mới thành công” và quay về trang Quản lý máy in tổng.
Alternative Flows	2.1. Admin nhấn nút “Đặt lại”. 2.2. Hệ thống reset lại form với trạng thái ban đầu.
Exception Flows	2E. Hệ thống báo lỗi “Vui lòng điền đủ thông tin” nếu admin điền thiếu thông tin. 2F. Hệ thống báo lỗi “Sai định dạng, vui lòng nhập lại” nếu admin nhập sai định dạng. 3E. Hệ thống báo lỗi “Bạn không thể thêm máy in lúc này” nếu xảy ra lỗi không thể thêm máy in.

Bảng 22: Mô tả use case - Xem thông tin chi tiết từng máy in

Tên use-case	Xem thông tin chi tiết từng máy in
Actor	Admin
Description	Admin xem thông tin chi tiết của từng máy in
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin đang ở trang Quản lý máy in tổng.
Postcondition	Hệ thống hiển thị toàn bộ thông số kỹ thuật và các dữ liệu vận hành khác của máy in
Trigger	1. Admin nhấn vào nút “Thông tin - !” ở máy in muốn xem Hoặc 2. Admin nhấn nút “Chi tiết - !” của Printer’s Info
Basic Flows	1. Hệ thống gửi thông báo “Bạn có chắc muốn xem thông tin máy in này?”. 2. Admin nhấn nút “Xác nhận” và hệ thống điều hướng sang trang xem thông tin chi tiết. 3. Hệ thống hiển thị thông tin chi tiết của máy in bao gồm các thông số kỹ thuật và các dữ liệu vận hành khác như hàng đợi, số lượt in trong ngày,...
Alternative Flows	2.1. Admin nhấn nút “Hủy” và không có gì xảy ra. 3.1. Admin nhấn nút “Chỉnh sửa”. 3.2. Admin nhấn nút “Chi tiết - !” của Lịch sử in.
Exception Flows	2E. Hệ thống báo lỗi “Bạn không thể xem thông tin máy in lúc này!” nếu xảy ra lỗi. 3E. Hệ thống hiển thị “N/A” cho các thông tin bị lỗi.

2.5 Use-case diagram cho Module Quản lí hàng đợi

- Use-case diagram



Hình 2.6: Use-case Quản lí hàng đợi

- Bảng mô tả use-case

Bảng 23: Mô tả use case - Xem hàng đợi từng máy in

Tên use-case	Xem hàng đợi từng máy in
Actor	Admin
Description	Admin xem hàng đợi của từng máy in
Precondition	Admin đang ở trang hàng đợi tổng
Postcondition	Hệ thống hiển thị danh sách các yêu cầu được gửi tới máy in
Trigger	Admin nhấn vào nút “Thông tin” ở máy in muốn xem
Basic Flows	1. Admin nhấn vào nút “Thông tin” ở máy in được chọn. 2. Hệ thống hiển thị thông tin máy in và danh sách hàng đợi (các yêu cầu được gửi tới) máy in đó.
Alternative Flows	None
Exception Flows	1. Ở bước 2, nếu máy in không có hàng đợi (yêu cầu) nào, hệ thống hiển thị thông tin máy in và thông báo “Máy in không có yêu cầu in nào!”. 2. Ở bước 2, nếu máy in có trạng thái “unable”, hệ thống hiển thị thông tin máy in và thông báo “Máy in hiện tại không khả dụng!”.

Bảng 24: Mô tả use case - Quản lý hàng đợi

Tên use-case	Quản lý hàng đợi
Actor	Admin
Description	Admin quản lý hàng đợi in của các máy in trong hệ thống.
Precondition	1. Admin đã đăng nhập vào hệ thống.
Postcondition	Hệ thống hiển thị thông báo xác nhận in thành công và gửi thông báo thành công đến người yêu cầu in.
Trigger	Admin nhấn vào nút " Quản lý hàng đợi " trên thanh điều hướng.
Basic Flows	1. Admin nhấn vào nút quản lý hàng đợi trên thanh điều hướng (navbar). 2. Hệ thống hiển thị danh sách các máy in và số hàng đợi (số yêu cầu in) của từng máy in đó. 3. Admin nhấn vào biểu tượng "i" ở máy in để xem thông tin hàng đợi máy in được chọn. 4. Hệ thống hiển thị thông tin máy in và hàng đợi (danh sách các yêu cầu in) được gửi tới máy in được chọn. 5. Admin bấm nút "In" để xác nhận yêu cầu in và gửi thông báo đến người yêu cầu in. 6. Hệ thống thông báo xác nhận in thành công và gửi thông báo thành công.
Alternative Flows	2.1. Admin tìm kiếm máy in bằng (tên máy in/vị trí máy in). 2.2. Hệ thống tự động lọc danh sách máy in và hiển thị ra các máy in khớp với tìm kiếm. 5.1. Admin bấm nút "Chi tiết" ở phần thông tin máy in. 5.2. Hệ thống hiển thị lịch sử in của máy in đó.
Exception Flows	4.1. Nếu máy in hiện không có hàng đợi (yêu cầu in được gửi tới), hệ thống hiển thị "Không có tài liệu đang đợi in!". 4.2. Nếu máy in có trạng thái "unable", hệ thống hiển thị "Máy in hiện không khả dụng!".

Bảng 25: Mô tả use case - Xác nhận yêu cầu in

Tên use-case	Xác nhận yêu cầu in
Actor	Admin
Description	Admin xác nhận yêu cầu in
Precondition	Admin đang ở trang xem hàng đợi từng máy in
Postcondition	Hệ thống thông báo in thành công và yêu cầu in không còn hiển thị nữa
Trigger	Admin nhấn vào nút "In" ở yêu cầu in muốn in
Basic Flows	1. Admin nhấn vào nút "In" ở yêu cầu in muốn in. 2. Hệ thống hiển thị thông báo in thành công và yêu cầu in không còn hiển thị nữa. 3. Hệ thống thông báo in thành công cho học sinh.
Alternative Flows	None
Exception Flows	1. Ở bước 2, nếu máy in không đủ giấy, hệ thống hiển thị thông báo in thất bại vì không đủ giấy, yêu cầu in không thay đổi.

Bảng 26: Mô tả use case - Xem lịch sử in của máy in

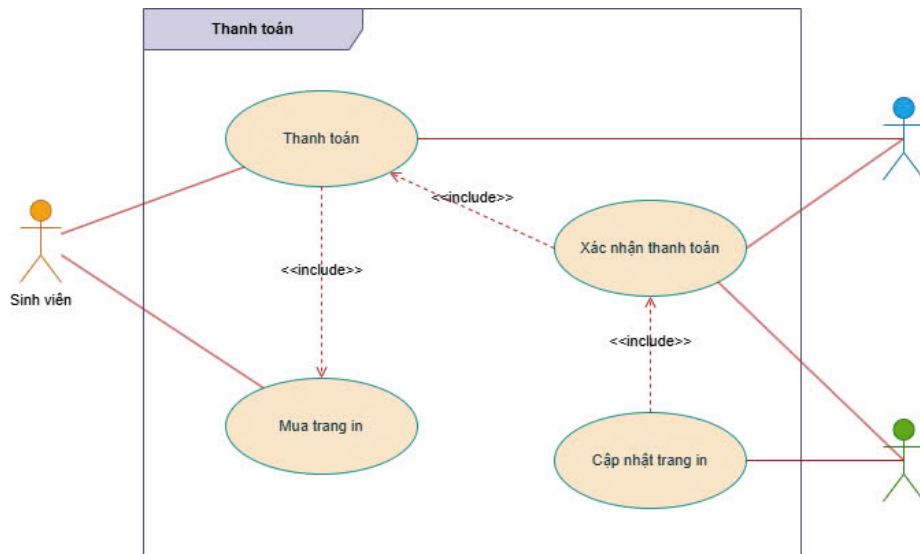
Tên use-case	Xem lịch sử in của máy in
Actor	Admin
Description	Admin xem lịch sử in của máy in
Precondition	1. Đã đăng nhập với role là admin, spso. 2. Admin đang ở trang “Thông tin chi tiết của máy in”.
Postcondition	Hệ thống hiển thị toàn bộ lịch sử in của máy in.
Trigger	Admin nhấn nút “Chi tiết – !” của Lịch sử in.
Basic Flows	1. Hệ thống gửi thông báo “Bạn có chắc muốn xem lịch sử in của máy in này không?”. 2. Admin nhấn nút “Xác nhận” và hệ thống điều hướng sang trang xem lịch sử máy in. 3. Hệ thống hiển thị thông tin cơ sở của máy in và lịch sử in của máy in.
Alternative Flows	1. Admin nhấn nút “Hủy” và không có gì xảy ra. 2. Admin nhấn nút “Xóa lịch sử in”. 3. Admin nhấn nút “Chi tiết – !” của Printer’s Info.
Exception Flows	1. Hệ thống báo lỗi “Bạn không thể xem thông tin máy in lúc này!” nếu xảy ra lỗi. 2. Hệ thống hiển thị “N/A” cho các thông tin bị lỗi.

