

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
PHÂN HIỆU TẠI TP. HỒ CHÍ MINH
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP
ĐỀ TÀI: NGHIÊN CỨU PHÁT TRIỂN ỨNG DỤNG VIẾT
NHẬT KÝ THEO PHONG CÁCH BULLET JOURNAL**

Giảng viên hướng dẫn : NGUYỄN QUANG PHÚC

Sinh viên thực hiện : VÂN THỊ NGÂN HÀ

Lớp : CÔNG NGHỆ THÔNG TIN

Khoá : 58

TP. Hồ Chí Minh, tháng 06 năm 2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
PHÂN HIỆU TẠI TP. HỒ CHÍ MINH
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP
ĐỀ TÀI: NGHIÊN CỨU PHÁT TRIỂN ỨNG DỤNG VIẾT
NHẬT KÝ THEO PHONG CÁCH BULLET JOURNAL**

Giảng viên hướng dẫn : NGUYỄN QUANG PHÚC

Sinh viên thực hiện : VẦN THỊ NGÂN HÀ

Lớp : CÔNG NGHỆ THÔNG TIN

Khoá : 58

TP. Hồ Chí Minh, tháng 06 năm 2021

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP
BỘ MÔN: CÔNG NGHỆ THÔNG TIN

-----***-----

Mã sinh viên: 5851071021

Họ tên SV: Văn Thị Ngân Hà

Khóa: 58

Lớp: Công Nghệ Thông Tin

1. Tên đề tài.

NGHIÊN CỨU PHÁT TRIỂN ỨNG DỤNG VIẾT NHẬT KÝ THEO PHONG CÁCH BULLET JOURNAL.

2. Mục đích, yêu cầu.

a. Mục đích.

- Xây dựng ứng dụng hỗ trợ người dùng như một người bạn đồng hành trong cuộc sống với các chức năng nổi bật: viết nhật ký, viết ghi chú, ghi lại hành trình và hoạt động hằng ngày.
- Ứng dụng hỗ trợ người dùng viết nhật ký, có thể tùy chỉnh thêm hình ảnh, văn bản, biểu tượng cảm xúc và địa chỉ tạo cảm hứng cho người sử dụng, thỏa sức sáng tạo trong những trang nhật ký của riêng mình.
- Xây dựng chức năng ghi chú, giúp người dùng thuận tiện trong việc ghi lại những việc làm cần thiết. Bên cạnh đó người sử dụng có thể tùy chọn thêm hình ảnh và ghi âm.
- Xây dựng chức năng ghi lại hành trình chuyến đi hỗ trợ thêm hình ảnh giúp chuyến đi của bạn trở nên sống động hơn và bạn cũng có thể lập kế hoạch du lịch.
- Xây dựng chức năng ghi lại các hoạt động hàng ngày giúp quản lý công việc một cách tốt hơn.

Yêu cầu.

- Tìm hiểu về ionic framework.
- Tìm hiểu về ngôn ngữ Dart và Flutter để xây dựng ứng dụng.

- Tìm hiểu về cách viết Bullet Journal từ đó xây dựng nên các chức năng.
- Thu thập hình ảnh, thông tin từ các hội nhóm chuyên viết Bullet Journal để hoàn thiện chức năng và phù hợp với nhu cầu thực tế.
- Tìm hiểu về cách thêm hình ảnh, văn bản và âm thanh, sử dụng thư viện lấy dữ liệu từ Google Map API nhận về địa chỉ hiện tại của người dùng.
- Nghiên cứu cách di chuyển các widget trong màn hình và lưu lại vị trí sau khi di chuyển.
- Nghiên cứu cách thêm bút tọa độ hỗ trợ người dùng tùy chỉnh cuộn lên hoặc cuộn xuống cho phép người dùng có nhiều không gian hơn, thoải mái trong việc sáng tạo trang nhật ký của mình.

3. Nội dung và phạm vi đề tài.

a. Nội dung đề tài.

- Giới thiệu về phong cách viết Bullet Journal.
- Nghiên cứu, phân tích phương pháp viết Bullet Journal từ đó phát triển các chức năng cho ứng dụng.
- Nghiên cứu cách xử lý hình ảnh và văn bản cho phép người dùng tùy chỉnh bố cục bài viết theo sở thích của mỗi người.
- Hoàn thiện sản phẩm, đáp ứng các chức năng đã khảo sát.

b. Phạm vi đề tài.

