

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KĨ THUẬT MÁY TÍNH



BÁO CÁO
BÀI TẬP LỚN
HỌC KỲ: 251 - NĂM HỌC: 2025-2026

Phát triển Ứng dụng trên Thiết bị Di động (CO3043)

GVHD: Hoàng Lê Hải Thanh

SVTH: Trần Nam Sơn 2212956

Hồ Quốc Khương 2211709

Nguyễn Phúc Bình 2210319

Phạm Quốc Toàn 2213539

TP. Hồ Chí Minh, Tháng 1/2026

Mục lục

1 Giới thiệu	1
1.1 Lý do chọn dự án	1
1.2 Mục tiêu dự án	1
1.3 Nghiên cứu thị trường	2
1.3.1 Xây dựng bộ tiêu chí đánh giá	2
1.3.2 Facebook Group	3
1.3.3 Facebook Marketplace	4
1.3.4 Oreka	6
1.3.5 RaoVatVN	7
1.3.6 Chợ Tốt	9
1.3.7 Tổng kết	11
1.4 Business Model Canvas	13
1.4.1 Ý tưởng (IDEA)	13
1.4.2 Bán hàng (SELL)	13
1.4.3 Mô hình kinh doanh (BUSINESS MODEL)	14
1.4.4 Giá trị nổi bật (Unique Value Proposition)	14
2 Các tính năng của StudyShare	15
2.1 Danh sách các tính năng chính	15
2.2 Ma trận tác động–nỗ lực (Impact–Effort Matrix)	16
2.2.1 Easy Wins	16
2.2.2 Strategical Wins	17
2.2.3 Weak Wins	17
2.2.4 No Wins	17
2.3 Các tính năng trong MVP	17
3 Hướng tiếp cận và phương pháp	18
3.1 Quy trình phát triển theo hướng Agile	18



3.2	Công cụ cộng tác và quản lý dự án	18
4	Chi tiết hiện thực	20
4.1	Frontend	20
4.2	Backend	21
5	Kiến trúc hệ thống	23
5.1	Model	23
5.2	View	24
5.3	Controller	24
6	Thiết kế Web Service	25
6.1	Authentication & Authorization	25
6.2	User Management	26
6.3	Common Data	26
6.4	Posts Management	26
6.5	Real-time Chat	27
6.6	Notification	27
6.7	File Upload	27
6.8	Validation & Error Handling	28
6.9	Security	28
7	Trải nghiệm người dùng	29
7.1	Đăng tin bán / cho tặng	29
7.1.1	User Story (nhu cầu sử dụng của người dùng)	29
7.1.2	Yêu cầu	29
7.1.3	Luồng hoạt động	30
7.2	Tìm kiếm và lọc sản phẩm	32
7.2.1	User Story (nhu cầu sử dụng của người dùng)	32
7.2.2	Yêu cầu	32
7.2.3	Luồng hoạt động	33



8 Chiến lược triển khai	35
9 Phản hồi người dùng	36
10 Tổng kết	38
10.1 Đánh giá	38
10.2 Khó khăn	38
10.3 Bài học	39
10.4 Định hướng	39

1 Giới thiệu

1.1 Lý do chọn dự án

Trong quá trình học tập tại Trường Đại học Bách Khoa, sinh viên thường xuyên có nhu cầu trao đổi tài liệu học tập và dụng cụ học tập như giáo trình, đề thi, sách tham khảo, linh kiện và các thiết bị học tập cũ. Hiện nay, các hoạt động này chủ yếu được thực hiện thông qua các nền tảng mạng xã hội phổ biến như Facebook, Zalo hoặc Telegram.

Tuy nhiên, việc sử dụng các nền tảng không chuyên biệt này bộc lộ nhiều hạn chế. Các thông tin được đăng tải rời rạc, thiếu tổ chức và khó tìm kiếm lại khi cần thiết. Bài viết nhanh chóng bị trôi theo dòng thời gian, dẫn đến việc người dùng dễ bỏ lỡ những tài nguyên học tập quan trọng. Bên cạnh đó, các nền tảng này không được thiết kế riêng cho mục đích chia sẻ học liệu, nên chưa đáp ứng tốt nhu cầu thực tế của sinh viên trong môi trường học tập đại học. Hiện tại, vẫn chưa tồn tại một nền tảng nội bộ chính thức dành riêng cho sinh viên trong trường để phục vụ việc chia sẻ và trao đổi tài nguyên học tập.

Xuất phát từ nhu cầu thực tiễn đó, StudyShare được xây dựng nhằm cung cấp một giải pháp tập trung, rõ ràng và tiện lợi cho sinh viên Trường Đại học Bách Khoa. Ứng dụng đóng vai trò như một không gian chia sẻ học tập chung, nơi sinh viên có thể dễ dàng đăng tải, tìm kiếm và trao đổi tài liệu cũng như dụng cụ học tập thông qua các hình thức như cho mượn, mua bán hoặc tặng lại. Việc tập trung hóa các hoạt động này giúp tiết kiệm thời gian tìm kiếm, nâng cao hiệu quả học tập và giảm sự phụ thuộc vào các nền tảng mạng xã hội không chuyên biệt.

Nhu cầu chia sẻ tài nguyên học tập là nhu cầu thiết yếu và diễn ra thường xuyên, đặc biệt đối với sinh viên các ngành kỹ thuật và công nghệ. Thông qua việc chuẩn hóa quy trình trao đổi và tổ chức thông tin một cách khoa học, StudyShare không chỉ giúp sinh viên tiếp cận tài nguyên học tập nhanh chóng hơn mà còn khuyến khích tinh thần chia sẻ và tái sử dụng trong cộng đồng sinh viên. Điều này góp phần giảm chi phí học tập, hạn chế lãng phí và thúc đẩy sự gắn kết giữa các sinh viên trong cùng khoa và toàn trường.

Về lâu dài, StudyShare có tiềm năng phát triển thành một nền tảng chia sẻ học liệu uy tín trong nội bộ sinh viên Khoa học Máy tính, từ đó mở rộng mô hình cho toàn bộ sinh viên Trường Đại học Bách Khoa, hướng tới xây dựng một hệ sinh thái học tập bền vững và hiệu quả.

1.2 Mục tiêu dự án

Dự án **StudyShare** được xây dựng nhằm giải quyết các vấn đề trên bằng cách phát triển một nền tảng chia sẻ tài liệu và dụng cụ học tập dành riêng cho sinh viên Trường Đại học Bách Khoa. Các mục tiêu chính của dự án bao gồm:

- Tạo ra một nền tảng tập trung, giúp sinh viên dễ dàng đăng tải, tìm kiếm và quản lý tài liệu cũng như dụng cụ học tập.
- Kết nối trực tiếp giữa người có nhu cầu và người sẵn sàng chia sẻ, mua bán hoặc tặng lại tài nguyên học tập.



- Giảm sự phụ thuộc vào các nền tảng mạng xã hội không chuyên biệt, đồng thời nâng cao tính hệ thống và hiệu quả trong việc trao đổi thông tin.
- Xây dựng một cộng đồng sinh viên có tinh thần chia sẻ, hỗ trợ lẫn nhau trong học tập.

1.3 Nghiên cứu thị trường

Trong bối cảnh số lượng sinh viên Trường Đại học Bách Khoa ngày càng gia tăng, các sinh viên sau khi học các môn học như **thí nghiệm** hoặc các môn **chính trị**, ... có nhu cầu cao về **chia sẻ học liệu** và **trao đổi dụng cụ học tập**. Tuy nhiên, việc tìm kiếm và chia sẻ các học liệu hiện có vẫn gặp nhiều rào cản.

Phần lớn tài liệu được chia sẻ rải rác qua các bài đăng trên **Facebook**, **Chợ Tốt**, **Oreka**, ... Điều này dẫn đến tình trạng **phân tán thông tin**, **thiếu hệ thống** và **khó tiếp cận**. Sinh viên thường phải mất nhiều thời gian để lục lọi các tài liệu từ các khóa trước, trong khi nhu cầu trao đổi sách, giáo trình và dụng cụ học tập cũng không có một kênh tập trung rõ ràng.

Để phát triển một giải pháp phù hợp, nhóm tiến hành **nghiên cứu thị trường** nhằm:

- Xác định nhu cầu thực tế và khó khăn của sinh viên, thông qua khảo sát các sinh viên tại Trường Đại học Bách Khoa.
- Phân tích các nền tảng tương tự hiện có trong nước, từ đó rút ra điểm mạnh, điểm yếu và các khoảng trống chưa được đáp ứng.
- Tìm kiếm cơ hội thị trường rõ ràng để **StudyShare** trở thành nền tảng chuyên biệt, tập trung vào việc hỗ trợ sinh viên Bách Khoa học tập hiệu quả hơn.

1.3.1. Xây dựng bộ tiêu chí đánh giá

Dựa trên **insight từ khảo sát sinh viên** về những khó khăn khi tìm kiếm, chia sẻ tài liệu và trao đổi dụng cụ học tập, nhóm xây dựng một **bộ tiêu chí đánh giá cụ thể** để phân tích 5 nền tảng cạnh tranh hiện tại như **Facebook Group**, **Facebook Marketplace**, **Telegram**, **ChoTot**, **Zalo**.

Bộ tiêu chí được chia thành **hai nhóm chính**:

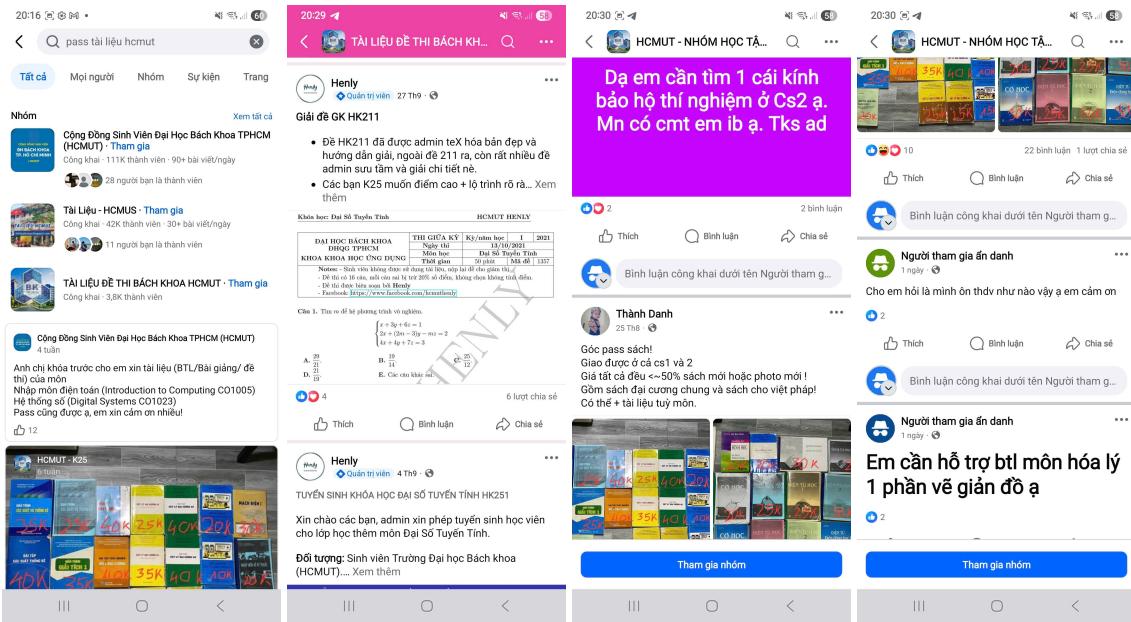
- **Nhóm Tính năng (Functional)**
- **Nhóm Trải nghiệm người dùng (UX)**

Nhóm	Tiêu chí	Thang đo	Mô tả đánh giá
Functional	Đăng bài trao đổi	1–5	1 = Thao tác phức tạp; 5 = Đăng dễ dàng, hiển thị tốt
Functional	Tìm kiếm và lọc	1–5	1 = Tìm cơ bản; 5 = Bộ lọc nâng cao
Functional	Dánh giá & bình luận	1–5	1 = Không hỗ trợ; 5 = Có đánh giá rõ ràng
Functional	Trao đổi dụng cụ học tập	1–5	1 = Không phân loại; 5 = Có khu vực riêng
Functional	Xác thực sinh viên	Có/Không	Hỗ trợ xác minh sinh viên nội bộ
UX	Mức độ dễ sử dụng	1–5	1 = Khó dùng; 5 = Giao diện thân thiện
UX	Ẩn bài đã hoàn tất	1–5	1 = Không hỗ trợ; 5 = Có đánh dấu hoàn tất
UX	Phù hợp sinh viên	1–5	1 = Ít sinh viên dùng; 5 = Phù hợp cao

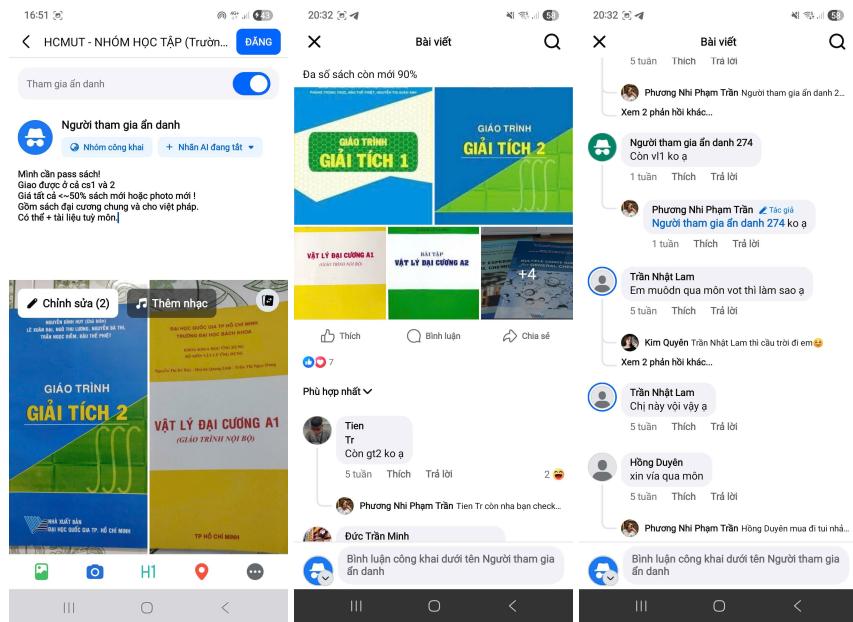
1.3.2. Facebook Group

Facebook Group là kênh trao đổi phổ biến nhất của sinh viên Bách Khoa hiện nay. Hầu như mọi hoạt động trao đổi học liệu, dụng cụ học tập đều diễn ra trên các nhóm như “Chợ BK”, “Ký túc xá Bách Khoa”, “Hội Sinh viên năm 3 – năm 4”,... Đây là nền tảng gần gũi, có lượng người dùng lớn và phản ánh rõ nhất thói quen thực tế của sinh viên.