Bảng 27: Mô tả use case - Xem hàng đợi tổng

Tên use-case	Xem hàng đợi tổng
Actor	Admin
Description	Admin xem hàng đợi của tất cả máy in
Precondition	Admin đã đăng nhập
Postcondition	Hệ thống hiển thị số hàng đợi của tất cả máy in
Trigger	Admin nhấn vào nút “Quản lý hàng đợi” trên thanh điều hướng
Basic Flows	1.Admin nhấn vào nút quản lý hàng đợi trên thanh điều hướng (navbar). Hệ thống hiển 2.thị danh sách các máy in và số hàng đợi (số yêu cầu in) của từng máy in đó.
Alternative Flows	Tìm kiếm máy in bằng (tên máy in/ vị trí máy in) sau bước 2.
Exception Flows	None

2.6 Use-case diagram cho Module Mua và Thanh Toán

- Use-case diagram



Hình 2.7: Use-case mua và thanh toán

- Bảng mô tả use-case

Bảng 28: Mô tả use case - Mua và thanh toán

Tên use-case	Mua và thanh toán
Actor	Sinh viên, PayOs, System
Description	Sinh viên mua và thanh toán trang in đã mua.
Precondition	Người dùng có tài khoản và đã đăng nhập vào dịch vụ in
Postcondition	Số trang in được cập nhật trong tài khoản của Học sinh.
Trigger	Học sinh chọn mục “Mua trang in”
Basic Flows	<ol style="list-style-type: none"> 1. Người dùng chọn “Mua thêm trang in”. 2. Người dùng nhập số lượng trang in cần mua và ấn vào nút thanh toán. 3. Hệ thống tính toán số tiền cần thanh toán và điều hướng người dùng đến link thanh toán qua PayOs. 4. Người dùng tiến hành quét mã QR và thanh toán số tiền cần thiết. 5. Hệ thống xác nhận thanh toán và trả người dùng về trang trước đó.
Alternative Flows	Không.
Exception Flows	<p>Trước bước 4, khi quá trình bị gián đoạn, hủy toàn bộ quá trình và bắt đầu lại từ đầu.</p> <p>Tại bước 4, nếu quá trình bị gián đoạn hoặc người dùng chọn hủy thanh toán, hệ thống thông báo “Thanh toán không thành công” và hủy toàn bộ quá trình.</p>

3 Thiết kế giao diện

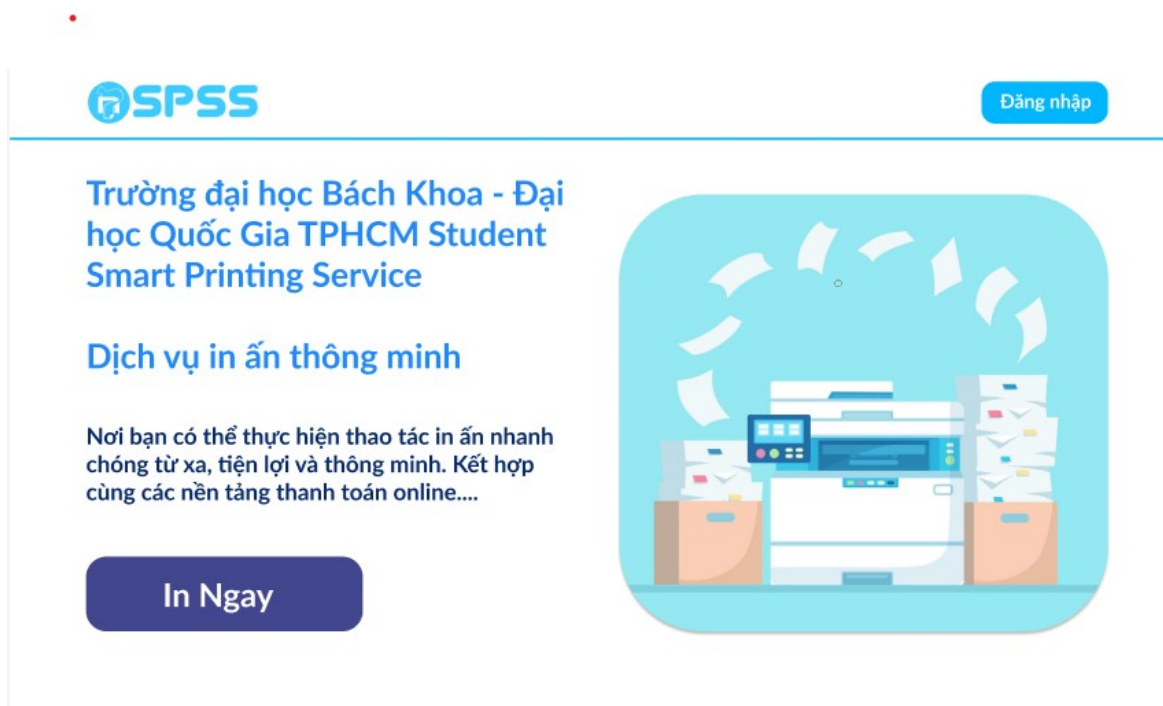
Phác thảo giao diện người dùng của Module quản lý in bằng Figma

3.1 Liên kết

- Liên kết đến Prototype (Figma) [tại đây](#).
- Liên kết đến Wireframe, Mockup (Figma) [tại đây](#).

3.2 Mô tả

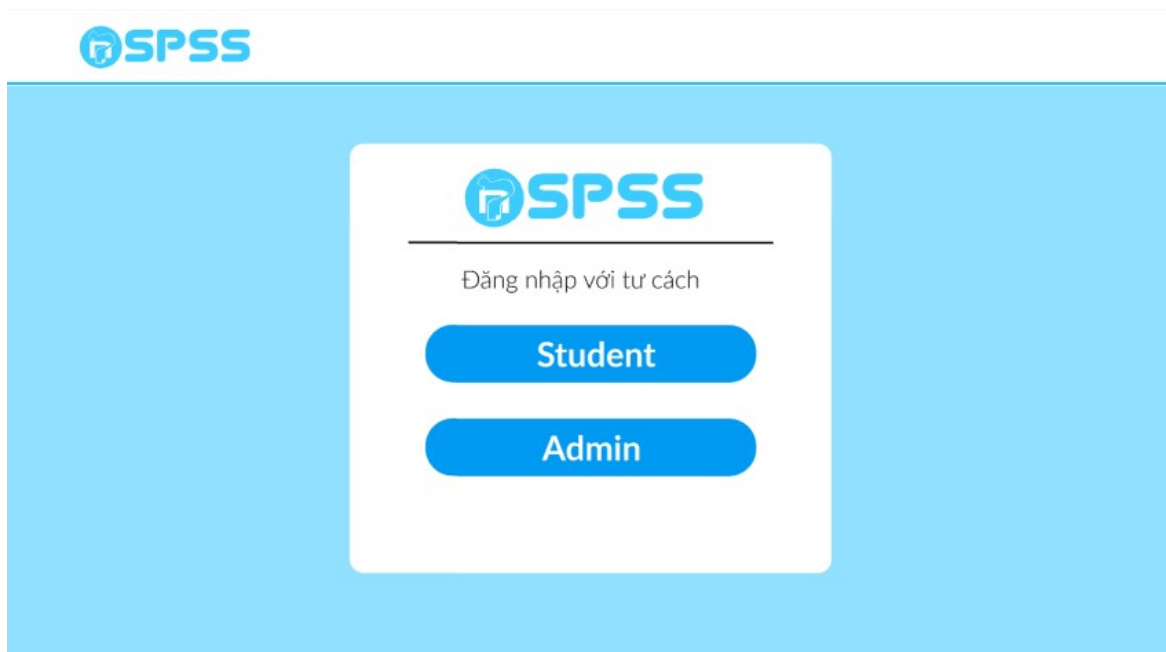
- **Giao diện trang chủ trước khi đăng nhập**
Khi vừa vào trang chính người dùng sẽ thấy giao diện như hình bên dưới:



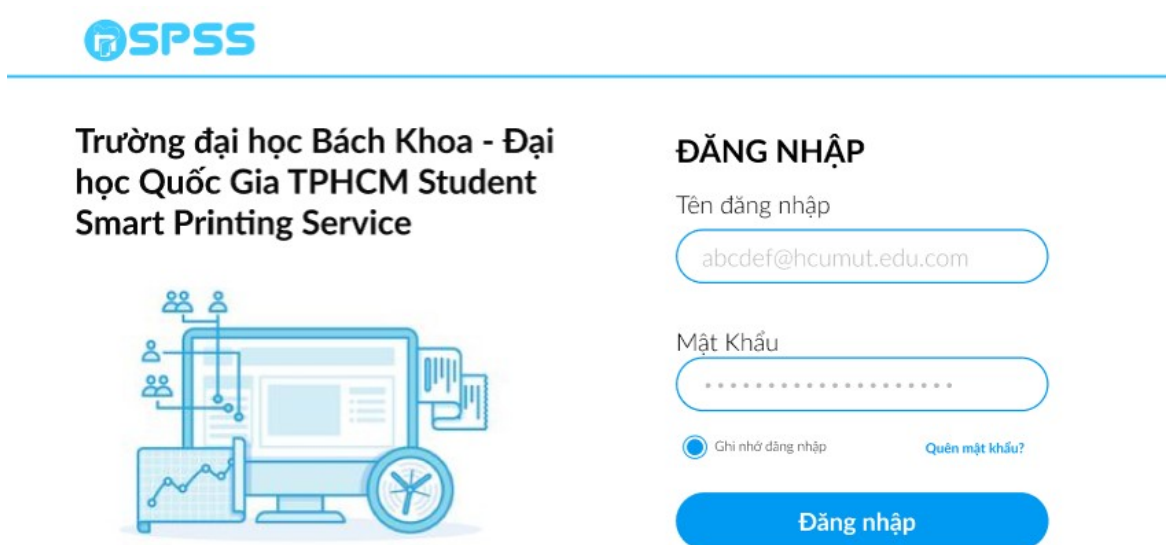
Hình 3.8: Giao diện trang chủ

- **Giao diện đăng nhập**
Nếu chưa đăng nhập vào ứng dụng thì khi người dùng nhấn vào bất kì nút nào trong các nút:
 - In ngay.
 - Đăng nhập.

trang cũng sẽ điều hướng người dùng tới giao diện đăng nhập như hình bên dưới:



Hình 3.9: Giao diện chọn tư cách đăng nhập



Hình 3.10: Giao diện đăng nhập

- **Giao diện trang chủ sau khi đăng nhập**

Sau khi đăng nhập, giao diện của người dùng là sinh viên sẽ như hình bên dưới:

The screenshot shows the SPSS printing interface. On the left is a 'File Upload' section with a dashed box and a 'Browse File to Upload' button. On the right is the 'Thông số in' (Print Settings) section, which is divided into two tabs: '1. Tải file lên' (Upload file) and '2. Chọn máy in' (Select printer). The '1. Tải file lên' tab is active. It contains the following settings:

- Hướng in** (Print orientation): Dropdown menu set to 'Dọc (Portrait)'.
- Chọn trang** (Select page): Input field with 'Ví dụ : 1-10' (Example: 1-10).
- Số bản** (Number of copies): Input field set to '1'.
- Kiểu in** (Print type): Dropdown menu set to '1 mặt' (1 side).
- Khổ giấy** (Paper size): Dropdown menu set to 'A4'.

At the bottom right of the 'Thông số in' section is a blue button labeled 'Tiếp theo >' (Next >).

Hình 3.11: Giao diện trang chủ

- **Giao diện đăng kí in**

Sau khi đăng nhập thành công, Người dùng tiến hành tải tệp và điều chỉnh thông số in phù hợp.

The screenshot shows the SPSS printing interface after a user has logged in. The 'File Upload' section on the left now displays a list of four files: 'BTL_MMT.pdf', 'BTL_DSA.pdf', 'MHHTH.pdf', and 'GT1.pdf'. Each file has a red 'x' icon next to it. Below the list is an 'UpLoad File' button. The 'Thông số in' (Print Settings) section on the right is still active, but the settings have been adjusted:

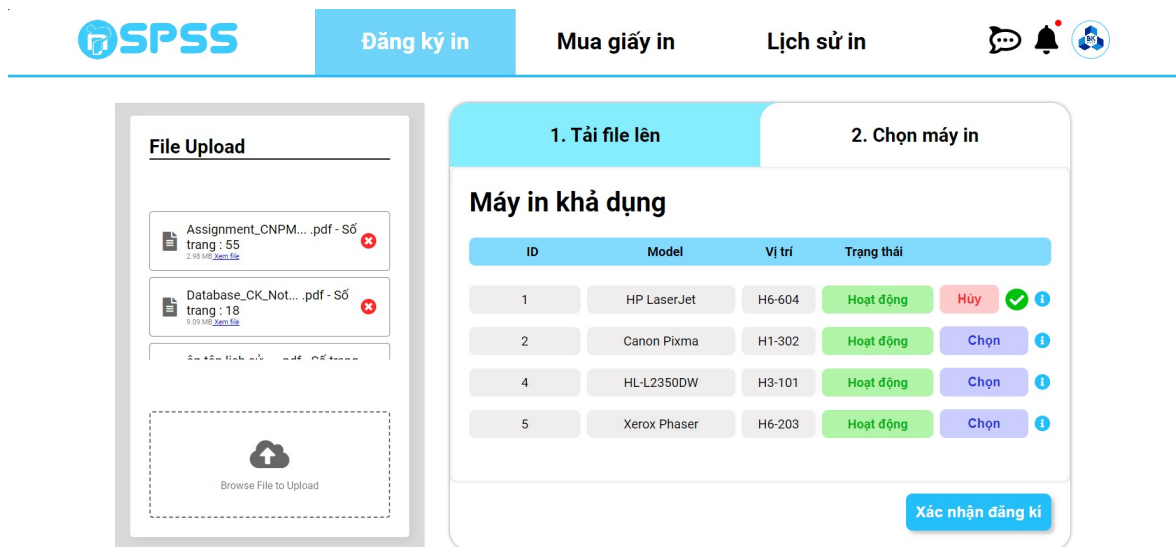
- Hướng in** (Print orientation): Dropdown menu set to 'Dọc' (Portrait).
- Chọn trang** (Select page): Input field set to '1-33'.
- Số bản** (Number of copies): Input field set to '5'.
- Kiểu in** (Print type): Dropdown menu set to '2 mặt' (2 sides).
- Khổ giấy** (Paper size): Input field set to 'A4'.