- Nguyên cứu các thẻ giao diện cho phép người dùng di chuyển, tùy chỉnh và xóa.
- Ứng dụng kiến thức đã tìm hiểu để xây dựng ứng dụng viết nhật ký theo phong

cách Bullet Journal.

4. Công nghệ, công cụ và ngôn ngữ lập trình.

a. Công nghệ: Ionic framework, Google Map API

b. Công cụ: Visual Code, Android Studio

c. Ngôn ngữ lập trình: Dart

5. Các kết quả chính dự kiến sẽ đạt được và ứng dụng

- Hoàn chỉnh cuốn báo cáo đề tài.
- Nắm rõ về cấu trúc ngôn ngữ Dart, cách xây dựng một ứng dụng hoàn chỉnh bằng Flutter và thành thạo các thư viện trong Flutter, tiêu biểu là thư viện xử lý về hình ảnh.

- Nắm được qui trình xây dựng một ứng dụng di động từ giai đoạn phân tích yêu cầu đến khi ứng dụng hoàn thành.
- Nắm được các áp dụng thư viện vào dự án, bên cạnh đó tùy chỉnh thư viện để phù hợp với các chức năng và giao diện của ứng dụng.
- Sử dụng Flutter để xây dựng được ứng dụng viết nhật ký theo phong cách Bullet Journal.

6. Giáo viên và cán bộ hướng dẫn

Họ tên: NGUYỄN QUANG PHÚC

Đơn vị công tác: Trường Đại học Kinh tế - Luật, ĐHQG Tp.HCM

Điện thoại: 0777207700

Email: phucnq.edu@gmail.com

Ngày tháng 6 năm 2021

BM Công Nghệ Thông Tin

Đã giao nhiệm vụ TKTN

Giáo viên hướng dẫn

Nguyễn Quang Phúc

Đã nhận nhiệm vụ TKTN

Sinh viên: Văn Thị Ngân Hà

Điện thoại: 0775881084

Ký tên:

Email: nganha2169@gmail.com

LỜI CẢM ƠN

Lời nói đầu tiên, em xin gửi lời cảm ơn tới Quý thầy cô trong Bộ môn Công Nghệ Thông Tin và Ban Giám Hiệu trường Đại học Giao thông Vận tải phân hiệu tại Thành phố Hồ Chí Minh, đã tạo điều kiện giúp em được nhận và thực hiện đồ án của mình.

Có rất nhiều yếu tố để em có thể hoàn thành đồ án. Yếu tố đầu tiên chính là sự cố gắng tìm tòi, học hỏi của bản thân. Tiếp theo, em xin gửi lời cảm ơn chân thành nhất đến thầy **Nguyễn Quang Phúc** – người trực tiếp hướng dẫn em, thầy luôn tận tình hướng dẫn, định hướng, giải đáp thắc mắc và giải quyết những khó khăn trong quá trình thực hiện đồ án. Sự hướng dẫn của thầy chính là yếu tố quan trọng nhất giúp em hoàn thành đề tài của mình. Cuối cùng, em xin một lần nữa gửi lời cảm ơn đến Quý thầy cô trong Bộ môn Công Nghệ Thông Tin đã dạy em những kiến thức nền tảng để em có đủ khả năng hoàn thành đồ án.

Mặc dù đã nỗ lực hết sức để hoàn thành đề tài, nhưng với năng lực và thời gian có hạn, chắc chắn sẽ khó tránh khỏi những sai sót. Em rất mong nhận được những sự đánh giá, góp ý khách quan nhất của Quý thầy cô, giúp em rút ra cho mình những bài học, kinh nghiệm quý báu sau này.

Cuối lời, em cũng không biết nói gì hơn ngoài kính chúc Quý thầy cô trong Bộ môn Công Nghệ Thông Tin và đặc biệt là thầy **Nguyễn Quang Phúc** thật dồi dào sức khỏe và ngày càng gặt hái được nhiều thành công hơn nữa trong cuộc sống cũng như trong sự nghiệp giảng dạy của mình.

Em xin chân thành cảm ơn !