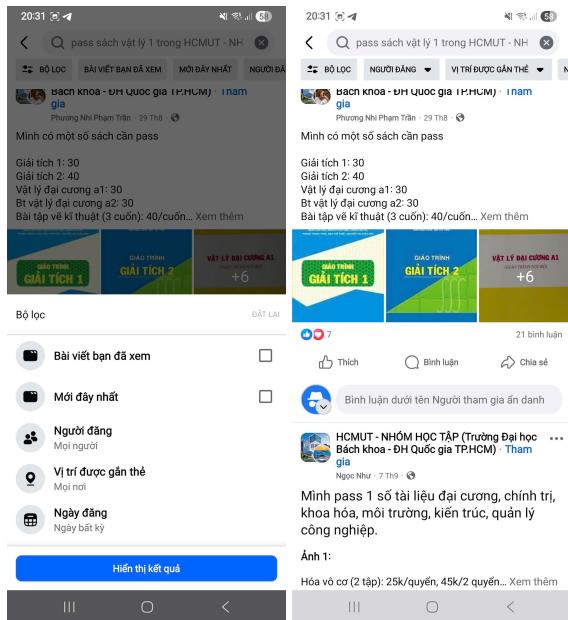
Facebook là nền tảng quen thuộc với sinh viên, thao tác đăng bài và tương tác (bình luận, nhắn tin) rất thuận tiện. Tuy nhiên, các group này không tập trung chuyên biệt cho việc trao đổi học liệu, nên bài đăng bị pha tạp, gây khó khăn khi tìm kiếm tài liệu.



Quy trình đăng bài rất đơn giản (chỉ 3 bước: vào nhóm, soạn bài, đăng), giúp người dùng dễ dàng chia sẻ. Tuy vậy, nhiều bài đăng cũ không được xóa sau khi hoàn tất trao đổi, khiến nhóm chứa nhiều thông tin lỗi thời và gây mất thời gian khi tìm kiếm.



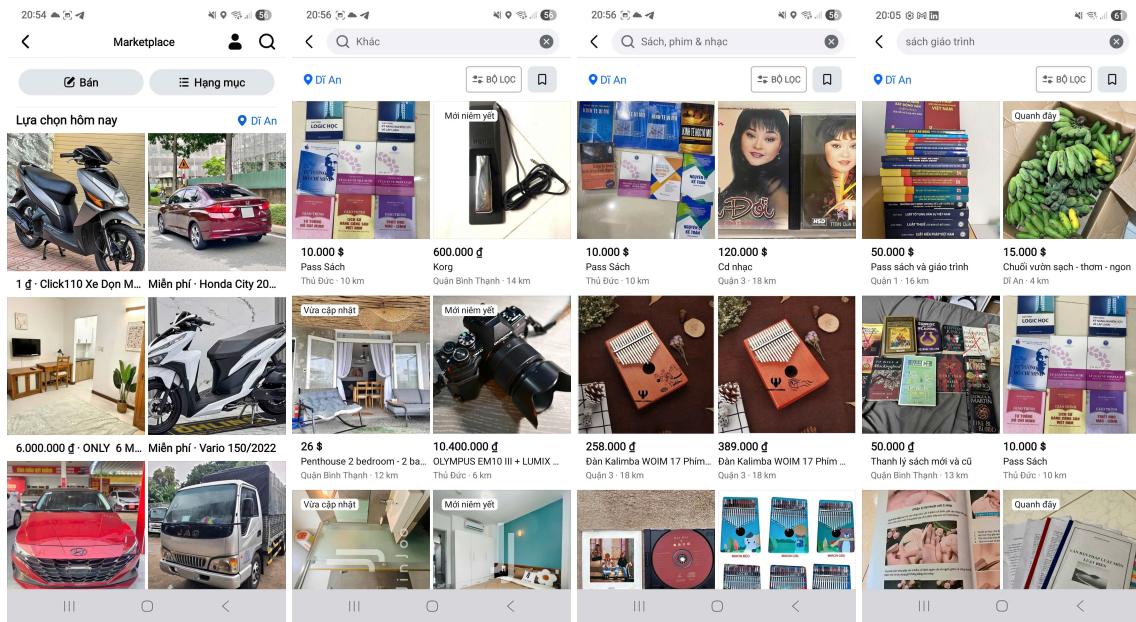
Chức năng tìm kiếm trong nhóm chỉ hỗ trợ theo từ khóa và bộ lọc cơ bản, không có hệ thống phân loại theo khoa, môn hay loại học liệu. Điều này làm việc tìm loại tài liệu cụ thể trở nên kém hiệu quả.



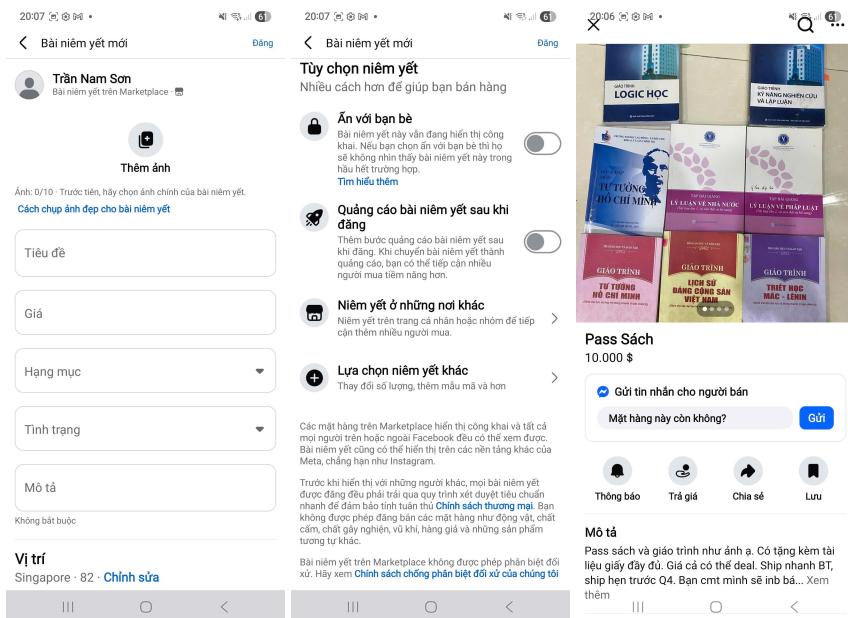
1.3.3. Facebook Marketplace

Marketplace là tính năng thương mại tích hợp sẵn trong Facebook, được nhiều sinh viên sử dụng để bán hoặc pass lại sách, áo blouse, đồ học tập, đặc biệt khi cần tiếp cận nhiều người nhanh chóng.

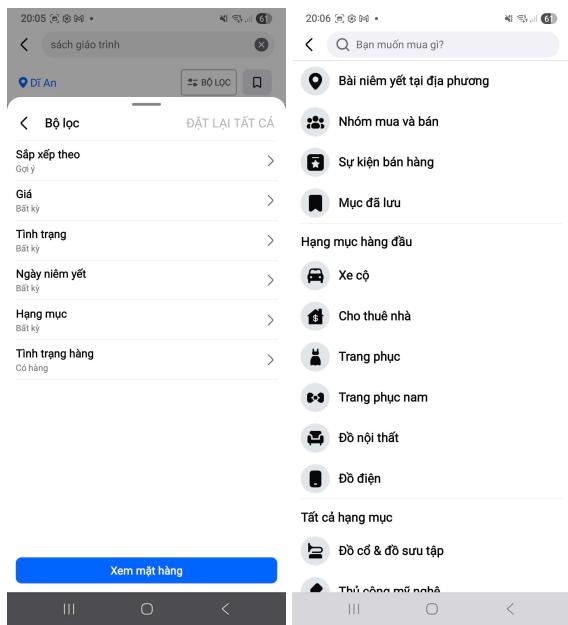
Facebook Marketplace có giao diện rõ ràng, trực quan, thao tác xem bài và liên hệ người đăng rất dễ dàng. Tuy nhiên, nền tảng này không được thiết kế riêng cho học liệu, nên các bài đăng về tài liệu/dụng cụ học tập bị lẫn với nhiều loại mặt hàng khác.



Quy trình đăng bài tương đối đơn giản, gồm 2 bước (điền thông tin cơ bản và đăng tải hình ảnh). Tuy vậy, việc phân loại không có mục riêng cho tài liệu học tập nên người đăng thường chọn danh mục ‘Khác’, dẫn đến khó tìm chính xác khi người mua tra cứu.



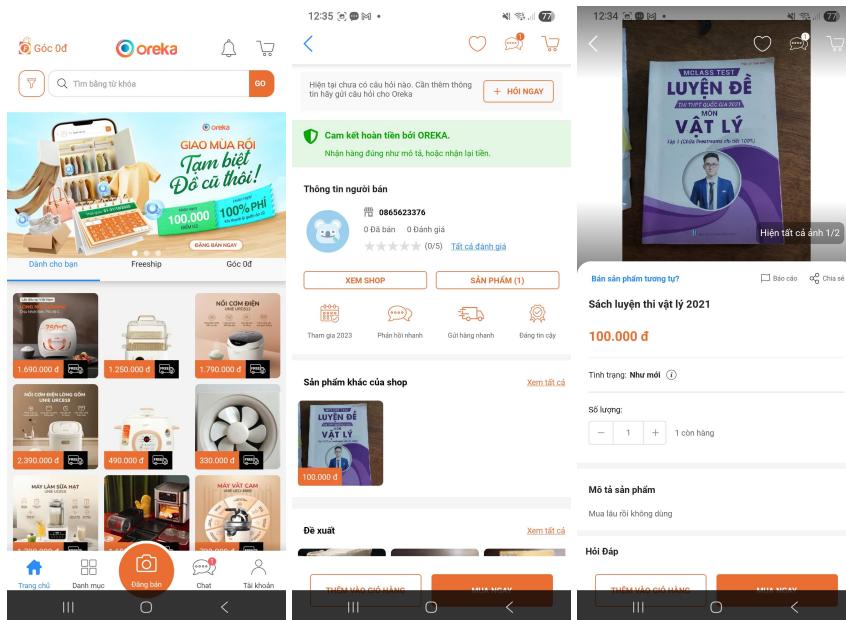
Tính năng tìm kiếm hỗ trợ lọc theo vị trí, giá và danh mục, nhưng lại không hỗ trợ phân loại theo khoa, môn hay học kỳ. Vì vậy, việc tìm kiếm học liệu cụ thể trên Marketplace khá mất thời gian và dễ bị lẫn với các bài đăng không liên quan.



1.3.4. Oreka

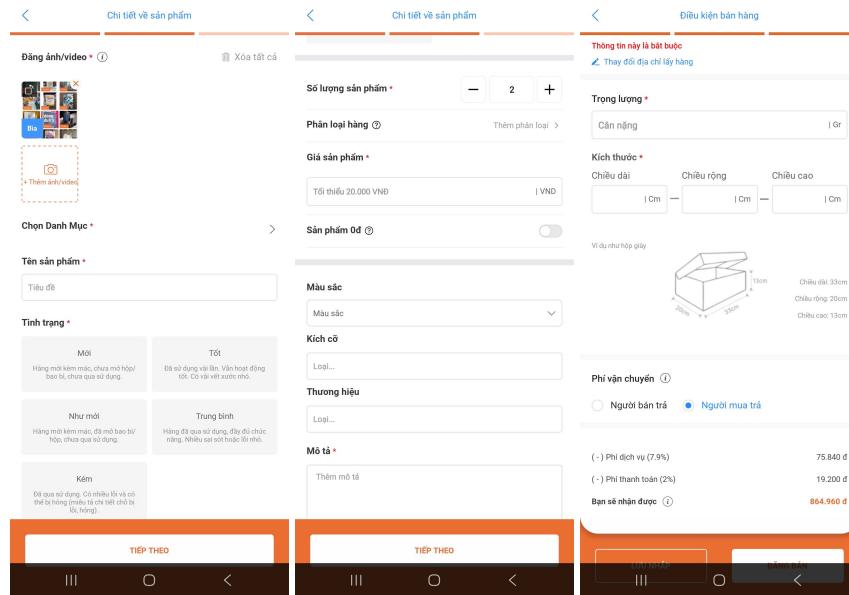
Oreka là nền tảng Việt Nam được xây dựng với mục tiêu thúc đẩy trao đổi và tái sử dụng đồ cũ, hướng đến cộng đồng sinh viên và người trẻ. Ứng dụng này cho phép người dùng đăng bài trao đổi, cho tặng hoặc mua bán đồ dùng cá nhân một cách nhanh chóng. Do có định hướng gần với mô hình “trao đổi học liệu – dụng cụ học tập”, Oreka được nhóm chọn để quan sát khả năng áp dụng cho sinh viên Bách Khoa.