At the bottom right of the 'Thông số in' section is a button labeled 'Next >'.

Hình 3.12: Giao diện tải tệp và điều chỉnh thông số in

- **Giao diện chọn máy in**

Sau khi up tệp và điều chỉnh thông số in phù hợp với nhu cầu. Người dùng sẽ nhấn vào **2.Chọn máy in** để điều hướng trang tới giao diện chọn máy in.



Hình 3.13: Giao diện chọn máy in

- **Giao diện xem hàng đợi của máy in trước khi xác nhận in**

Sau khi up tệp và điều chỉnh thông số in phù hợp với nhu cầu. Người dùng sẽ nhấn vào để điều hướng trang tới giao diện xem hàng đợi của máy in để lựa chọn máy in có hàng đợi mong muốn.



Hình 3.14: Giao diện xem hàng đợi của máy in

Sau khi xem lại người dùng nhấn vào nút < để quay về giao diện để gửi yêu cầu về cho máy in và chờ xác nhận từ hệ thống

- **Giao diện xem lịch sử in**

Tại phần *header* giao diện trang chủ, khi người dùng nhấn vào nút *Lịch sử in* trang sẽ điều hướng tới giao diện xem lịch sử in

[Đăng ký in](#)
[Mua giấy in](#)
[Lịch sử in](#)

Lịch Sử In Cá Nhân

Tên: vta
ID: 7
Số dư trang in: 575
Email: std003@gmail.com
Trạng thái: Hoạt động

Máy in	Thời gian in	Trang/Khổ giấy	File In	Tình trạng
Printer B2 H1-302	3/1/2025 01:14:49	5 (x1) A4	Database_CK_Not... .pdf	Đang đợi in
Printer A1 H6-604	3/1/2025 01:14:16	10 (x1) A4	Đồ án CNPM (DEM... .pdf)	In thành công

Hình 3.15: Giao diện xem lịch sử in

• Giao diện mua giấy in

Tại phần *header* giao diện trang chủ, khi người dùng nhấn vào nút *Mua giấy in* trang sẽ điều hướng tới giao diện mua.

[Đăng ký in](#)
[Mua giấy in](#)
[Lịch sử in](#)

Thông tin

- Trang in mà bạn mua sẽ mặc định là A4, khi bạn tiến hành in tài liệu, 1 trang A3 sẽ dùng 2 trang in A4 mà bạn có và tương tự cho các khổ giấy khác.
- Giá mỗi trang in là 250 vnd, vui lòng xem kỹ trước khi ấn thanh toán.

Chọn số lượng

Vui lòng nhập số lượng bạn muốn mua
Lưu ý: trang in sẽ mặc định là A4

-

Số lượng: 100

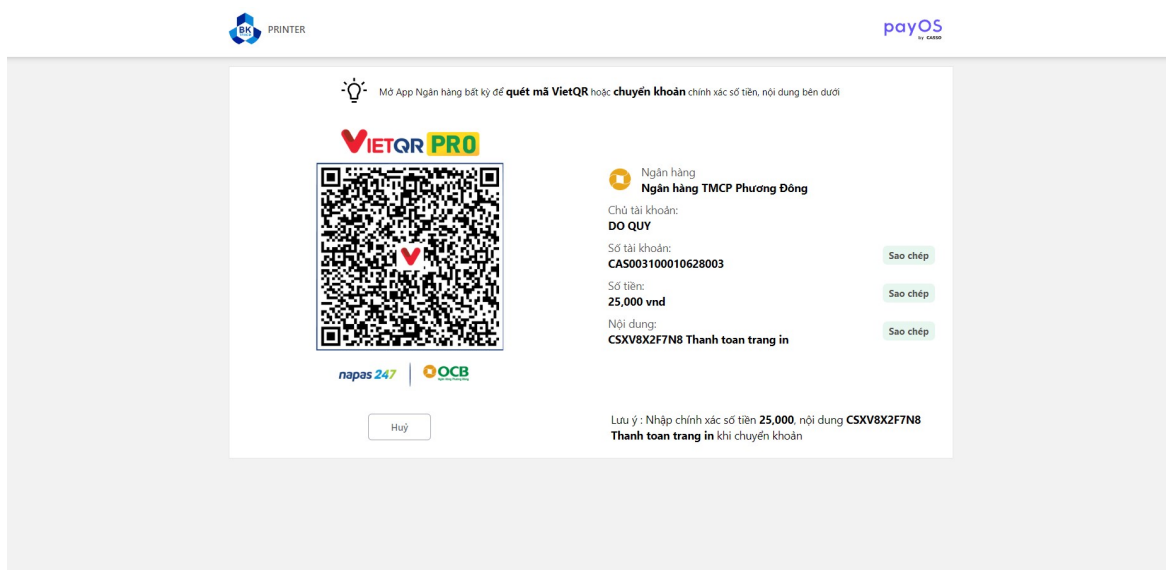
+

Thanh toán

Hình 3.16: Giao diện mua giấy in

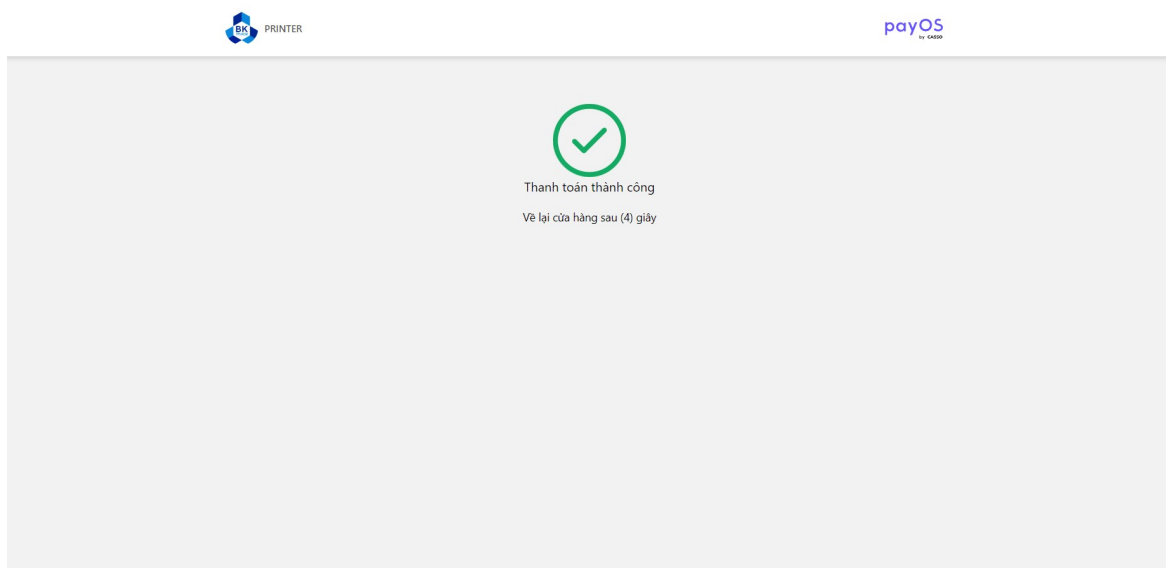
• Giao diện lịch sử mua trang và thanh toán

Sau khi chọn được gói giấy mà mình mong muốn, người dùng nhấn vào nút *Thanh toán* trang sẽ được điều hướng tới màn hình thanh toán.



Hình 3.17: Giao diện thanh toán

Hệ thống sẽ hiện thông báo khi người dùng thanh toán thành công cho số trang đã thêm vào giỏ.



Hình 3.18: Thông báo thanh toán thành công

4 Lựa chọn công nghệ

4.1 Front-End

4.1.1 HTML (Hyper Text Markup Language)

HTML là ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và nội dung cho các trang web và được dùng để:

- Xây dựng cấu trúc trang web: HTML tổ chức các thành phần như tiêu đề, đoạn văn, hình ảnh, liên kết, biểu mẫu, bảng biểu. . .
- Hiển thị nội dung: Mọi nội dung mà bạn thấy trên một trang web (văn bản, hình ảnh, video, âm thanh) đều được định nghĩa bằng HTML.