TP. Hồ Chí Minh, ngày tháng năm 2021

Sinh viên thực hiện

Văn Thị Ngân Hà

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày tháng năm 2021

Giáo viên hướng dẫn

Nguyễn Quang Phúc

MỤC LỤC

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP	i
LỜI CẢM ƠN	iv
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	v
MỤC LỤC	vi
DANH MỤC THUẬT NGỮ	ix
DANH MỤC HÌNH ẢNH	x
CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI	1
1.1 Đặt vấn đề	1
1.2 Tình hình nghiên cứu	2
1.3 Quá trình nghiên cứu	3
1.4 Cấu trúc báo cáo đồ án tốt nghiệp	4
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	5
2.1 Giới thiệu về Ionic Framework.	5
2.1.1 Khái niệm	5
2.1.2 Ưu điểm	6
2.1.3 Nhược điểm	6
2.2 Giới thiệu về Flutter	7
2.2.1 Khái niệm	7
2.2.2 Ngôn lập trình – Dart	8
2.2.3 Ưu điểm	9
2.2.4 Nhược điểm	10
2.3 Kiến trúc phần mềm	10
2.3.1 Kiến trúc phần mềm là gì?	10

2.3.2 Mẫu kiến trúc	11
2.3.3 Mô hình kiến trúc Model – View – ViewModel (MVVM)	12
1.3.4 Bloc Pattern trong Flutter	13
1.4 Giới thiệu về Bullet Journal	15
CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	17
3.1 Sơ đồ usecase của hệ thống	17
3.2 Sơ đồ phân rã chức năng.	18
3.2.1 Sơ đồ phân rã chức năng tổng quát.	18
3.2.2 Mô tả chức năng.	19
3.3 Sơ đồ hoạt động	21
3.3.1 Chức năng quản lí nhật ký.	21
3.3.2 Chức năng quản lí ghi chú.	26
2.3.3 Chức năng quản lí chuyến đi.	30
2.2.4 Chức năng đăng nhập.	33
CHƯƠNG 4. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH	34
4.1 Hiện trạng	34
4.2 Yêu cầu hệ thống	34
4.2.1 Yêu cầu chức năng	34
4.2.2 Yêu cầu phi chức năng	34
4.3 Xây dựng giao diện ứng dụng	35
4.3.1 Màn hình đăng nhập	35
4.3.2 Màn hình quản lí nhật ký.	36
4.3.3 Màn hình chỉnh sửa nhật ký.....	38
4.3.4 Màn hình quản lí ghi chú	42
4.3.5 Màn hình chỉnh sửa ghi chú.....	43
4.3.6 Màn hình quản lí chuyến đi	44

KẾT LUẬN	47
Kết quả đạt được.....	47
Nhược điểm.....	47
Hướng phát triển.....	47
TÀI LIỆU THAM KHẢO.....	48

DANH MỤC THUẬT NGỮ

STT	THUẬT NGỮ	Ý NGHĨA TIẾNG VIỆT	TỪ VIẾT TẮT	GHI CHÚ
1	Information Technology	Công Nghệ Thông Tin	CNTT	
2	User Interface	Giao diện người dùng	UI	
3	User Experience	Trải nghiệm người dùng	UX	
4	Software architecture	Kiến trúc phần mềm		
5	Architecture patterns	Mẫu kiến trúc		
6	Cross-platform	Đa nền tảng		
7	Hybrid app	Ứng dụng đa nền tảng		
8	Native app	Ứng dụng gốc		
9	Component	Thành phần		
10	Code-base	Mã lập trình do nhà phát triển viết		
11	Developer	Nhà phát triển (phần mềm)		

DANH MỤC HÌNH ẢNH

Hình 2.1 Mô phỏng công nghệ Ionic ¹	5
Hình 2.2 Giới thiệu về Flutter.	7
Hình 2.3 Software architecture.....	11
Hình 2.4 Mô hình MVVM.....	12
Hình 2.5 Bloc Parttern.	14
Hình 2.6 Bloc Parttern for Flutter	14
Hình 2.7 Ví dụ về Bullet Journal ⁷	15
Hình 3.1 Sơ đồ usecase tổng quát	17
Hình 3.2 Sơ đồ phân rã chức năng tổng quát.....	18
Hình 3.3 Chức năng thêm nhật ký.....	21
Hình 3.4 Chức năng chỉnh sửa nhật ký.....	22
Hình 3.5 Chức năng xóa nhật ký.....	23
Hình 3.6 Chức năng thêm hình ảnh.....	24
Hình 3.7 Chức năng chỉnh sửa hình ảnh.....	25
Hình 3.8 Chức năng thêm ghi chú.....	26
Hình 3.9 Chức năng chỉnh sửa ghi chú.....	27
Hình 3.10 Chức năng xóa ghi chú.	28
Hình 3.11 Chức năng thêm tệp ghi âm.	29
Hình 3.12 Chức năng quản lí chuyến đi.	30
Hình 3.13 Chức năng chỉnh sửa chuyến đi.	31
Hình 3.14 Chức năng xóa chuyến đi.	32
Hình 3.15 Chức năng đăng nhập.	33
Hình 4.1 Màn hình đăng nhập.....	35
Hình 4.2 Màn hình quản lí nhật kí.....	36
Hình 4.3 Màn hình chỉnh sửa nhật kí.	38
Hình 4.4 Màn hình thêm hình ảnh.....	39
Hình 4.5 Giao diện thêm biểu tượng cảm xúc	40
Hình 4.6 Giao diện thêm văn bản.....	40
Hình 4.7 Giao diện thay đổi phông chữ, màu sắc.	41
Hình 4.8 Màn hình quản lý ghi chú.....	42
Hình 4.9 Màn hình chỉnh sửa ghi chú.....	43