Oreka có giao diện hiện đại, màu sắc tươi sáng, dễ nhìn và dễ thao tác. Đây là nền tảng chuyên về trao đổi, mua bán đồ dùng cá nhân, nên có phần tương đồng với mục tiêu trao đổi của sinh viên. Tuy nhiên, nền tảng chưa có danh mục riêng cho tài liệu hoặc dụng cụ học tập, nên sinh viên khó tìm được đúng đối tượng hoặc nội dung phù hợp.

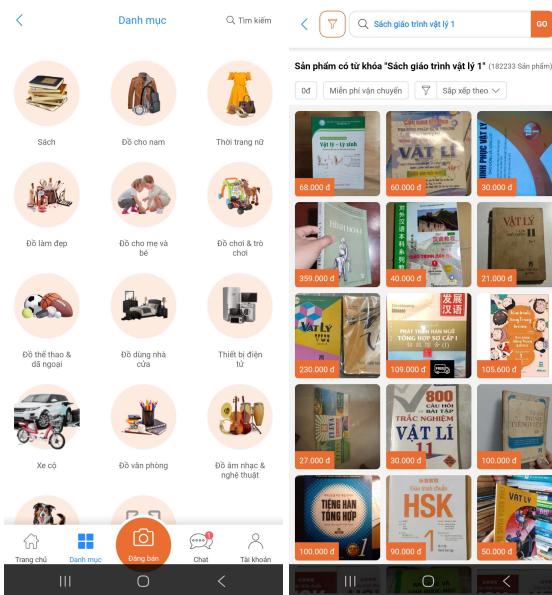


Quy trình đăng bài trên Oreka tương đối phức tạp và nhiều bước. Điều này có thể tồn

nhiều thời gian hơn khi người bán đăng bài nhưng nó cũng sẽ giúp họ mô tả chi tiết hơn để người mua dễ hiểu hơn. Tuy nhiên, đối với mục đích trao đổi tài liệu/dụng cụ học tập cho sinh viên thì việc này khá dư thừa và tốn thời gian. Đồng thời, người dùng sẽ phải chịu phí trung gian khi bán hoặc mua trên ứng dụng.



Oreka có thanh tìm kiếm hoạt động ổn định và hỗ trợ lọc theo khu vực, tình trạng vật phẩm. Tuy nhiên, không có hệ thống lọc nâng cao theo chủ đề học tập hay loại học liệu, nên người dùng vẫn phải xem thủ công từng bài đăng để tìm vật phẩm phù hợp.

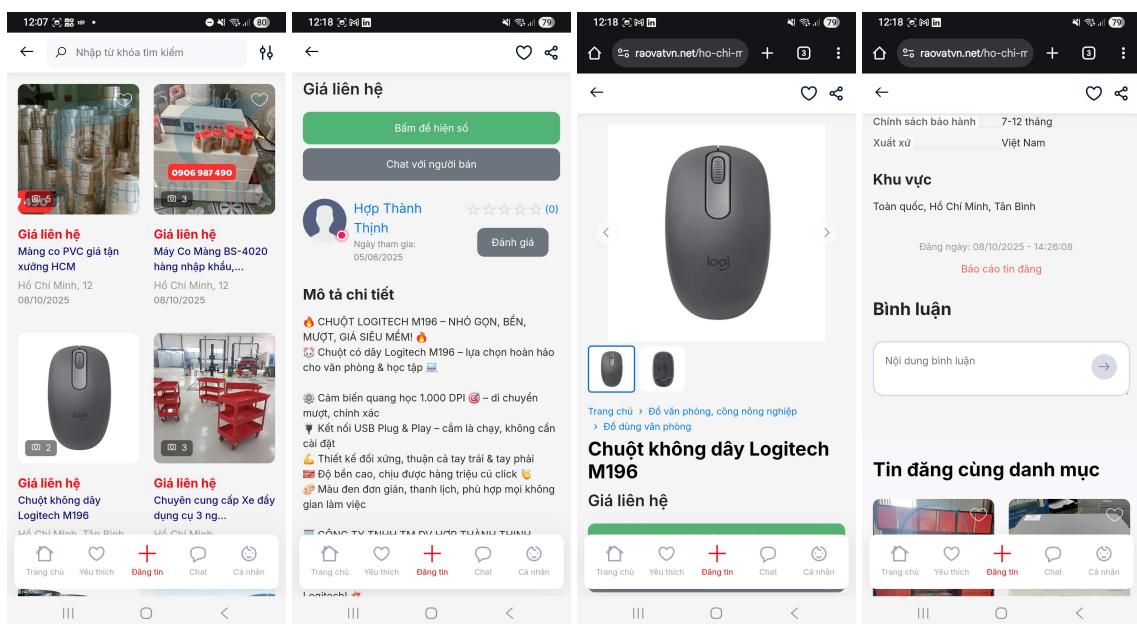


1.3.5. RaoVatVN

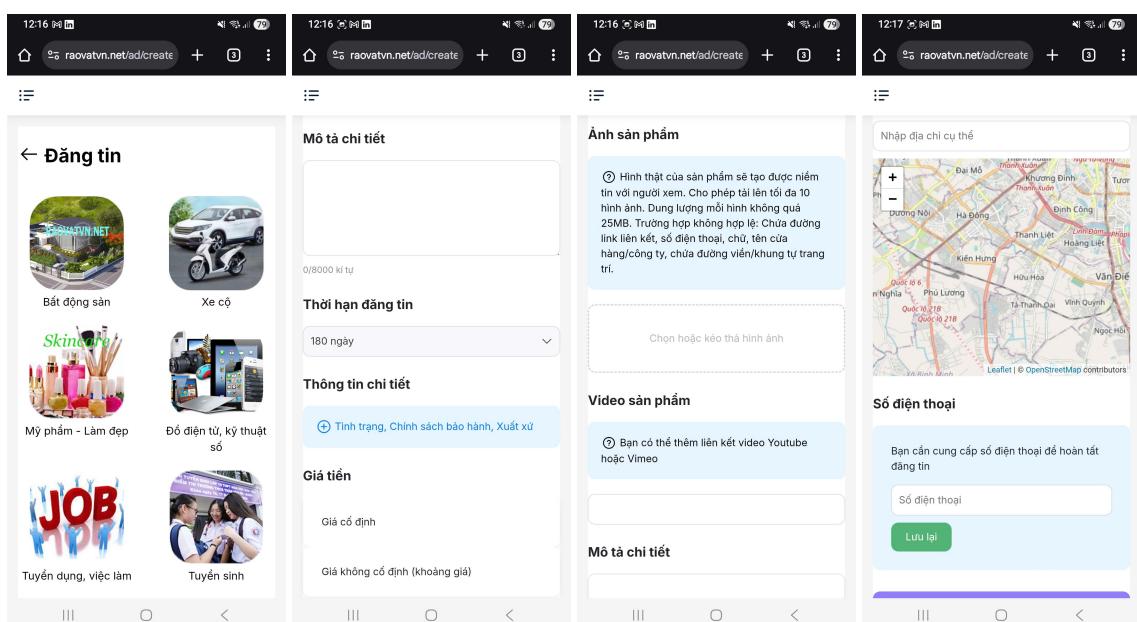
RaoVatVN là một trong những trang rao vặt trực tuyến lâu đời và phổ biến tại Việt Nam, tập hợp nhiều danh mục như đồ dùng cá nhân, việc làm, bất động sản, xe cộ, v.v. Nền tảng này được chọn để khảo sát vì có hệ thống đăng tin và tìm kiếm đa dạng, phản ánh

rõ mô hình giao dịch truyền thống trên web, giúp nhóm đánh giá điểm mạnh và hạn chế khi áp dụng cho môi trường sinh viên.

RaoVatVN có giao diện truyền thống, bố cục nhiều chữ và mục hiển thị dày đặc. Trang web hoạt động ổn định nhưng chưa thân thiện với người dùng trẻ và hiển thị chưa tối ưu trên thiết bị di động. Ngoài ra, do hướng đến nhiều lĩnh vực khác nhau (việc làm, bất động sản, dịch vụ...), nên không tập trung vào mảng trao đổi học liệu cho sinh viên. Có tính năng bình luận và đánh giá sản phẩm



Quy trình đăng bài khá phức tạp do phải nhập rất nhiều thông tin, không thể đăng video mà chỉ có thể liên kết một video mà người dùng có sẵn. Tuy nhiên, người dùng được lựa chọn thời gian tin sẽ xuất hiện (tối đa 180 ngày) và các thông tin chi tiết khác có thể hỗ trợ cho người tìm kiếm biết được chi tiết hơn.



Công cụ tìm kiếm chỉ hỗ trợ theo từ khóa và vị trí, người dùng có thể lọc theo giá, danh mục, thời gian đã đăng,...

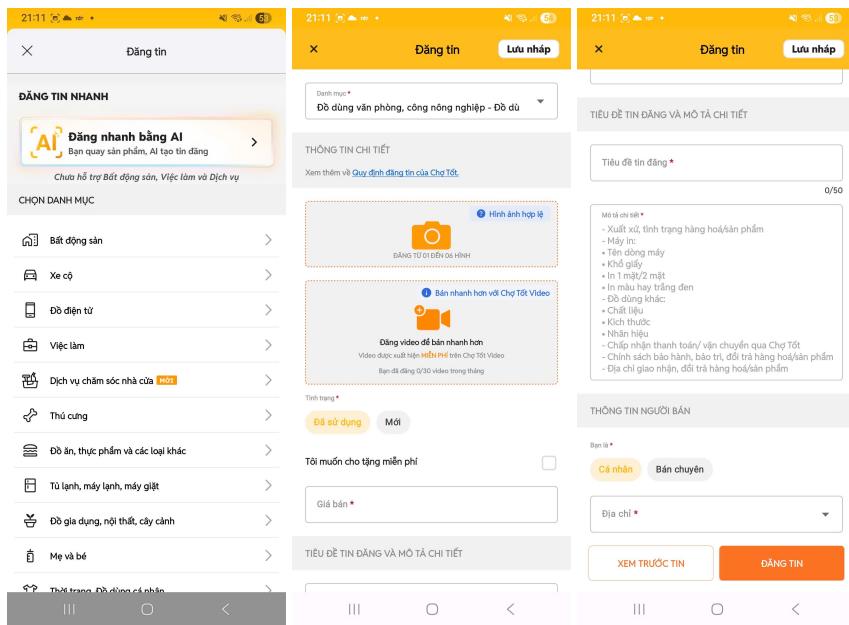
1.3.6. Chợ Tốt

Là nền tảng mua bán đồ cũ lớn nhất Việt Nam, Chợ Tốt có nhiều sinh viên sử dụng để pass sách giáo trình hoặc dụng cụ học tập. Đây là một đối thủ gián tiếp tiềm năng.

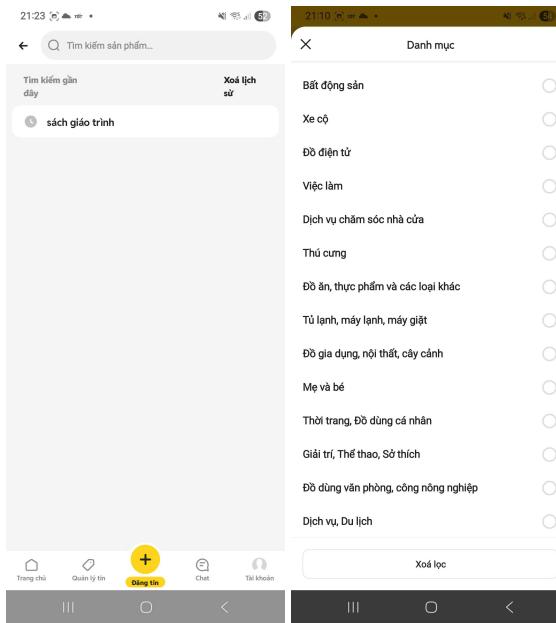
Chợ Tốt có giao diện khá trực quan, tập trung vào việc mua bán. Người dùng dễ dàng xem thông tin, hình ảnh và liên hệ với người bán qua tin nhắn hoặc gọi điện trực tiếp. Tuy nhiên, nền tảng này không được xây dựng riêng cho sinh viên nên thiếu các tính năng đặc thù như lọc theo môn học, học kỳ hoặc khoa.

Đăng bài khá nhanh, gồm 3 bước đơn giản (chọn danh mục, nhập thông tin và tải hình

ảnh). Tuy nhiên, do không có mục riêng cho học liệu/dụng cụ học tập, người dùng thường đăng vào mục ‘Đồ dùng văn phòng, công nông nghiệp’, dẫn đến việc tìm kiếm có thể bị lẫn với các loại sản phẩm khác.



Chức năng tìm kiếm mạnh với từ khóa, giá và khu vực, nhưng không hỗ trợ phân loại chuyên sâu. Người dùng phải lọc thủ công và đọc từng bài, gây tốn thời gian khi cần tìm đúng loại tài liệu hoặc dụng cụ học tập cụ thể.