Ưu điểm của HTML: Ngôn ngữ đơn giản và dễ tiếp cận, được hỗ trợ bởi tất cả các trình duyệt web hiện đại và là nền tảng cho các công nghệ web khác như CSS và JavaScript.

Hạn chế của HTML: HTML chỉ định nghĩa nội dung và cấu trúc, không thể xử lý logic hoặc thêm tính năng động và cần CSS để định dạng giao diện.

Lý do chọn HTML làm công nghệ thiết kế dự án:

- Tương thích cao với các trình duyệt: HTML là ngôn ngữ chuẩn của web, được hỗ trợ bởi tất cả các trình duyệt. Điều này đảm bảo người dùng có thể truy cập hệ thống in ấn thông qua bất kỳ thiết bị có trình duyệt nào bao gồm cả máy tính, điện thoại và máy tính bảng.
- Dễ dàng thiết kế giao diện: HTML kết hợp với CSS và JavaScript giúp thiết kế giao diện người dùng thân thiện, tùy biến và dễ sử dụng, phù hợp với nhu cầu của sinh viên và quản trị viên.
- Khả năng tích hợp tốt: HTML cho phép tích hợp hệ thống in ấn với các công nghệ khác như RESTful API, PayOS cho thanh toán hoặc các cơ sở dữ liệu để quản lý thông tin người dùng và máy in.
- Khả năng phản hồi tốt (Responsive): HTML cho phép tạo giao diện linh hoạt, tự động điều chỉnh trên các kích thước màn hình khác nhau, giúp hệ thống dễ dàng truy cập trên mọi thiết bị.
- Dễ dàng bảo trì và mở rộng: HTML là ngôn ngữ đơn giản, dễ học và bảo trì. Hệ thống in ấn của chúng tôi sẽ dễ dàng được nâng cấp hoặc bổ sung tính năng trong tương lai.

4.1.2 CSS (Cascading Style Sheets)

CSS là ngôn ngữ được sử dụng để định kiểu và bố trí giao diện cho các trang web. CSS hoạt động cùng với HTML để làm cho trang web trông đẹp mắt và chuyên nghiệp hơn bằng cách kiểm soát cách các phần tử HTML hiển thị trên trình duyệt. CSS hoạt động dựa trên quy tắc cascade (thác nước), nghĩa là các styles được áp dụng theo thứ tự từ trên xuống dưới, kiểu khai báo sau sẽ ghi đè kiểu khai báo trước nếu có xung đột.

Vai trò của CSS:

- Thay đổi màu sắc, phông chữ, kích thước, khoảng cách, đường viền,.. của các thành phần HTML .
- Sắp xếp các phần tử trên trang, tạo bố cục như lưới (grid), cột (column), hoặc định vị các phần tử (positioning).
- Giúp giao diện trang web tự động điều chỉnh để hiển thị tốt trên nhiều thiết bị khác nhau.

Lý do chọn CSS:

- Tạo giao diện đẹp mắt và linh hoạt: CSS cho phép tạo ra các thiết kế đẹp mắt và dễ dàng tùy chỉnh giao diện người dùng có bố cục linh hoạt và phù hợp với mọi kích thước màn hình.
- Tách biệt cấu trúc và kiểu dáng: Sử dụng CSS giúp tách biệt phần cấu trúc HTML (nội dung) và phần kiểu dáng (thiết kế), giúp mã nguồn dễ đọc, bảo trì và nâng cấp.
- Hiệu suất cao: Việc sử dụng CSS giúp tối ưu hiệu suất của hệ thống, đặc biệt trong việc xử lý giao diện người dùng trên các thiết bị có cấu hình khác nhau.
- Tính tương thích cao: CSS được hỗ trợ rộng rãi trên tất cả các trình duyệt hiện đại, giúp giao diện của hệ thống đảm bảo hoạt động mượt mà trên mọi nền tảng mà không cần lo lắng về vấn đề tương thích.

4.1.3 JS (JavaScript)

JavaScript (JS) là một ngôn ngữ lập trình phổ biến dùng để tạo ra các trang web tương tác động. Khác với HTML (tạo cấu trúc) và CSS (định dạng giao diện), JavaScript giúp thêm các tính năng tương tác và xử lý logic cho trang web.

Vai trò của JavaScript:

- Thêm tính năng động (Dynamic Content): Cập nhật nội dung trang web mà không cần tải lại toàn bộ trang. Hiển thị thông báo, đếm ngược thời gian, hoặc thay đổi giao diện dựa trên hành động của người dùng.
- Tương tác với người dùng: Xử lý các sự kiện như nhấp chuột (click), di chuột (hover) hoặc nhập dữ liệu vào form.
- Xử lý logic: Kiểm tra và xử lý dữ liệu nhập từ người dùng trước khi gửi đến máy chủ. Thực hiện các tính toán ngay trên trình duyệt.
- Kết nối với máy chủ: Gửi và nhận dữ liệu từ máy chủ mà không cần tải lại trang thông qua AJAX hoặc Fetch API.

Lý do sử dụng JavaScript trong dự án:

- Tương tác động (Interactive User Experience): JavaScript cho phép bạn tạo ra các tương tác động và mượt mà trên giao diện người dùng mà không cần tải lại trang. Điều này cực kỳ quan trọng trong một hệ thống như in ấn, nơi người dùng cần thao tác nhanh chóng (chọn máy in, tải lên tệp, thanh toán,...) mà không muốn bị gián đoạn.
- Quản lý sự kiện và phản hồi tức thì: JavaScript giúp dễ dàng xử lý các sự kiện như nhấp chuột, nhập liệu, thay đổi lựa chọn từ người dùng.
- Xử lý dữ liệu và tương tác với API: Trong hệ thống in ấn, JavaScript rất hữu ích khi chúng tôi cần gọi các API để lấy dữ liệu mà không làm gián đoạn trải nghiệm người dùng. Thông qua AJAX (Asynchronous JavaScript and XML) hoặc Fetch API để thực hiện các yêu cầu không đồng bộ, giúp giao diện luôn cập nhật mà không cần tải lại trang.
- Xử lý lỗi và thông báo cho người dùng: JavaScript có thể dễ dàng kiểm tra và xử lý các lỗi trong quá trình người dùng thao tác. Ví dụ như kiểm tra xem người dùng đã tải đủ số trang cần thiết chưa hoặc nếu có lỗi kết nối khi in, có thể thông báo cho người dùng về các lỗi đó một cách dễ hiểu và kịp thời.
- Tạo hiệu ứng và giao diện động: JavaScript hỗ trợ việc tạo ra các hiệu ứng động cho giao diện người dùng. Điều này sẽ cải thiện trải nghiệm người dùng và làm cho hệ thống trở nên dễ sử dụng hơn.
- Tương thích với các nền tảng hiện đại: JavaScript là ngôn ngữ chủ yếu trong phát triển web và được hỗ trợ trên mọi trình duyệt hiện đại, điều này giúp hệ thống hoạt động tốt trên tất cả các thiết bị và nền tảng mà không cần phải lo lắng về vấn đề tương thích.
- Tiết kiệm tài nguyên và tối ưu hóa hiệu suất: JavaScript chỉ cần tải những dữ liệu và phần giao diện cần thiết, thay vì phải tải lại toàn bộ trang. Điều này làm tăng tốc độ phản hồi của hệ thống, đặc biệt quan trọng trong các hệ thống xử lý dữ liệu lớn như in ấn.

4.2 Back-End

4.2.1 NodeJS

Node.js là một môi trường chạy JavaScript trên phía máy chủ. Khác với JavaScript truyền thống chỉ chạy trên trình duyệt, Node.js cho phép sử dụng JavaScript để xây dựng các ứng dụng phía server như API, ứng dụng thời gian thực (real-time), hoặc quản lý cơ sở dữ liệu.

Đặc điểm của Node JS:

- Xử lý không đồng bộ (Asynchronous): Node.js sử dụng Event Loop để xử lý nhiều yêu cầu cùng lúc mà không cần chờ một yêu cầu hoàn thành. Điều này giúp tiết kiệm tài nguyên và tăng hiệu suất.
- Sử dụng chung JavaScript: Node.js cho phép sử dụng JavaScript cho cả front-end và back-end, giúp đồng bộ hóa ngôn ngữ trong toàn bộ dự án.
- Hiệu suất cao: Nhờ vào V8 Engine, Node.js có tốc độ thực thi nhanh và tối ưu hóa tốt cho các ứng

dụng thời gian thực.

-Hệ sinh thái lớn: Node.js có một kho thư viện khổng lồ gọi là npm (Node Package Manager), cung cấp hàng triệu thư viện miễn phí để phát triển ứng dụng.

Lý do sử dụng Node JS trong dự án:

-Xử lý yêu cầu bất đồng bộ (Asynchronous): Node.js có thể xử lý các yêu cầu bất đồng bộ hiệu quả, đặc biệt hữu ích trong các hệ thống yêu cầu nhiều tác vụ như tải tệp, kiểm tra trạng thái máy in,... mà không làm gián đoạn trải nghiệm người dùng. Hệ thống in ấn có thể xử lý đồng thời nhiều yêu cầu từ người dùng mà không bị chậm trễ.

-Tạo API và giao tiếp với các dịch vụ bên ngoài: Node.js có thể giúp bạn xây dựng các API RESTful hoặc GraphQL cho phép hệ thống giao tiếp với các dịch vụ khác, lấy thông tin và trả kết quả nhanh chóng cho người dùng.

-Quản lý hàng đợi in: Node.js có thể xử lý các tác vụ liên quan đến quản lý hàng đợi in, việc này rất quan trọng để đảm bảo các công việc in được thực hiện chính xác và đúng theo thứ tự yêu cầu.

-Xử lý dữ liệu và lưu trữ: Node.js có thể giao tiếp với cơ sở dữ liệu (SQL) để lưu trữ các thông tin về tài khoản người dùng, lịch sử in.... Node.js có thể sử dụng các thư viện như Sequelize (cho SQL) để truy xuất, thêm, sửa và xóa dữ liệu từ cơ sở dữ liệu.

-Xử lý thanh toán và giao tiếp với PayOs: Hệ thống in ấn có thể tích hợp với hệ thống thanh toán như PayOs để xử lý việc mua trang in. Điều này cho phép người dùng thanh toán trực tuyến nhanh chóng và dễ dàng.

-Tích hợp giao diện người dùng (UI) và backend: Node.js có thể làm cầu nối giữa giao diện người dùng (frontend) và hệ thống backend. Với việc sử dụng Express.js (framework của Node.js), có thể xử lý các yêu cầu HTTP từ phía người dùng và trả kết quả một cách nhanh chóng, hiệu quả.

-Hiển thị thông báo và cập nhật thời gian thực: Với sự hỗ trợ của các thư viện như Socket.io, Node.js có thể cung cấp các thông báo và cập nhật thời gian thực. Ví dụ, khi trạng thái in của tài liệu thay đổi hoặc khi có lỗi trong quá trình in, hệ thống có thể gửi thông báo trực tiếp đến người dùng mà không cần phải tải lại trang.

-Tối ưu hóa hiệu suất và phân phối tải: Node.js có thể xử lý nhiều kết nối đồng thời mà không gây tắc nghẽn. Điều này rất quan trọng trong các hệ thống in ấn với nhiều người dùng truy cập đồng thời.

-Chạy trên nhiều nền tảng: Node.js có thể chạy trên nhiều hệ điều hành khác nhau như Windows, Linux và macOS. Điều này giúp hệ thống in có thể triển khai linh hoạt trên các máy chủ khác nhau mà không gặp phải vấn đề tương thích.

4.2.2 Express.js

Express.js là một framework web nhẹ, nhanh và linh hoạt được xây dựng trên nền tảng Node.js. Nó giúp đơn giản hóa việc phát triển các ứng dụng web và API bằng cách cung cấp các công cụ và tính năng mạnh mẽ để xử lý yêu cầu HTTP, định tuyến, và quản lý middleware.

Đặc điểm nổi bật của Express.js:

-Quản lý Routing: Hỗ trợ định nghĩa các đường dẫn (route) dễ dàng để xử lý các yêu cầu HTTP như GET, POST, PUT, DELETE.

-Middleware mạnh mẽ: Middleware trong Express.js giúp xử lý yêu cầu (request), phản hồi (response) hoặc thực hiện các tác vụ trung gian như xác thực hay nén dữ liệu.

-Hỗ trợ View Engine: Tích hợp dễ dàng với các engine hiển thị (view engines) như EJS, Pug, hoặc Handlebars để xây dựng giao diện động.

-Tích hợp tốt với Node.js: Sử dụng toàn bộ điểm mạnh của Node.js và cộng đồng npm để mở rộng tính năng.

Lý do sử dụng Express.js trong dự án:

-Tạo API nhanh chóng và dễ dàng: Express.js giúp xây dựng các API RESTful một cách đơn giản và nhanh chóng, cung cấp các tính năng như routing, middleware, và quản lý các phương thức HTTP (GET, POST, PUT, DELETE), giúp xử lý các yêu cầu của người dùng dễ dàng và hiệu quả.

-Middleware và quản lý yêu cầu linh hoạt: Trong hệ thống in ấn, có thể dễ dàng sử dụng middleware

để kiểm tra các yêu cầu (ví dụ: kiểm tra xem người dùng có đủ số trang in để in tài liệu hay không) hoặc để xử lý lỗi khi các máy in gặp sự cố.

-Khả năng mở rộng dễ dàng: Express.js là một framework rất linh hoạt và nhẹ, giúp bạn dễ dàng mở rộng và thêm các tính năng mới khi cần.

-Tích hợp dễ dàng với cơ sở dữ liệu: Express.js giúp dễ dàng kết nối và quản lý các cơ sở dữ liệu như MongoDB, MySQL, PostgreSQL, hoặc các cơ sở dữ liệu khác.

-Tối ưu hóa hiệu suất: Express.js rất nhẹ và tốc độ nhanh nên sẽ không làm chậm hệ thống khi xử lý hàng loạt yêu cầu từ người dùng.

-Quản lý phiên người dùng và bảo mật: Với Express.js, có thể dễ dàng tích hợp các giải pháp quản lý phiên người dùng (như JWT, Passport.js), giúp bảo mật hệ thống in ấn và xác thực người dùng một cách hiệu quả.

4.2.3 PayOS

PayOS là một nền tảng tích hợp thanh toán giúp các doanh nghiệp kết nối và quản lý nhiều phương thức thanh toán từ các nhà cung cấp dịch vụ khác nhau. PayOS tập trung vào việc đơn giản hóa quy trình thanh toán trực tuyến, hỗ trợ nhiều loại hình thanh toán như thẻ tín dụng, ví điện tử, ngân hàng trực tuyến, và các phương thức thanh toán địa phương khác.

Lý do sử dụng PayOs trong dự án:

- Hỗ trợ thanh toán đa dạng: PayOs hỗ trợ nhiều phương thức thanh toán như thẻ tín dụng, thẻ ghi nợ, ví điện tử, và mã QR. Điều này giúp người dùng dễ dàng thanh toán theo cách họ mong muốn, cải thiện trải nghiệm người dùng. Tích hợp đa dạng giúp hệ thống của bạn phục vụ nhiều nhóm người dùng khác nhau, đặc biệt là sinh viên với nhiều cách thanh toán linh hoạt.