1.3.7. Tổng kết

Tiêu chí	Facebook Group	Facebook Marketplace	Chợ Tốt	Oreka	RaoVatVN
Đăng bài trao đổi	4	4	4	3	3
Tìm kiếm và lọc	2	4	3	4	4
Phù hợp với sinh viên	5	3	3	3	2
Đánh giá & bình luận	3	1	4	4	3
Giúp sinh viên trao đổi dụng cụ học tập	3	2	3	2	1
Xác thực sinh viên nội bộ	0	0	0	0	0
Mức độ dễ sử dụng	5	4	4	4	3
Mức độ phổ biến / lan tỏa	5	3	3	3	3
Hỗ trợ ẩn bài đã hoàn tất	2	4	4	4	3

Qua quá trình khảo sát, nhóm nhận thấy các nền tảng hiện nay (Facebook Group, Marketplace, Chợ Tốt, Oreka, RaoVatVN) đều có điểm mạnh nhất định trong việc hỗ trợ người dùng đăng và tìm kiếm bài viết trao đổi tài liệu hoặc dụng cụ học tập:

- Hầu hết sinh viên đã tương đối quen thuộc với Facebook, Chợ Tốt hoặc các nền tảng mua bán tương tự, nên việc thao tác đăng bài, xem bài và nhắn tin rất thuận tiện.
- Facebook Group là kênh có mức độ lan tỏa cao nhất nhờ lượng thành viên lớn, giúp bài đăng nhanh chóng tiếp cận cộng đồng sinh viên.
- Các nền tảng như Chợ Tốt, Marketplace hay Oreka có quy trình đăng bài rõ ràng, hỗ trợ hình ảnh và mô tả chi tiết, giúp người dùng dễ chia sẻ và tìm hiểu thông tin.
- Các ứng dụng như Marketplace, Chợ Tốt hay RaoVatVN cho phép lọc theo giá, tình trạng, vị trí – hỗ trợ tìm kiếm tốt hơn so với Facebook Group.

Tổng thể, các nền tảng hiện nay phù hợp cho việc mua bán nhanh hoặc trao đổi vật dụng phổ thông, nhưng chưa thật sự đáp ứng được nhu cầu đặc thù của sinh viên trong việc chia sẻ học liệu và dụng cụ học tập.

Mặc dù có nhiều ưu điểm, các giải pháp hiện tại vẫn còn tồn tại những khoảng trống lớn khi xét đến nhu cầu của sinh viên đại học, đặc biệt trong môi trường học thuật:

- Không có danh mục riêng cho học liệu / dụng cụ học tập: Người dùng phải đăng bài trong các danh mục chung như “Đồ dùng văn phòng” hoặc “Khác”, khiến bài đăng dễ bị lẫn và khó tìm.
- Thiếu hệ thống lọc theo nhu cầu học tập: Không có cách nào lọc theo môn học, khoa, học kỳ hoặc chương trình đào tạo, làm việc tìm kiếm tài liệu cụ thể trở nên mất thời gian và kém hiệu quả.
- Không xác thực đối tượng sinh viên: Mọi người đều có thể đăng và xem bài, dẫn đến rủi ro về spam, lừa đảo hoặc đăng sai đối tượng. Điều này đặc biệt bất tiện khi sinh viên chỉ muốn trao đổi trong phạm vi trường hoặc khoa.



- Không có trạng thái quản lý bài đăng sau khi trao đổi: Nhiều bài cũ vẫn còn hiển thị dù giao dịch đã hoàn tất, gây nhiễu thông tin và giảm trải nghiệm người dùng.

Từ đó có thể thấy, nhu cầu trao đổi – chia sẻ học liệu giữa sinh viên vẫn chưa được phục vụ đầy đủ bởi các nền tảng hiện có.

Dựa trên những khoảng trống nêu trên, nhóm nhận thấy cơ hội rõ ràng cho việc phát triển một ứng dụng chuyên biệt cho sinh viên trao đổi học liệu và dụng cụ học tập, với định hướng như sau:

- Tập trung vào nhu cầu học tập:** Ứng dụng chỉ hướng tới trao đổi, chia sẻ tài liệu dụng cụ học tập.
- Cá nhân hóa theo môi trường học:** Có thể phân loại và lọc bài đăng theo khoa, môn học, học kỳ, giúp sinh viên nhanh chóng tìm đúng tài liệu cần thiết.
- Xác thực người dùng theo trường / email sinh viên:** Chỉ cho phép đăng và tương tác trong phạm vi cộng đồng sinh viên, đảm bảo an toàn và đúng đối tượng.
- Quản lý bài đăng thông minh:** Cung cấp trạng thái “Đang trao đổi” – “Đã hoàn tất”, hoặc tự động ẩn bài cũ để đảm bảo thông tin luôn cập nhật.
- Tăng tính tương tác:** Thêm tính năng bình luận, hỏi đáp, đánh giá người đăng.

Các nền tảng hiện tại tuy đã đáp ứng một phần nhu cầu trao đổi của sinh viên, nhưng chưa có giải pháp nào thật sự tập trung vào khía cạnh học tập. Vì vậy, nhóm đề xuất phát triển một ứng dụng hướng tới cộng đồng sinh viên, giúp đăng – tìm – trao đổi học liệu và dụng cụ học tập nhanh chóng, đúng người, đúng nhu cầu.

1.4 Business Model Canvas

1.4.1. Ý tưởng (IDEA)

Vấn đề (Problem)	Giải pháp (Solution)	Giá trị mang lại (Value)
<ul style="list-style-type: none"> Sinh viên Trường Đại học Bách Khoa có nhu cầu trao đổi, mua bán hoặc chia sẻ lại tài liệu và dụng cụ học tập đã qua sử dụng, nhưng hiện chưa có một nền tảng chuyên biệt và đảm bảo an toàn cho mục đích này. Các nền tảng phổ biến như Facebook Group hoặc Facebook Marketplace không được thiết kế riêng cho môi trường sinh viên, dẫn đến việc thông tin bị pha tạp, khó tìm kiếm và thiếu kiểm soát. Nhiều sinh viên lãng phí tài nguyên học tập (sách, dụng cụ, áo blouse) do không có kênh phù hợp để chuyển giao cho các khóa sau. 	<ul style="list-style-type: none"> Mô hình chia sẻ học liệu nội bộ StudyShare, được xây dựng dành riêng cho sinh viên Trường Đại học Bách Khoa. Nền tảng cho phép người dùng đăng tải, tìm kiếm và trao đổi học liệu, sách và dụng cụ học tập trong phạm vi nội bộ, có xác thực tài khoản sinh viên nhằm đảm bảo tính an toàn và tin cậy. 	<ul style="list-style-type: none"> Giảm thiểu thời gian tìm kiếm và tiếp cận học liệu cần thiết. Tăng tỷ lệ tái sử dụng tài nguyên học tập trong cộng đồng sinh viên. Giảm chi phí học tập trung bình của sinh viên thông qua việc tái sử dụng. Nâng cao tính an toàn, minh bạch và hiệu quả trong hoạt động trao đổi học liệu nội bộ.

1.4.2. Bán hàng (SELL)

Lợi thế cạnh tranh (Advantage)	Đối tượng người dùng (Customer/User)	Chỉ số đo lường thành công (Metrics)
<ul style="list-style-type: none"> Công đồng nội bộ được xác thực bằng email sinh viên (@hcmut.edu.vn). Bộ lọc đặc thù cho trao đổi giữa sinh viên, cho phép tra cứu và lọc theo môn học và ngành học. Môi trường đáng tin cậy có kiểm duyệt nhằm hạn chế bài đăng spam hoặc người ngoài trường. Tích hợp tính năng đánh giá và bình luận để tăng độ tin cậy trong quá trình trao đổi. Cho phép người dùng nhắn tin trực tiếp trong ứng dụng. 	<ul style="list-style-type: none"> Sinh viên Đại học Bách Khoa TP.HCM từ năm nhất đến năm cuối và sinh viên sắp ra trường. 	<ul style="list-style-type: none"> Số lượng học liệu hoặc dụng cụ được đăng tải mỗi tháng. Số lượng học liệu hoặc tài liệu trao đổi thành công mỗi tháng. Mức độ hài lòng của người dùng sau khi sử dụng. Tỷ lệ tái sử dụng học liệu qua các học kỳ. Tỷ lệ người dùng quay lại (Retention Rate) hàng tháng.

1.4.3. Mô hình kinh doanh (BUSINESS MODEL)

Kênh tiếp cận (Channels)	Chi phí vận hành (Costs)	Doanh thu (Revenue)
<ul style="list-style-type: none"> Truyền thông qua các Câu lạc bộ sinh viên, fanpage Khoa, group Facebook chính thức của trường. Marketing nội bộ trong khuôn viên trường (poster, sự kiện, truyền miệng). Tận dụng email sinh viên để gửi mail mời đăng ký tài khoản. Lập các kênh mạng xã hội để quảng bá (Tiktok, Facebook). 	<ul style="list-style-type: none"> Chi phí hạ tầng hệ thống (hosting, lưu trữ dữ liệu, bảo mật). Chi phí quản lý cộng đồng và kiểm duyệt nội dung. Chi phí truyền thông nội bộ và hỗ trợ kỹ thuật. 	<ul style="list-style-type: none"> Giai đoạn đầu (MVP): miễn phí hoàn toàn. Giai đoạn sau có thể mở rộng: <ul style="list-style-type: none"> Quảng cáo nội bộ (ưu tiên bài đăng). Gói “đẩy tin” hoặc “nổi bật” cho người bán. Phí dịch vụ nhỏ cho các giao dịch có hỗ trợ trung gian.

1.4.4. Giá trị nổi bật (Unique Value Proposition)

StudyShare	Khác biệt so với nền tảng hiện tại
Nội bộ sinh viên Bách Khoa (xác thực email @hcmut.edu.vn)	So với Facebook, Chợ Tốt, Oreka – nền tảng công khai, thiếu kiểm duyệt.
Tối ưu cho giao dịch học liệu và dụng cụ học tập	Không bị nhiễu bởi các bài đăng ngoài mục đích học tập.
Chat trực tiếp trong ứng dụng	Không cần qua nền tảng thứ ba, tiết kiệm thời gian.
Tập trung vào trải nghiệm di động đơn giản, thân thiện	Phù hợp với thói quen sử dụng điện thoại của sinh viên.
Phát triển theo hướng cộng đồng học tập	Hướng đến xây dựng văn hóa chia sẻ và hỗ trợ lẫn nhau giữa các khóa.

2 Các tính năng của StudyShare

Dựa trên kết quả nghiên cứu thị trường và phân tích nhu cầu thực tế của sinh viên Trường Đại học Bách Khoa, nhóm xác định rằng một nền tảng chia sẻ học liệu hiệu quả cần đáp ứng đồng thời ba mục tiêu chính:

- hỗ trợ **trao đổi học liệu và dụng cụ học tập thuận tiện**,
- đảm bảo **trải nghiệm người dùng thân thiện**,
- duy trì **môi trường cộng đồng an toàn, đúng đối tượng sinh viên**.

Từ đó, nhóm xây dựng hệ thống tính năng của StudyShare theo các nhóm chức năng rõ ràng, bao gồm tính năng cốt lõi, tính năng trải nghiệm người dùng, bảo mật – cộng đồng và các tính năng mở rộng trong tương lai.

2.1 Danh sách các tính năng chính

Bảng dưới đây tổng hợp các tính năng của StudyShare, được phân loại theo nhóm chức năng và mô tả ngắn gọn mục đích của từng tính năng.

Nhóm chức năng	Tính năng cụ thể	Mô tả ngắn gọn
Cốt lõi – Giao dịch	Đăng tin bán / cho tặng	Người dùng đăng sản phẩm kèm ảnh, mô tả, giá, tình trạng (“Mới”, “Đã sử dụng”).
	Tìm kiếm & lọc sản phẩm	Tìm theo từ khóa, danh mục, giá, khoa, hoặc năm học.
	Chat trực tiếp với người bán	Nhắn tin, trao đổi chi tiết hoặc thương lượng giá.
Trải nghiệm người dùng	Danh mục sản phẩm	Sắp xếp theo nhóm: “Sách – Giáo trình”, “Dụng cụ thí nghiệm”, “Khác”, ...
	Lưu sản phẩm yêu thích	Đánh dấu sản phẩm để xem sau.
	Bình luận dưới bài đăng	Cho phép người dùng bình luận công khai dưới sản phẩm (ví dụ: “sách có ghi chú không?”).
	Đánh giá người bán	Giúp xây dựng uy tín, tránh gian lận.
Bảo mật & cộng đồng	Xác thực sinh viên (email @hcmut.edu.vn)	Đảm bảo người dùng là sinh viên Bách Khoa thật, hạn chế spam và quảng cáo ngoài.
	Hồ sơ cá nhân	Quản lý bài đăng, sản phẩm đã bán hoặc cho tặng, chỉnh sửa thông tin.
	Lọc tin rác	Giúp lọc các bài viết không liên quan, không phù hợp với cộng đồng.
Mở rộng – Giá trị cộng thêm	Thông báo sản phẩm mới	Thông báo khi có bài đăng thuộc danh mục người dùng quan tâm.
	Gợi ý sản phẩm liên quan	Đề xuất dựa trên hành vi tìm kiếm và lượt xem.

Nhìn chung, các tính năng trên được thiết kế nhằm giải quyết trực tiếp các vấn đề tồn tại trong việc trao đổi học liệu hiện nay, đồng thời tạo nền tảng để phát triển một cộng đồng sinh viên trao đổi học tập lành mạnh và hiệu quả.