- Tích hợp dễ dàng: PayOs cung cấp API và SDK thân thiện, dễ tích hợp với các nền tảng như Node.js (kèm theo Express.js), điều này giúp giảm thời gian phát triển và triển khai hệ thống thanh toán.

- Bảo mật cao: PayOs tuân thủ các tiêu chuẩn bảo mật như PCI DSS (Payment Card Industry Data Security Standard), đảm bảo thông tin thanh toán của người dùng được bảo vệ.

- Tích hợp xác nhận giao dịch tự động: PayOs cung cấp tính năng webhook để tự động thông báo trạng thái giao dịch (thành công, thất bại) cho hệ thống. Điều này rất hữu ích để tự động cập nhật số trang in cho sinh viên sau khi họ thanh toán thành công.

- Theo dõi và báo cáo giao dịch: PayOs cho phép theo dõi chi tiết các giao dịch trực tiếp trên dashboard của họ, cung cấp báo cáo giao dịch theo thời gian thực.

- Độ tin cậy cao: PayOs được thiết kế để xử lý khối lượng giao dịch lớn mà vẫn đảm bảo tốc độ và độ ổn định.

- Chi phí cạnh tranh: PayOs thường cung cấp mức phí giao dịch cạnh tranh, giúp tối ưu hóa chi phí khi triển khai thanh toán trong hệ thống. Phí dịch vụ rõ ràng, minh bạch, phù hợp với một dự án có ngân sách giới hạn như hệ thống in ấn cho trường học.

- Phù hợp với nhu cầu hệ thống in ấn: Hệ thống in ấn thông minh cần một nền tảng thanh toán nhanh, an toàn và dễ quản lý. PayOs đáp ứng đầy đủ các yêu cầu này, từ việc thanh toán nhanh qua mã QR cho đến quản lý trạng thái giao dịch một cách tự động.

4.3 Cơ sở dữ liệu MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở, được phát triển bởi Oracle Corporation. Nó sử dụng ngôn ngữ SQL (Structured Query Language) để quản lý và truy xuất dữ liệu. MySQL là một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất, được sử dụng rộng rãi trong các ứng dụng web, hệ thống thương mại điện tử và lưu trữ dữ liệu doanh nghiệp.

Lý do sử dụng MySQL trong dự án:

- Hiệu suất cao: MySQL được tối ưu hóa để xử lý các thao tác đọc/ghi dữ liệu nhanh, phù hợp với các ứng dụng cần truy vấn nhanh như quản lý dữ liệu người dùng, đơn hàng, lịch sử in, và hàng đợi in trong hệ thống in ấn.

- Dễ tích hợp: MySQL dễ dàng tích hợp với các ngôn ngữ lập trình phổ biến như Node.js nên việc triển khai các tính năng trong hệ thống in ấn trở nên dễ dàng.

- Khả năng mở rộng: MySQL hỗ trợ cả các hệ thống nhỏ lẫn hệ thống lớn. Do đó, trong tương lai nếu có nhu cầu mở rộng thì không cần thay đổi nhiều về kiến trúc hệ thống.
- Hỗ trợ đa người dùng: MySQL có khả năng xử lý tốt các tác vụ từ nhiều người dùng cùng lúc, điều này rất quan trọng với hệ thống in ấn khi có hàng trăm sinh viên thực hiện thao tác tải file, mua trang in, hoặc kiểm tra lịch sử in cùng thời điểm.
- Bảo mật tốt: MySQL cung cấp các tính năng bảo mật mạnh mẽ như Authentication plugins (xác thực người dùng), Encryption (mã hóa dữ liệu trong quá trình truyền tải), Access control (kiểm soát quyền truy cập theo vai trò: sinh viên, quản trị viên).
- Khả năng xử lý dữ liệu quan hệ: Hệ thống in ấn thông minh có nhiều thực thể dữ liệu liên kết với nhau như sinh viên, máy in, đơn hàng,... và MySQL có thể hỗ trợ tốt các truy vấn dữ liệu phức tạp dựa trên các mối quan hệ giữa bảng.
- Hỗ trợ sao lưu và phục hồi dữ liệu: MySQL cung cấp các công cụ sao lưu và phục hồi dữ liệu mạnh mẽ, giúp bảo vệ dữ liệu của hệ thống trước các lỗi không mong muốn, chẳng hạn như lỗi phần cứng hoặc mất kết nối mạng.

4.4 Công cụ hỗ trợ

4.4.1 Docker

Docker Desktop là một nền tảng phát triển ứng dụng container hóa, cung cấp môi trường để xây dựng, thử nghiệm và triển khai ứng dụng. Docker Desktop đơn giản hóa quy trình tạo và quản lý container, giúp các nhà phát triển làm việc với Docker dễ dàng hơn.

Lý do sử dụng Docker Desktop trong dự án:

- Môi trường nhất quán: Docker Desktop cung cấp môi trường phát triển nhất quán, giúp loại bỏ các vấn đề liên quan đến sự khác biệt về môi trường giữa máy phát triển và máy chủ.
- Tích hợp với CI/CD: Docker Desktop dễ dàng tích hợp với các hệ thống CI/CD (Continuous Integration/Continuous Deployment), giúp tự động hóa quy trình build, test và deploy ứng dụng.
- Hỗ trợ nhiều hệ điều hành: Docker Desktop có thể chạy trên nhiều hệ điều hành như Windows, macOS và Linux, giúp linh hoạt trong việc triển khai môi trường phát triển.
- Quản lý dễ dàng: Docker Desktop cung cấp giao diện người dùng thân thiện, giúp quản lý và giám sát các container, images và volumes dễ dàng.
- Hiệu suất cao: Docker Desktop được tối ưu hóa để chạy các container với hiệu suất cao, đảm bảo ứng dụng hoạt động mượt mà và ổn định.

4.4.2 Postman

Postman là một công cụ API hỗ trợ các nhà phát triển thiết kế, thử nghiệm và tài liệu hóa các API một cách dễ dàng. Postman cung cấp giao diện trực quan và nhiều tính năng hữu ích để làm việc với API.

Lý do sử dụng Postman trong dự án:

- Thiết kế và thử nghiệm API: Postman cho phép thiết kế, gửi yêu cầu và kiểm tra kết quả API nhanh chóng, giúp phát hiện và sửa lỗi kịp thời.
- Tài liệu hóa API: Postman cung cấp các công cụ để tài liệu hóa API, giúp chia sẻ và duy trì tài liệu API một cách hiệu quả.
- Tích hợp với CI/CD: Postman hỗ trợ tích hợp với các hệ thống CI/CD, giúp tự động hóa quy trình kiểm tra API trong quá trình phát triển.
- Quản lý collection: Postman cho phép tạo và quản lý các collection, giúp tổ chức các yêu cầu API một cách có hệ thống.
- Hỗ trợ nhiều định dạng dữ liệu: Postman hỗ trợ nhiều định dạng dữ liệu như JSON, XML, và HTML, giúp làm việc với nhiều loại API khác nhau.

4.4.3 Figma

Figma là một công cụ thiết kế giao diện người dùng (UI) trực tuyến, hỗ trợ cộng tác nhóm và tạo prototype một cách hiệu quả. Figma cung cấp môi trường làm việc trực quan và nhiều tính năng mạnh mẽ để thiết kế UI/UX.

Lý do sử dụng Figma trong dự án:

- Cộng tác thời gian thực: Figma cho phép nhiều người dùng cùng làm việc trên một thiết kế trong thời gian thực, giúp tăng hiệu quả làm việc nhóm.
- Tạo prototype: Figma cung cấp công cụ để tạo prototype và kiểm tra trải nghiệm người dùng, giúp phát hiện và cải thiện các vấn đề về giao diện.
- Tích hợp với các công cụ khác: Figma dễ dàng tích hợp với các công cụ khác như Slack, Zeplin, và các hệ thống quản lý dự án, giúp tối ưu hóa quy trình làm việc.
- Thiết kế linh hoạt: Figma hỗ trợ nhiều công cụ thiết kế mạnh mẽ, giúp tạo ra các giao diện đẹp và chuyên nghiệp.
- Quản lý version: Figma cung cấp tính năng quản lý version, giúp theo dõi và phục hồi các phiên bản thiết kế trước đó.