2.2 Ma trận tác động-nỗ lực (Impact–Effort Matrix)

Để xác định thứ tự ưu tiên phát triển các tính năng, nhóm sử dụng **ma trận Impact–Effort**, trong đó:

- **Impact** thể hiện mức độ tác động của tính năng đến trải nghiệm và giá trị người dùng.
- **Effort** thể hiện mức độ phức tạp và nguồn lực cần thiết để triển khai.

Mục tiêu của phân tích này là lựa chọn các tính năng **mang lại giá trị cao nhất với chi phí phát triển hợp lý**, đặc biệt trong giai đoạn xây dựng MVP.

	Effort thấp	Effort ca��
Impact cao	EASY WINS <ul style="list-style-type: none">Đăng bài (văn bản + ảnh)Tìm kiếm & lọc sản phẩmChat trực tiếp với người bánDanh mục sản phẩmĐánh giá người bánXác thực sinh viên (email @hcmut.edu.vn)	STRATEGICAL WINS <ul style="list-style-type: none">Lọc tin rácBình luận dưới bài đăngHồ sơ cá nhân
Impact thấp	WEAK WINS <ul style="list-style-type: none">Lưu sản phẩm yêu thích	NO WIN <ul style="list-style-type: none">Thông báo sản phẩm mớiGợi ý sản phẩm liên quan

2.2.1. Easy Wins

Tính năng	Giải thích
Đăng tin bán / cho tặng	Là tính năng trung tâm của StudyShare, cho phép sinh viên đăng tài liệu, sách hoặc dụng cụ học tập một cách đơn giản. Dựa trên mô hình CRUD nên dễ phát triển, dễ thử nghiệm và tạo ra giá trị ngay lập tức .
Tìm kiếm & lọc sản phẩm	Giúp người dùng tìm được món phù hợp nhanh nhất , tăng trải nghiệm sử dụng và giữ chân người dùng ngay từ lần đầu.
Danh mục sản phẩm	Tổ chức thông tin khoa học, nâng cao cảm nhận chuyên nghiệp và hỗ trợ tìm kiếm trực quan.
Chat trực tiếp với người bán	Tạo kênh giao tiếp nhanh, riêng tư, tác động mạnh đến trải nghiệm thực tế và tỷ lệ hoàn tất giao dịch.
Đánh giá người bán	Xây dựng lòng tin trong cộng đồng và giảm rủi ro gian lận.
Xác thực sinh viên bằng email @hcmut.edu.vn	Đảm bảo độ tin cậy và tính đặc thù của cộng đồng sinh viên.

2.2.2. Strategical Wins

Tính năng	Giải thích
Bình luận dưới bài đăng	Tăng tính minh bạch và hỗ trợ kiểm chứng thông tin.
Lọc tin rác	Duy trì chất lượng nội dung và môi trường trao đổi lành mạnh.
Hồ sơ cá nhân	Xây dựng danh tính và mức độ uy tín của người dùng.

2.2.3. Weak Wins

Tính năng	Giải thích
Lưu sản phẩm yêu thích	Nâng cao trải nghiệm cá nhân hóa và khuyến khích người dùng quay lại.

2.2.4. No Wins

Tính năng	Giải thích
Thông báo sản phẩm mới	Chưa mang lại giá trị đủ lớn ở giai đoạn MVP.
Gợi ý sản phẩm liên quan	Phụ thuộc dữ liệu hành vi người dùng, chưa cần thiết ban đầu.

Các tính năng thuộc nhóm Easy Wins sẽ được chọn cho phiên bản MVP, vì vừa dễ triển khai vừa tạo ra giá trị cốt lõi cho người dùng.

Những nhóm còn lại sẽ được phát triển ở giai đoạn sau, khi nền tảng đạt đủ lượng người dùng và dữ liệu để mở rộng tính năng.

2.3 Các tính năng trong MVP

Dựa trên phân tích Impact–Effort, nhóm xác định các tính năng cần thiết cho **phiên bản MVP của StudyShare**.

Must Have Features (MVP)	Future Phase
Đăng tin bán / cho tặng	Bình luận dưới bài đăng
Tìm kiếm & lọc sản phẩm	Lọc tin rác
Danh mục sản phẩm	Hồ sơ cá nhân
Chat trực tiếp với người bán	Lưu sản phẩm yêu thích
Đánh giá người bán	Thông báo sản phẩm mới
Xác thực sinh viên bằng email @hcmut.edu.vn	Gợi ý sản phẩm liên quan

Việc lựa chọn các tính năng cho MVP giúp StudyShare tập trung giải quyết bài toán cốt lõi của sinh viên Bách Khoa trong việc trao đổi học liệu và dụng cụ học tập, đồng thời cho phép mở rộng và hoàn thiện nền tảng trong các giai đoạn tiếp theo.

3 Hướng tiếp cận và phương pháp

Để đảm bảo dự án được triển khai có hệ thống, đúng tiến độ và đáp ứng tốt yêu cầu thực tế của người dùng, nhóm áp dụng phương pháp phát triển phần mềm theo hướng **Agile**, kết hợp với các công cụ cộng tác trực tuyến nhằm tối ưu hiệu quả làm việc nhóm và quản lý dự án.

3.1 Quy trình phát triển theo hướng Agile

Dự án được triển khai theo mô hình **làm việc theo milestone**, phù hợp với quy mô nhóm gồm 4 thành viên và thời gian thực hiện giới hạn. Thay vì phát triển toàn bộ hệ thống trong một giai đoạn duy nhất, nhóm chia dự án thành nhiều milestone nhỏ, mỗi milestone tập trung giải quyết một tập chức năng hoặc mục tiêu cụ thể (ví dụ: phân tích yêu cầu, xây dựng MVP, hoàn thiện và tối ưu).

Cách tiếp cận này giúp nhóm:

- Dễ dàng theo dõi tiến độ phát triển của dự án.
- Linh hoạt điều chỉnh phạm vi và ưu tiên tính năng khi có thay đổi về yêu cầu.
- Phát hiện sớm các vấn đề kỹ thuật hoặc thiết kế để kịp thời cải tiến.

Sau mỗi milestone, nhóm tiến hành rà soát lại kết quả đạt được, đánh giá mức độ hoàn thành so với mục tiêu ban đầu và thống nhất kế hoạch cho giai đoạn tiếp theo.

3.2 Công cụ cộng tác và quản lý dự án

Nhóm sử dụng **GitHub** làm nền tảng trung tâm cho việc quản lý mã nguồn và phối hợp phát triển. Cụ thể:

- **GitHub Repository** được dùng để lưu trữ toàn bộ mã nguồn của dự án, đảm bảo tính nhất quán và dễ dàng kiểm soát phiên bản.
- **GitHub Issues** được sử dụng để ghi nhận các công việc cần thực hiện, lỗi phát sinh và đề xuất cải tiến. Mỗi issue được mô tả rõ ràng và gán cho thành viên phụ trách.
- **Pull Request (PR)** được áp dụng trong quá trình phát triển nhằm đảm bảo mã nguồn được rà soát trước khi hợp nhất vào nhánh chính. Điều này giúp giảm lỗi, nâng cao chất lượng code và tăng tính minh bạch trong làm việc nhóm.
- **GitHub Projects** được sử dụng như một bảng quản lý công việc (Kanban), trong đó các issue được sắp xếp theo từng milestone và trạng thái (To Do – In Progress – Done), giúp cả nhóm theo dõi tiến độ một cách trực quan.

Bên cạnh đó, nhóm kết hợp các công cụ giao tiếp để hỗ trợ trao đổi nhanh và họp nhóm:

- **Zalo** được sử dụng cho trao đổi hằng ngày, thảo luận nhanh các vấn đề phát sinh.



- **Google Meet** được dùng cho các buổi họp nhóm định kỳ, thống nhất hướng phát triển, phân công công việc và đánh giá kết quả sau mỗi milestone.

4 Chi tiết hiện thực

4.1 Frontend

Phần Frontend của dự án sử dụng các công nghệ sau:

- **React Native**
- **Expo**
- **TypeScript**
- **Node.js**
- **Expo Application Services (EAS)**
- **NativeWind (Tailwind CSS for React Native)**
- **React Context**
- **React Hooks**
- **Expo Router**
- **Axios**
- **RESTful API**

Frontend của ứng dụng được phát triển bằng **React Native kết hợp với Expo**, cho phép xây dựng ứng dụng di động đa nền tảng (Android và iOS) từ một codebase duy nhất. Việc sử dụng Expo giúp rút ngắn thời gian phát triển, đơn giản hóa quá trình cấu hình native và giảm chi phí thiết lập môi trường, từ đó nhóm có thể tập trung nhiều hơn vào logic nghiệp vụ và trải nghiệm người dùng.

Ứng dụng được viết bằng **TypeScript** trên nền **Node.js**, giúp tăng độ an toàn kiểu dữ liệu, phát hiện lỗi sớm trong quá trình phát triển và giảm thiểu lỗi runtime. TypeScript cũng hỗ trợ refactor hiệu quả và cải thiện khả năng bảo trì mã nguồn trong dài hạn.

React Native đóng vai trò là framework chính, mang lại hiệu năng gần với ứng dụng native và khả năng tái sử dụng code cao. **Expo** cung cấp sẵn các API truy cập thiết bị như camera, lưu trữ, quản lý quyền và networking mà không cần cấu hình native phức tạp, giúp quá trình phát triển diễn ra nhanh chóng và ổn định.

Quá trình build và phân phối ứng dụng được thực hiện thông qua **Expo Application Services (EAS)**. EAS cho phép build file APK trên nền tảng cloud, không phụ thuộc vào môi trường máy local, đồng thời hỗ trợ quản lý cấu hình theo từng môi trường (development, staging, production).

Về giao diện người dùng, ứng dụng sử dụng **NativeWind**, một framework styling dựa trên Tailwind CSS dành cho React Native. Cách tiếp cận utility-first giúp xây dựng giao diện nhanh, nhất quán, dễ tái sử dụng và thuận tiện cho việc mở rộng hệ thống UI.

Về kiến trúc, frontend được tổ chức theo **component-based architecture**, trong đó các màn hình, component tái sử dụng và logic nghiệp vụ được tách biệt rõ ràng. **React Context** được sử dụng để quản lý trạng thái toàn cục như thông tin người dùng và trạng thái đăng nhập, trong khi **React Hooks** xử lý trạng thái cục bộ của giao diện. Việc điều hướng trong ứng dụng được triển khai bằng **Expo Router**, áp dụng mô hình routing theo cấu trúc thư mục, giúp mã nguồn rõ ràng và dễ mở rộng.

Frontend giao tiếp với backend thông qua **RESTful API**, sử dụng **Axios** làm HTTP client. Axios được cấu hình tập trung trong một service layer để thống nhất cách gửi request, xử lý response và quản lý lỗi. Các thông tin cấu hình như API base URL được quản lý bằng biến môi trường của Expo, đảm bảo tính bảo mật và khả năng tách biệt giữa các môi trường triển khai.

4.2 Backend

Phần Backend của dự án sử dụng các công nghệ sau:

- **Python 3.11**
- **FastAPI**
- **PostgreSQL**
- **SQLAlchemy ORM (2.0)**
- **Alembic**
- **JWT Authentication (Access Token & Refresh Token)**
- **Google OAuth**
- **WebSocket**
- **Swagger / OpenAPI**
- **Railway**

Backend của StudyShare được phát triển bằng **FastAPI** trên nền **Python 3.11**, là một framework hiện đại, hiệu năng cao và hỗ trợ bất đồng bộ. FastAPI được lựa chọn nhờ cú pháp rõ ràng, tốc độ xử lý nhanh và khả năng tự động sinh tài liệu API thông qua **OpenAPI/Swagger**, giúp quá trình tích hợp và kiểm thử trở nên thuận tiện.

Hệ thống sử dụng **PostgreSQL** làm hệ quản trị cơ sở dữ liệu quan hệ, kết hợp với **SQLAlchemy ORM (phiên bản 2.0)** để quản lý truy vấn và ánh xạ dữ liệu một cách an toàn và hiệu quả. Việc quản lý phiên bản cơ sở dữ liệu được thực hiện bằng **Alembic**, cho phép tạo, áp dụng và rollback các migration một cách nhât quán giữa các môi trường.

Về bảo mật và xác thực, backend triển khai **JWT Authentication** với cơ chế **Access Token và Refresh Token**. Access Token có thời gian sống ngắn nhằm tăng cường bảo mật cho các request API, trong khi Refresh Token được lưu trữ trong cơ sở dữ liệu để hỗ trợ gia hạn phiên đăng nhập. Bên cạnh đó, hệ thống tích hợp **Google OAuth**, cho



phép người dùng đăng nhập nhanh bằng tài khoản Google, đồng thời backend chịu trách nhiệm xác minh Google ID Token.

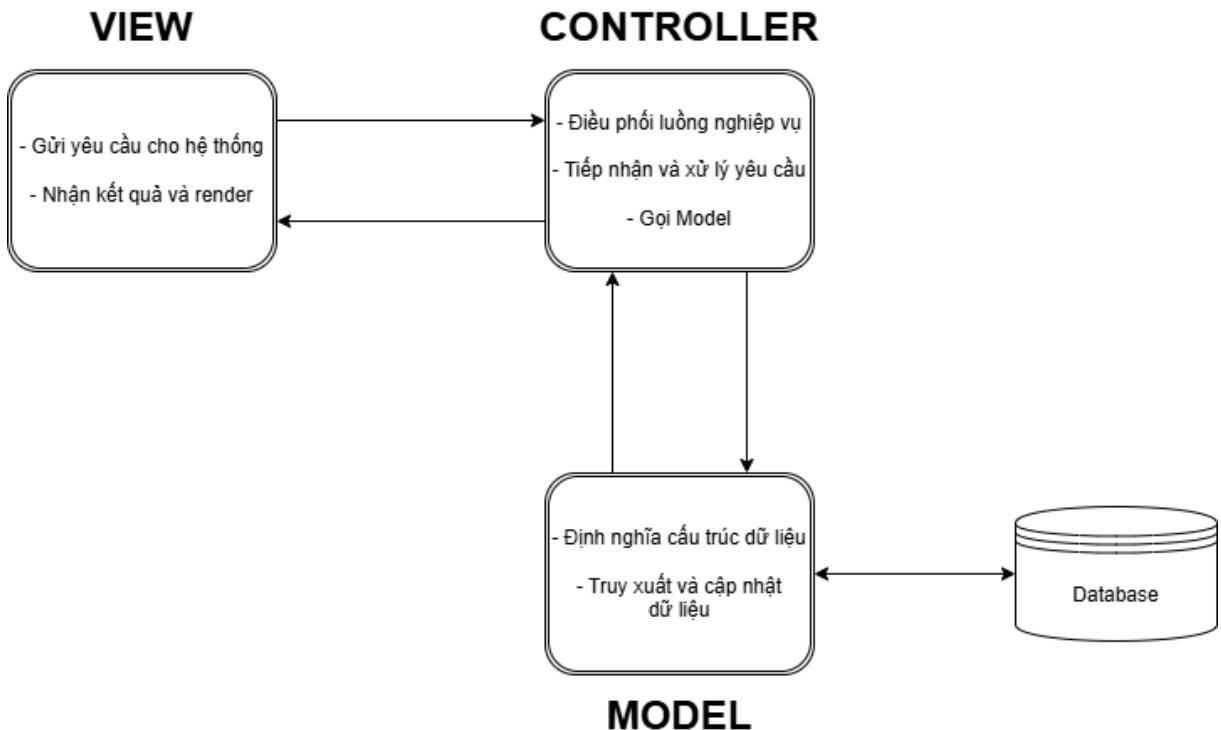
Đối với các tính năng yêu cầu giao tiếp thời gian thực, hệ thống sử dụng **WebSocket** để triển khai chức năng chat giữa người dùng. Giải pháp này cho phép truyền dữ liệu hai chiều với độ trễ thấp, phù hợp với các tương tác nhắn tin trực tiếp trong ứng dụng.

Backend được tổ chức theo **Clean Architecture**, tách biệt rõ ràng giữa các lớp **router – service – model**, giúp mã nguồn dễ đọc, dễ bảo trì và thuận tiện cho việc mở rộng hoặc kiểm thử. Toàn bộ API được version hóa theo chuẩn /api/v1 và tuân thủ thiết kế **RESTful**, đảm bảo tính nhất quán và khả năng mở rộng trong tương lai.

Hệ thống backend được thiết kế sẵn sàng để triển khai trên nền tảng cloud như **Railway**, cho phép quản lý biến môi trường tập trung, tự động deploy và mở rộng tài nguyên theo nhu cầu sử dụng thực tế.

5 Kiến trúc hệ thống

Hệ thống StudyShare được thiết kế theo **kiến trúc Model–View–Controller (MVC)** nhằm đảm bảo tính rõ ràng trong tổ chức mã nguồn, dễ bảo trì và thuận lợi cho việc mở rộng trong tương lai. Kiến trúc này cho phép tách biệt rõ ràng giữa **giao diện người dùng, xử lý nghiệp vụ, và quản lý dữ liệu**, giúp nhóm phát triển song song các thành phần mà không gây phụ thuộc chặt chẽ.



Hình 1: Sơ đồ kiến trúc hệ thống

Ở mức tổng thể, hệ thống bao gồm ba lớp chính:

- **Frontend (View):** chịu trách nhiệm hiển thị giao diện và tiếp nhận tương tác từ người dùng.
- **Backend (Controller):** xử lý logic nghiệp vụ, xác thực, phân quyền và điều phối luồng dữ liệu.
- **Database (Model):** lưu trữ và quản lý dữ liệu của hệ thống.

Các thành phần này giao tiếp với nhau thông qua các giao diện lập trình (API), đảm bảo dữ liệu được truyền tải nhất quán và an toàn.

5.1 Model

Lớp **Model** đại diện cho dữ liệu cốt lõi của hệ thống và các quy tắc liên quan đến dữ liệu. Thành phần này chịu trách nhiệm:



- Định nghĩa cấu trúc dữ liệu cho các thực thể chính như người dùng, bài đăng, sản phẩm, đánh giá và tin nhắn.
- Thực hiện các thao tác truy xuất, thêm, sửa và xóa dữ liệu.
- Đảm bảo tính toàn vẹn và nhất quán của dữ liệu thông qua các ràng buộc và quy tắc nghiệp vụ.

Model được thiết kế độc lập với giao diện người dùng, giúp dữ liệu có thể được tái sử dụng cho nhiều loại giao diện hoặc nền tảng khác nhau trong tương lai.

5.2 View

Lớp **View** đảm nhiệm vai trò hiển thị thông tin và tương tác trực tiếp với người dùng. Trong StudyShare, View bao gồm:

- Các màn hình hiển thị danh sách sản phẩm, chi tiết bài đăng, giao diện tìm kiếm và lọc.
- Các biểu mẫu nhập liệu như đăng tin bán/cho tặng, đăng nhập và xác thực người dùng.
- Các thành phần giao diện hỗ trợ trải nghiệm người dùng như danh mục sản phẩm, trang cá nhân và hệ thống đánh giá.

View chỉ tập trung vào việc trình bày dữ liệu và gửi yêu cầu đến Controller, không xử lý trực tiếp logic nghiệp vụ, từ đó giúp giao diện dễ chỉnh sửa và cải tiến mà không ảnh hưởng đến các phần còn lại của hệ thống.

5.3 Controller

Lớp **Controller** đóng vai trò trung gian giữa View và Model, chịu trách nhiệm:

- Tiếp nhận yêu cầu từ người dùng thông qua giao diện.
- Kiểm tra dữ liệu đầu vào, xác thực người dùng và phân quyền truy cập.
- Thực thi logic nghiệp vụ, xử lý các quy trình như đăng bài, tìm kiếm, nhắn tin và đánh giá.
- Gọi Model để truy xuất hoặc cập nhật dữ liệu và trả kết quả về cho View.

Việc tập trung logic xử lý tại Controller giúp hệ thống dễ kiểm soát luồng xử lý, đồng thời giảm sự phụ thuộc trực tiếp giữa giao diện và dữ liệu.

6 Thiết kế Web Service

Hệ thống backend của **StudyShare** được thiết kế theo mô hình **RESTful Web Service**, đóng vai trò là lớp trung gian giữa ứng dụng di động và cơ sở dữ liệu. Backend chịu trách nhiệm xử lý toàn bộ logic nghiệp vụ cốt lõi như xác thực người dùng, quản lý bài đăng học liệu, nhắn tin thời gian thực, thông báo, và tải tệp hình ảnh.

Toàn bộ API được tổ chức theo chuẩn REST với **API versioning** (/api/v1), sử dụng định dạng dữ liệu **JSON** nhằm đảm bảo tính nhất quán, dễ tích hợp và mở rộng trong tương lai. Các endpoint được phân chia rõ ràng theo từng module chức năng như auth, users, posts, chat, notifications, upload và common.

Thiết kế web service tuân thủ nguyên tắc **SoC (separation of concerns)**, phù hợp với kiến trúc **MVC / Clean Architecture**, trong đó:

- **Router** chịu trách nhiệm định nghĩa endpoint và xử lý request/response,
- **Service layer** xử lý logic nghiệp vụ,
- **Model** quản lý tương tác với cơ sở dữ liệu.

6.1 Authentication & Authorization

Hệ thống xác thực được xây dựng dựa trên **JWT (JSON Web Token)**, kết hợp giữa **Access Token** và **Refresh Token** nhằm cân bằng giữa bảo mật và trải nghiệm người dùng.

- **Access Token** có thời hạn ngắn (15 phút), được sử dụng cho các request cần xác thực.
- **Refresh Token** có thời hạn dài hơn (7 ngày), được lưu trong cơ sở dữ liệu để phục vụ việc cấp lại access token khi hết hạn.

Ngoài hình thức đăng nhập bằng email/mật khẩu, hệ thống còn hỗ trợ **Google OAuth**, trong đó backend đảm nhiệm việc **xác thực Google ID Token** do ứng dụng di động gửi lên. Cách tiếp cận này giúp giảm rào cản đăng ký, đồng thời tăng độ tin cậy của người dùng.

Các endpoint xác thực chính bao gồm:

- Đăng ký (/auth/register)
- Đăng nhập (/auth/login)
- Đăng nhập Google (/auth/google)
- Làm mới token (/auth/refresh)
- Đăng xuất (/auth/logout)



Tất cả các endpoint nhạy cảm đều yêu cầu header Authorization: Bearer <access_token>.

6.2 User Management

Dịch vụ quản lý người dùng cho phép:

- Truy xuất hồ sơ cá nhân của người dùng đang đăng nhập,
- Cập nhật thông tin cá nhân (tên, ảnh đại diện),
- Xem danh sách bài đăng của một người dùng bất kỳ (public endpoint).

Việc tách riêng các endpoint liên quan đến người dùng giúp dễ dàng kiểm soát quyền truy cập và mở rộng trong tương lai (ví dụ: phân quyền, quản lý hồ sơ nâng cao).

6.3 Common Data

Các dữ liệu dùng chung như **khoa**, **môn học**, và **danh mục học liệu** được cung cấp thông qua các endpoint public, không yêu cầu xác thực. Đây là các dữ liệu ít thay đổi, được thiết kế để:

- Giảm tải cho hệ thống bằng cách cho phép client cache,
- Đảm bảo tính nhất quán khi người dùng tạo hoặc lọc bài đăng.

Việc tách nhóm endpoint common giúp hệ thống rõ ràng về mặt trách nhiệm và dễ bảo trì.

6.4 Posts Management

Dịch vụ quản lý bài đăng là **chức năng cốt lõi** của StudyShare, cho phép sinh viên đăng, tìm kiếm và quản lý học liệu.

Các endpoint hỗ trợ đầy đủ các thao tác:

- Tạo bài đăng mới (yêu cầu xác thực),
- Lấy danh sách bài đăng với **pagination** và **bộ lọc nâng cao** (theo môn học, danh mục, trạng thái, từ khóa),
- Xem chi tiết bài đăng (tự động tăng lượt xem),
- Cập nhật và xóa bài đăng (chỉ chủ sở hữu).

Bài đăng không bị xóa vĩnh viễn khỏi hệ thống mà sử dụng **soft delete** thông qua trạng thái HIDDEN, giúp đảm bảo tính toàn vẹn dữ liệu và thuận tiện cho việc phân tích hoặc khôi phục.



6.5 Real-time Chat

Hệ thống chat được thiết kế theo mô hình **kết hợp REST API và WebSocket**:

- REST API dùng để tạo cuộc hội thoại và lấy lịch sử tin nhắn,
- WebSocket dùng cho nhắn tin thời gian thực.

Mỗi cuộc trò chuyện được định danh bằng `conversation_id`, và kết nối WebSocket yêu cầu access token hợp lệ để xác thực người dùng. Thiết kế này đảm bảo:

- Tin nhắn được truyền tải tức thời,
- Lịch sử hội thoại được lưu trữ an toàn trong cơ sở dữ liệu,
- Hỗ trợ mở rộng cho các loại tin nhắn khác (TEXT, IMAGE).

6.6 Notification

Hệ thống thông báo cho phép người dùng:

- Nhận thông báo khi có tin nhắn mới hoặc tương tác liên quan,
- Xem danh sách thông báo với phân trang,
- Đánh dấu thông báo đã đọc hoặc đọc tất cả.

Thiết kế notification tách biệt giúp backend có thể dễ dàng mở rộng sang các loại thông báo khác như hệ thống, tương tác bài đăng hoặc nhắc nhở trong tương lai.

6.7 File Upload

Dịch vụ upload hình ảnh được thiết kế riêng nhằm:

- Giảm độ phức tạp cho các endpoint khác,
- Đảm bảo kiểm soát định dạng và dung lượng tệp.

Quy trình sử dụng gồm hai bước:

1. Upload ảnh để nhận URL công khai,
2. Sử dụng URL này khi tạo bài đăng hoặc cập nhật hồ sơ.

Cách tiếp cận này giúp backend linh hoạt khi chuyển sang các giải pháp lưu trữ đám mây (S3, CDN, v.v.).



6.8 Validation & Error Handling

Tất cả dữ liệu đầu vào đều được validate bằng Pydantic, giúp:

- Phát hiện lỗi sớm,
- Trả về thông báo lỗi rõ ràng, có cấu trúc.

Hệ thống sử dụng **chuẩn HTTP status code**, kết hợp với định dạng lỗi thông nhất (detail) để ứng dụng di động dễ dàng xử lý và hiển thị cho người dùng.