4.5 GitHub

GitHub là một nền tảng lưu trữ mã nguồn và quản lý dự án, hỗ trợ các nhà phát triển làm việc nhóm và quản lý phiên bản mã nguồn. GitHub cung cấp nhiều tính năng hữu ích để quản lý dự án và cộng tác.

Lý do sử dụng GitHub trong dự án:

- Quản lý phiên bản: GitHub sử dụng Git để quản lý phiên bản mã nguồn, giúp theo dõi và phục hồi các thay đổi trong dự án.
- Cộng tác nhóm: GitHub cung cấp các công cụ để làm việc nhóm, giúp chia sẻ mã nguồn và cộng tác dễ dàng.
- Tích hợp với CI/CD: GitHub hỗ trợ tích hợp với các hệ thống CI/CD, giúp tự động hóa quy trình build, test và deploy ứng dụng.
- Tài liệu hóa dự án: GitHub cung cấp các công cụ để tài liệu hóa dự án, giúp duy trì và chia sẻ tài liệu một cách hiệu quả.
- Quản lý issue và pull request: GitHub cho phép quản lý issue và pull request, giúp theo dõi và giải quyết các vấn đề trong dự án một cách có hệ thống.

5 Tổng kết

5.1 Link source của nhóm

Liên kết đến Github [tại đây](#).

5.2 Nhận xét

Ưu điểm của hệ thống in ấn thông minh: Hệ thống cho phép sinh viên dễ dàng tải tài liệu, chọn máy in, cài đặt thông số in trực tuyến và thanh toán nhanh chóng thông qua QR code. Việc tự động hóa các quy trình này giúp tiết kiệm thời gian, giảm thiểu sai sót và tối ưu hóa trải nghiệm người dùng. Ngoài ra, hệ thống cung cấp các công cụ cho cán bộ quản lý để theo dõi nhu cầu in ấn theo khu vực và ID máy in. Điều này giúp tối ưu hóa việc phân bổ máy in, giảm lãng phí tài nguyên thiết bị và đảm bảo rằng các khu vực có nhu cầu cao luôn được phục vụ tốt. Tính năng lưu trữ lịch sử in ấn giúp sinh viên dễ dàng tra cứu các tài liệu đã in, hạn chế lãng phí. Đồng thời, cán bộ quản lý có thể kiểm tra báo cáo chi tiết để phân tích và cải thiện quy trình vận hành. Bên cạnh đó, hệ thống linh hoạt trong việc cập nhật các thông số như số trang in miễn phí mỗi kỳ, các định dạng tài liệu được hỗ trợ... giúp đáp ứng các yêu cầu thay đổi của nhà trường.

Nhược điểm của hệ thống in ấn thông minh: Việc triển khai hệ thống cần đầu tư vào cơ sở hạ tầng, bao gồm máy in thông minh, thiết bị quản lý và chi phí phát triển phần mềm. Đây có thể là rào cản đối với các trường học có ngân sách hạn chế. Thêm vào đó, hệ thống hoạt động hiệu quả dựa trên sự ổn định của mạng internet và phần mềm. Nếu có sự cố về kết nối hoặc lỗi phần mềm, quá trình in ấn có thể bị gián đoạn, gây bất tiện cho người dùng. Hệ thống có thể gây khó khăn ban đầu cho những người dùng không quen thuộc với công nghệ hoặc ứng dụng web, đặc biệt là các học sinh không sở hữu thiết bị kết nối internet hoặc cán bộ lớn tuổi, dẫn đến việc cần thêm thời gian và tài

liệu hướng dẫn để họ sử dụng thành thạo. Ngoài, ở thời điểm hiện tại, hệ thống chỉ tích hợp thanh toán thông qua QR code nên sẽ gây bất tiện đối với những người dùng là học sinh và chưa mở tài khoản ngân hàng hoặc các hệ thống ví điện tử để quét mã thanh toán.

5.3 Hướng phát triển trong tương lai

Mở rộng tích hợp các phương thức thanh toán: Để tăng tính linh hoạt và tiện lợi cho người dùng, hệ thống sẽ được phát triển để hỗ trợ nhiều phương thức thanh toán hơn ngoài QR code, bao gồm thanh toán qua thẻ cào, chuyển khoản qua SMS hoặc các ví điện tử phổ biến khác. Điều này sẽ giúp học sinh chưa có tài khoản ngân hàng hoặc ví điện tử dễ dàng sử dụng hệ thống mà không gặp trở ngại.

Xây dựng ứng dụng di động: Để nâng cao trải nghiệm người dùng, hệ thống có thể được mở rộng dưới dạng ứng dụng di động. Ứng dụng này sẽ giúp sinh viên dễ dàng quản lý tài liệu, kiểm tra lịch sử in ấn và thực hiện các thao tác như nạp tiền hoặc mua thêm trang in trực tiếp trên điện thoại.

Tăng cường khả năng chịu lỗi và tối ưu hóa hiệu suất: Để giảm thiểu sự phụ thuộc vào mạng internet và phần mềm, hệ thống cần được phát triển thêm các tính năng hoạt động ngoại tuyến tạm thời, cho phép lưu trữ các yêu cầu in cục bộ trên máy in. Khi kết nối mạng được khôi phục, các yêu cầu này sẽ được tự động đồng bộ hóa. Đồng thời, việc sử dụng các công nghệ mới như điện toán đám mây và cân bằng tải sẽ giúp đảm bảo hiệu suất ổn định và khả năng mở rộng cho hệ thống.

Tích hợp trí tuệ nhân tạo (AI): AI có thể được áp dụng để phân tích dữ liệu sử dụng hệ thống, từ đó đề xuất các giải pháp tối ưu hóa. Ví dụ, AI có thể dự đoán nhu cầu in ấn theo thời gian thực, giúp cán bộ quản lý phân bổ máy in hiệu quả hơn.

Cải thiện trải nghiệm người dùng và cung cấp tài liệu hướng dẫn: Để hỗ trợ những người không quen thuộc với công nghệ, hệ thống sẽ cung cấp giao diện đơn giản hơn cùng với các tài liệu hướng dẫn chi tiết. Các tài liệu này có thể bao gồm video hướng dẫn hoặc trợ lý ảo giúp sinh viên và cán bộ từng bước làm quen với hệ thống.

Hỗ trợ đa ngôn ngữ và tùy chỉnh theo từng trường học: Hệ thống sẽ được mở rộng để hỗ trợ đa ngôn ngữ, giúp các trường có sinh viên quốc tế sử dụng dễ dàng hơn. Đồng thời, các tính năng tùy chỉnh như định dạng tài liệu, chi phí trang in, hoặc phương thức thanh toán có thể được linh hoạt cấu hình theo nhu cầu cụ thể của từng trường học.

6 Tài liệu tham khảo

Tài liệu

- [1] Duckett, J. (2011). *HTML and CSS: Design and Build Websites*. Wiley.
- [2] Ferguson, M. (2013). *Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics (4th ed.)*. O'Reilly Media.
- [3] Cantelon, M., Harter, M., Holowaychuk, T., Rajlich, N. (2014). *Node.js in Action (2nd ed.)*. Manning Publications.
- [4] Wexler, A., Telles, M. (2017). *Express in Action: Writing, Building, and Testing Node.js Applications*. Manning Publications.
- [5] Klee, B. (2021). *MySQL Cookbook: Solutions for Database Developers and Administrators (4th ed.)*. O'Reilly Media.
- [6] Kingsley-Hughes, K. (2018). *Node.js Web Development (5th ed.)*. Packt Publishing.



- [7] Chinnathambi, B. (2021). *Practical Node.js: Building Real-World Scalable Web Apps* (2nd ed.). Apress.