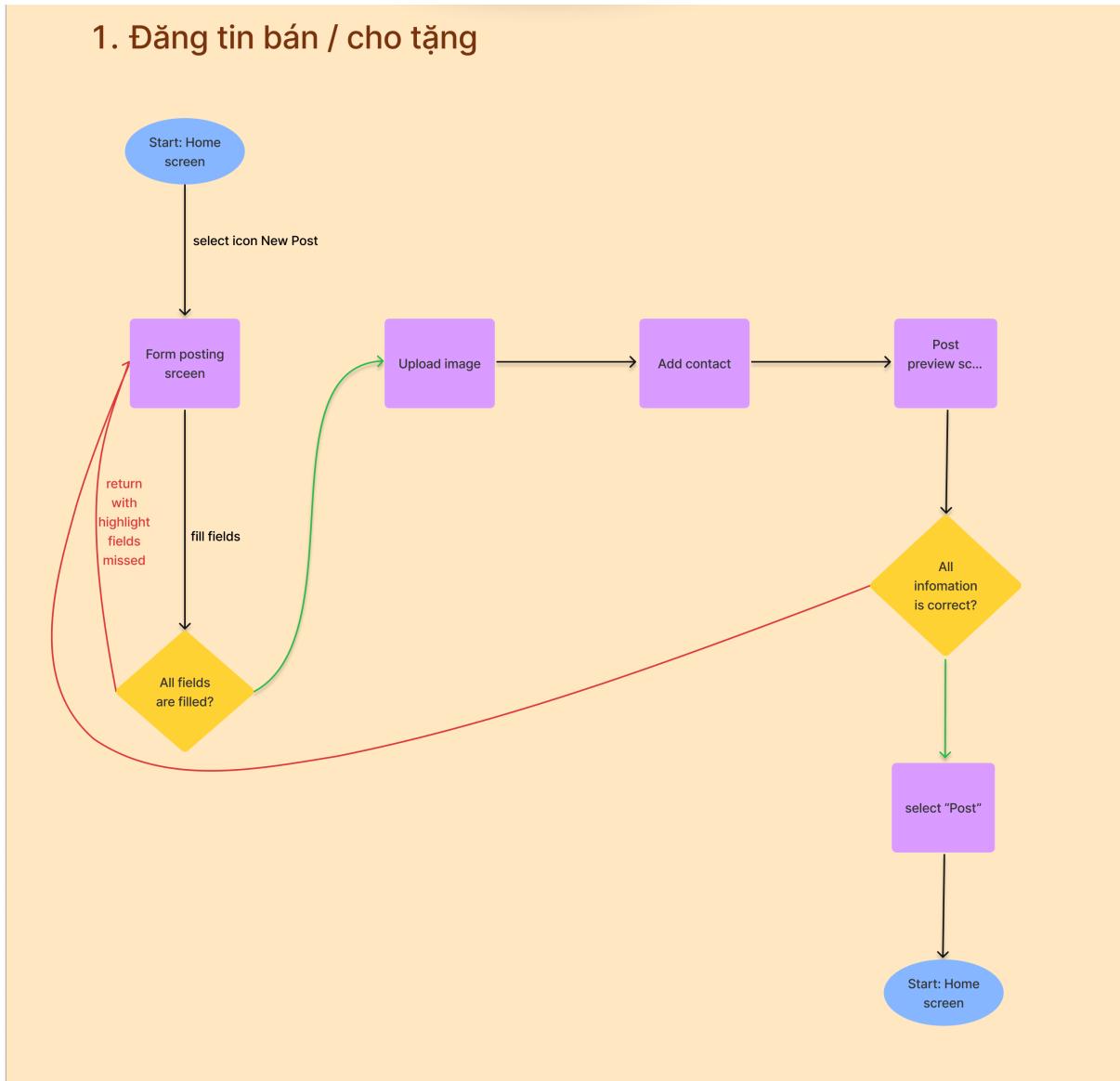
6.9 Security

Các biện pháp bảo mật chính được áp dụng bao gồm:

- JWT cho xác thực và phân quyền,
- Hash mật khẩu an toàn,
- Lưu refresh token trong database để kiểm soát vòng đời đăng nhập,
- Giới hạn quyền truy cập theo vai trò (chủ bài đăng),
- Kiểm soát WebSocket bằng token hợp lệ.

7 Trải nghiệm người dùng

7.1 Đăng tin bán / cho tặng



7.1.1. User Story (nhu cầu sử dụng của người dùng)

"Là một sinh viên Bách Khoa, tôi muốn tạo một bài đăng để cho/tặng, trao đổi hoặc pass lại học liệu, dụng cụ học tập, để các sinh viên khác có thể dễ dàng tìm và tái sử dụng chúng."

7.1.2. Yêu cầu

- Người dùng phải đăng nhập bằng email xác thực @hcmut.edu.vn.
- Người dùng có thể truy cập tính năng “Đăng bài mới” từ trang chủ.
- Mẫu đăng bài bao gồm các trường:



- Tiêu đề
- Nội dung / mô tả chi tiết
- Chọn giá
- Chọn danh mục
- Vị trí (Cơ sở 1, Cơ sở 2, ...)
- Ảnh (chọn từ thư viện thiết bị)
- Thông tin liên hệ (Facebook, số điện thoại, hoặc chat trực tiếp trong app)
- Các trường bắt buộc phải được điền; nếu thiếu sẽ được **tô đỏ và hiển thị thông báo “Must fill”**.
- Sau khi hoàn tất, người dùng có thể xem trước bài đăng (**Post Preview**).
- Ở màn hình xem trước:
 - “Post” để xác nhận đăng bài.
 - “Edit” để quay lại chỉnh sửa.
- Sau khi đăng thành công, hệ thống trở về trang chủ và bài vừa đăng sẽ xuất hiện đầu tiên.
- Có tùy chọn **“Cancel Post”** ở mọi bước để hủy bài.

7.1.3. Luồng hoạt động

1. **Start:** Home screen.
2. Chọn **“New Post”**.
3. Màn hình đăng bài hiện ra với các trường: Tiêu đề, nội dung, loại bài đăng (Cho/tặng, trao đổi, pass), Vị trí (Cơ sở 1, 2,...).
4. Nếu các trường chưa được điền đầy đủ thì các fields đó sẽ được **highlight đỏ và hiển thi “Must fill”**.
5. Sau khi điền đầy đủ các fields → chọn ảnh từ thư viện.
6. Sau khi chọn ảnh → để lại contact (Facebook, số điện thoại) hoặc chọn **liên hệ trực tiếp qua app**.
7. Sau khi hoàn tất → chuyển đến màn hình **“Post preview”** để kiểm tra lại thông tin.
8. Nếu thông tin đã chính xác → chọn **“Post”** → đăng thành công → trở về Home screen → bài vừa đăng xuất hiện đầu tiên.
9. Nếu muốn sửa thông tin → chọn **“Edit”** để quay lại màn hình nhập fields.
10. Ở mọi bước, có thể chọn **“Cancel Post”** để hủy bài đăng.



The first screenshot shows the main product listing screen with categories like 'Sách bài tập', 'Sách giáo trình', 'Slide bài giảng', and 'Sách thí nghiệm'. It also displays a specific product card for 'GIÁO TRÌNH XÁC SUẤT VÀ THỐNG KẾ'.

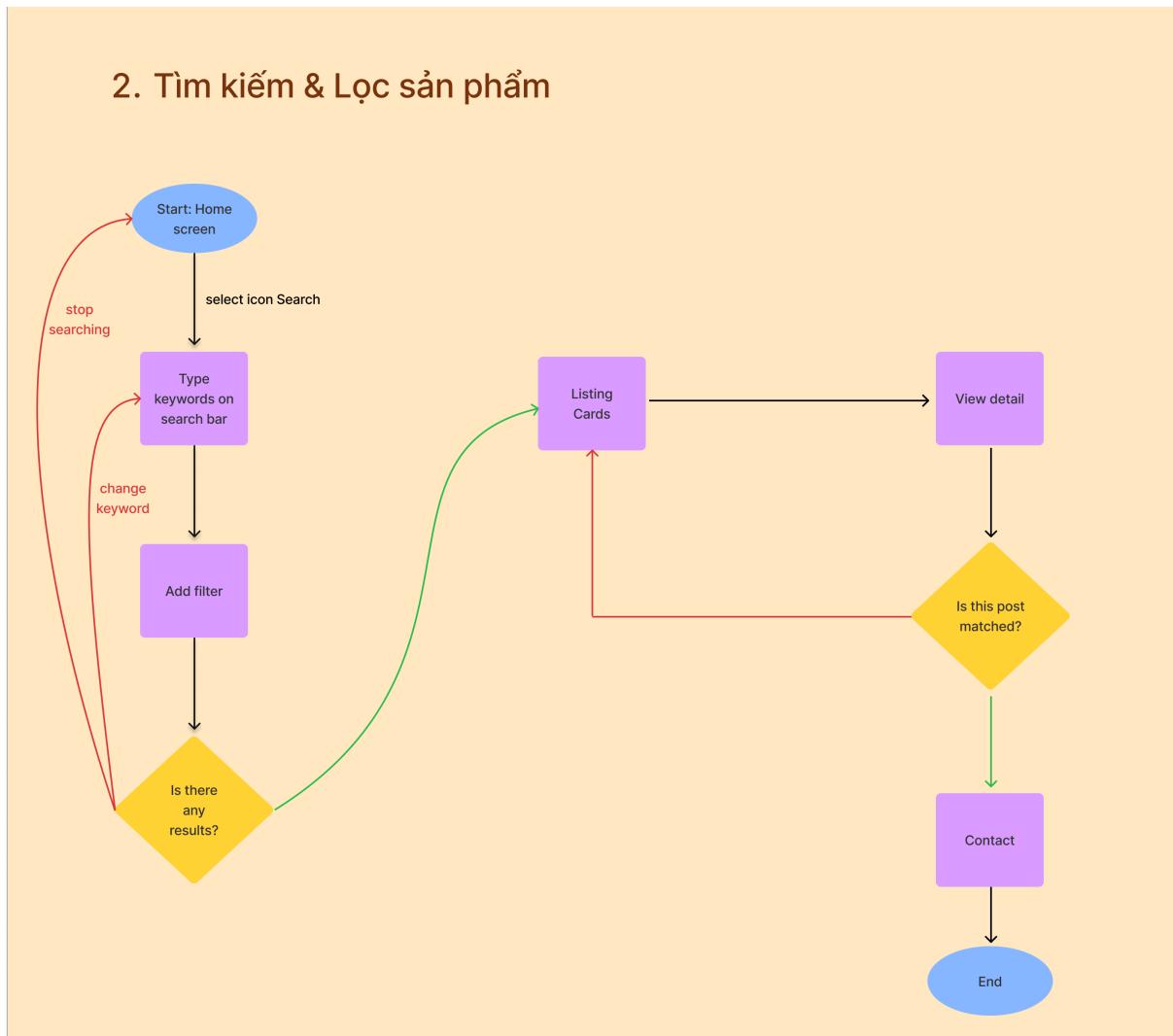
The second screenshot shows the 'Chọn phân loại cho sản phẩm của bạn' (Select product categories) screen. It lists categories under 'Danh mục', 'Khoa', and 'Môn học' with checkboxes.

The third screenshot shows the 'Mô tả cho bài viết' (Description for article) screen. It includes fields for 'Tiêu đề của bài viết' (Article title), 'Tiêu đề nhỏ' (Subtitle), 'Tình trạng' (Status), 'Mô tả' (Description), and 'Giá tiền' (Price).

The first screenshot shows the 'Chọn hình ảnh cho sản phẩm' (Select image for product) screen with a placeholder icon for adding images.

The second screenshot shows the final product listing screen with the uploaded images, detailed description, status, price, and category selection.

7.2 Tìm kiếm và lọc sản phẩm



7.2.1. User Story (nhu cầu sử dụng của người dùng)

"Là một sinh viên đang tìm kiếm học liệu, tôi muốn có thể tìm tài liệu và liên hệ với người đăng bài, để nhanh chóng tìm được học liệu phù hợp và trao đổi an toàn."

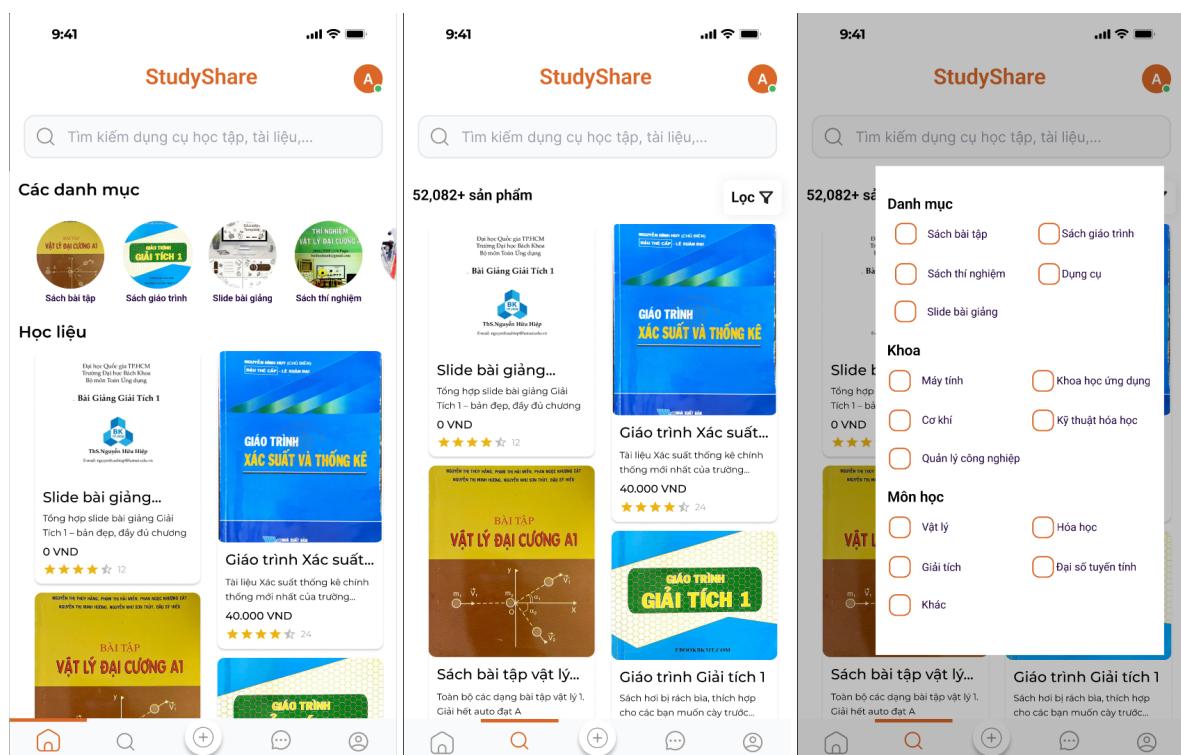
7.2.2. Yêu cầu

- Người dùng phải đăng nhập.
- Có thể nhập từ khóa tìm kiếm trong thanh tìm kiếm.
- Có thể sử dụng các bộ lọc:
 - Loại bài đăng (**Trao đổi / Pass / Cho tặng**)
 - Vị trí (**Cơ sở 1, Cơ sở 2, ...**)
 - Ngành học hoặc môn học (**tùy chọn**)

- Nếu **không có kết quả** → hiển thị thông báo “Không tìm thấy bài đăng phù hợp” và gợi ý thử lại từ khóa khác.
- Nếu **có kết quả** → hiển thị **danh sách bài đăng dạng thẻ** (thumbnail + tiêu đề + loại).
- Người dùng có thể:
 - Nhấn “Xem chi tiết” để xem bài cụ thể.
 - Nhấn “Quay lại” để trở về danh sách.
- Trong trang chi tiết:
 - Có thể **để lại bình luận**.
 - Có thể **liên hệ trực tiếp qua nút “Chat ngay”**.

7.2.3. Luồng hoạt động

1. **Start:** Home screen.
2. Click vào **search bar** và nhập **Keyword** cần tìm.
3. Thêm các **bộ lọc**: Loại bài đăng (swap, pass, free), Vị trí,...
4. Nếu **không có kết quả** → dừng search hoặc đổi keyword khác.
5. Nếu **có kết quả** → chuyển sang màn hình danh sách các **cards**.
6. Từ các cards → chọn “**View detail**”.
7. Có nút “**Back**” để trở lại danh sách cards.
8. Nếu bài phù hợp → có thể **bình luận** hoặc **liên hệ trực tiếp** qua nút “**Chat ngay**”.





The image displays three screenshots of the StudyShare mobile application interface, likely from an Android device, showing course materials and user interactions.

Screenshot 1: Shows a search bar at the top with placeholder text "Tim kiếm dụng cụ học tập, tài liệu...". Below it is a grid of course materials. The first item is "BÀI TẬP VẬT LÝ ĐẠI CƯỜNG AI" (Physics Practice) by "ThS.Nguyễn Hữu Hiệp", priced at 40,000 VND, with a 4-star rating. The second item is "Giáo trình Giải tích 1" (Calculus 1) by "ThS.Nguyễn Hữu Hiệp", priced at 50,000 VND, with a 4-star rating. Other items visible include "Sách Kiến trúc máy..." and "Giáo trình Giải tích 2".

Screenshot 2: Shows a course page for "Bài Giảng Giải Tích 1" (Lesson Plan Calculus 1) by "ThS.Nguyễn Hữu Hiệp". It includes a profile picture of the teacher, contact information (Email: nguyenhuuhiem@hcmut.edu.vn), and a price of 140,000 VND. A message bubble says "Tôi muốn mua lại giải tích 1 ấy".

Screenshot 3: Shows a user profile for "Tui tên Sơn" (My name is Son). It includes a profile picture, contact information (Email: sonnguyentuan123@gmail.com), and a message bubble saying "Bạn có qua cơ sở 1 được không nhỉ?".

8 Chiến lược triển khai

Hệ thống StudyShare được triển khai theo mô hình đơn giản, phù hợp với quy mô dự án học phần và giai đoạn MVP. Backend của ứng dụng được deploy trên nền tảng Railway, một dịch vụ cloud hỗ trợ triển khai nhanh các ứng dụng web thông qua tích hợp trực tiếp với GitHub.

Quy trình deploy backend được thực hiện như sau: nhóm liên kết repository GitHub của dự án với Railway, sau đó chọn branch cần deploy. Các biến môi trường (environment variables) như thông tin kết nối cơ sở dữ liệu, secret key và cấu hình JWT được khai báo trực tiếp trên Railway. Sau khi hoàn tất cấu hình, Railway tự động thực hiện quá trình build và deploy ứng dụng mà không cần can thiệp thủ công thêm. Việc triển khai chỉ cần một thao tác (one-click deploy), giúp tiết kiệm thời gian và giảm độ phức tạp trong quá trình vận hành.

Về mặt CI/CD, hệ thống tận dụng cơ chế tự động build và deploy của Railway thông qua GitHub integration. Mỗi khi có thay đổi được push lên branch đã cấu hình, Railway sẽ tự động kéo mã nguồn mới, build lại ứng dụng và triển khai phiên bản cập nhật. Nhóm không sử dụng pipeline CI/CD tùy chỉnh như GitHub Actions, mà dựa hoàn toàn vào quy trình tự động sẵn có của nền tảng Railway, phù hợp với mục tiêu triển khai nhanh và ổn định cho phiên bản thử nghiệm.

9 Phản hồi người dùng

Nhóm tiến hành khảo sát mức độ hài lòng của người dùng đối với ứng dụng StudyShare thông qua Google Form. Mặc dù số lượng mẫu còn nhỏ, kết quả khảo sát vẫn phản ánh được trải nghiệm ban đầu của người dùng đối với phiên bản hiện tại của ứng dụng. Đa số người tham gia không thường xuyên sử dụng các ứng dụng học tập tương tự, do đó phản hồi thu thập được chủ yếu đến từ góc nhìn của người dùng phổ thông.

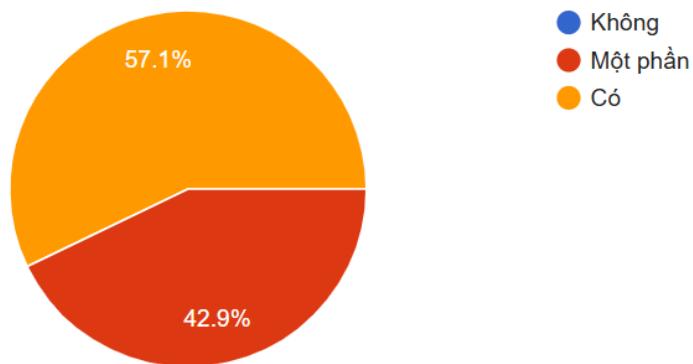
Về khả năng sử dụng, StudyShare nhận được đánh giá khá tích cực. Hơn 85% người dùng đánh giá ứng dụng dễ sử dụng (mức 4–5) và không có phản hồi tiêu cực. Tất cả người tham gia đều cho biết họ không gặp khó khăn khi điều hướng giữa các màn hình, cho thấy luồng thao tác và bố cục giao diện tương đối hợp lý.

D. Đánh giá giá trị thực tế

Theo bạn, ứng dụng này có giải quyết tốt vấn đề ban đầu đặt ra không?

Copy

7 responses



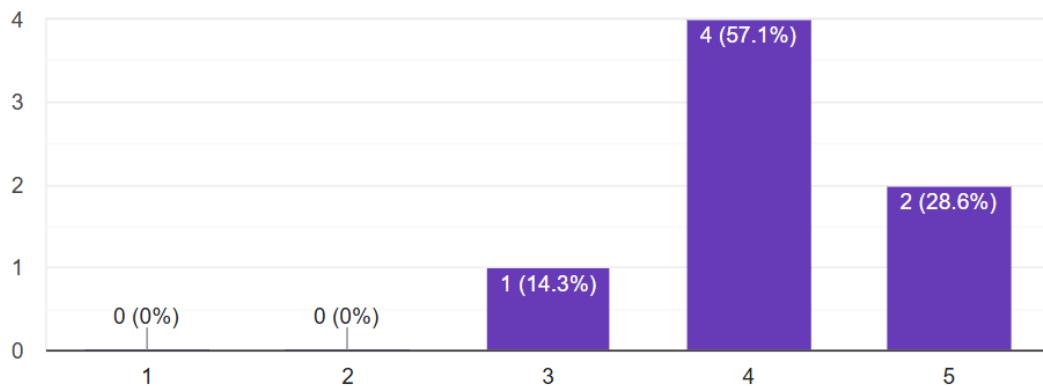
Tuy nhiên, mức độ rõ ràng của giao diện vẫn còn hạn chế. Phần lớn người dùng chỉ đánh giá giao diện ở mức trung bình đến khá, trong khi số người đánh giá “rất rõ ràng” còn ít. Điều này cho thấy giao diện chưa thực sự trực quan và vẫn cần cải thiện để thân thiện hơn với người dùng mới.

C. Mức độ hài lòng

Mức độ hài lòng tổng thể của bạn đối với ứng dụng?

Copy

7 responses



Xét về mức độ hài lòng tổng thể, đa số người tham gia cho biết họ hài lòng với ứng dụng, với hơn 85% đánh giá từ mức 4 trở lên. Tuy vậy, số lượng đánh giá ở mức cao nhất chưa nhiều, cho thấy ứng dụng chưa tạo được ấn tượng đủ mạnh. Tất cả người dùng đều cho biết có khả năng tiếp tục sử dụng StudyShare trong tương lai, nhưng chưa ai khẳng định sẽ gắn bó lâu dài. Điều này cho thấy ứng dụng đang đi đúng hướng, nhưng vẫn cần hoàn thiện thêm để tăng giá trị sử dụng và mức độ gắn bó của người dùng.



10 Tổng kết

10.1 Đánh giá

StudyShare đã hoàn thành đúng mục tiêu ban đầu là xây dựng một **nền tảng chuyên biệt hỗ trợ sinh viên Trường Đại học Bách Khoa chia sẻ và trao đổi học liệu, dụng cụ học tập** một cách tập trung, rõ ràng và hiệu quả hơn so với các giải pháp hiện có trên thị trường.

Về mặt chức năng, phiên bản MVP của hệ thống đã đáp ứng đầy đủ các nhu cầu cốt lõi:

- Đăng và quản lý bài viết trao đổi tài liệu, dụng cụ học tập
- Tìm kiếm và lọc thông tin theo danh mục, môn học
- Giao tiếp trực tiếp giữa người mua – người bán thông qua chat thời gian thực
- Cơ chế đánh giá người bán giúp tăng độ tin cậy trong cộng đồng

Hệ thống backend được thiết kế theo **kiến trúc rõ ràng, có khả năng mở rộng**, tuân thủ các best practices về API design, bảo mật, validation và documentation. Frontend (mobile app) được xây dựng xoay quanh trải nghiệm người dùng sinh viên, tập trung vào thao tác nhanh, dễ hiểu và phù hợp với bối cảnh sử dụng thực tế.

Nhìn chung, StudyShare đã **giải quyết trực tiếp các hạn chế** của những nền tảng như Facebook Group hay các chợ trực tuyến tổng quát, đồng thời tạo ra một giải pháp có **tính ứng dụng thực tiễn cao trong môi trường học đường**.

10.2 Khó khăn

Bên cạnh những kết quả đạt được, nhóm cũng gặp phải một số khó khăn trong quá trình triển khai:

- **Giới hạn về thời gian và nguồn lực:** Với số lượng thành viên hạn chế và thời gian thực hiện có hạn, nhóm buộc phải ưu tiên triển khai MVP thay vì đầy đủ toàn bộ các tính năng mong muốn.
- **Độ phức tạp của các tính năng thời gian thực:** Việc triển khai hệ thống chat real-time bằng WebSocket đòi hỏi xử lý nhiều vấn đề như xác thực kết nối, quản lý trạng thái đọc tin nhắn và đảm bảo hiệu năng.
- **Cân bằng giữa tính năng và trải nghiệm người dùng:** Một số tính năng nâng cao có tiềm năng cải thiện trải nghiệm nhưng chưa phù hợp để đưa vào giai đoạn đầu do chi phí phát triển cao.
- **Thiếu dữ liệu người dùng thực tế:** Do dự án mới ở giai đoạn thử nghiệm, nhóm chưa có đủ dữ liệu lớn để triển khai các tính năng phân tích hành vi hoặc cá nhân hóa sâu.



10.3 Bài học

Through the StudyShare project, the group has extracted many important learning materials:

- Invested in market research to avoid building unnecessary features.
- Concentrated on value proposition to quickly identify strengths and weaknesses.
- Understands the source code and workflow, which greatly affects the quality.
- Good design helps save energy and resources for the future.

10.4 Định hướng

If we continue to develop the next versions, the group will propose several directions:

- **Bổ sung các tính năng thuộc Strategical Wins:**
 - Bình luận dưới bài đăng
 - Hồ sơ cá nhân chi tiết
 - Hệ thống lọc và phát hiện tin rác
- **Cải thiện trải nghiệm người dùng:**
 - Tối ưu giao diện và luồng thao tác
 - Bổ sung trạng thái bài đăng rõ ràng hơn
- **Phát triển tính năng thông minh:**
 - Thông báo sản phẩm mới theo danh mục quan tâm
 - Gợi ý sản phẩm dựa trên hành vi người dùng
- **Tăng cường phân tích và đánh giá hệ thống:**
 - Thu thập dữ liệu hành vi người dùng
 - Kết hợp survey và analytics để cải thiện chất lượng
- **Mở rộng phạm vi sử dụng:**
 - Thử nghiệm triển khai cho các khoa khác
 - Mở rộng cho toàn bộ sinh viên trong trường