Hình 4.10 Màn hình quản lý chuyến đi.	44
Hình 4.11 Màn hình chỉnh sửa chuyến đi.	45
Hình 4.12 Giao diện chỉnh sửa lịch trình.	46
Hình 4.13 Cửa sổ chọn ngày.	46

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI

1.1 Đặt vấn đề

Trong cuộc sống hiện đại và tập nập, mỗi người trong chúng ta ai cũng có một mối bận tâm cho riêng mình. Phải chăng là nỗi lo về cơm áo gạo tiền của các bậc cha mẹ, cũng có thể là những bản khoản của các em học sinh cuối cấp sắp phải rời xa mái trường thân thuộc và đối mặt với kì thi quan trọng quyết định cuộc đời của các em,... Và đặc biệt nỗi lo chung của toàn dân Việt Nam hiện nay chính là sự hoành hành của Covid-19, dịch bệnh đã làm thay đổi đi cách sống của mỗi người. Bỏ qua những ảnh hưởng tiêu cực từ Covid-19, thời gian này cho phép người ta có thêm thời gian gần gũi với gia đình và suy ngẫm về những việc đã làm cũng như những dự định trong tương lai. Nhưng cho mình hỏi, đã có bao giờ bạn chia sẻ hết những nỗi lo trong lòng cho một ai đó? Theo kinh nghiệm sống 20 năm ít ỏi của mình cùng với những kiến thức mình đã đúc kết trong sách vở thì sự thật là bạn không thể chia sẻ hết những điều mình suy nghĩ cho người khác. Có thể là vì thời điểm đó chưa phù hợp hoặc người bạn muốn chia sẻ không hiểu hoàn cảnh của bạn hay một điều gì đó rất nghiêm trọng và bạn không muốn cho người thân biết. Từ những lý do trên, mình đã nghĩ ra ý tưởng xây dựng lên một ứng dụng như một người bạn đồng hành có thể chia sẻ những kỉ niệm buồn vui cùng bạn và những bí mật mà bạn chỉ muốn giữ cho riêng mình.

Khi mình đưa ra ý tưởng này, chắc chắn một điều rằng nhiều bạn sẽ thắc mắc liệu ứng dụng này có cần thiết không? Những ai sẽ muốn sử dụng nó? Và có chắc rằng nhiều người sẽ thích viết nhật ký hay không? Đầu tiên, mình xin đưa ra lý do vì sao chúng ta nên tạo thói quen viết nhật ký. Nhật ký không chỉ là nơi để bạn lưu lại ký ức mà còn đem lại rất nhiều lợi ích khác – những cuốn nhật ký giúp nuôi dưỡng sự sáng tạo và tâm hồn bạn. Khi thói quen viết nhật ký được hình thành, kỹ năng viết lách của bạn sẽ ngày càng tốt hơn. Bên cạnh đó, viết nhật ký mỗi ngày còn giúp chúng ta cải thiện trí nhớ hiệu quả – theo báo cáo của tạp chí Forbes. Mỗi khi viết nhật kí, bạn phải sâu chuỗi lại những sự kiện trong ngày và sắp xếp thời gian một cách logic vì vậy nó sẽ giúp ích rất nhiều cho não bộ của bạn. Trong nghiên cứu của trường Đại học Michigan công bố vào năm 2017 cho rằng viết nhật ký có thể giảm thiểu lo âu. Theo nghiên cứu, viết nhật ký có thể giảm cảm giác lo lắng cho những ai đang bản khoản hay có khuynh hướng quan trọng hóa vấn đề. Tiến sĩ Jason Moser, giáo sư tâm lý học và Giám đốc Phòng thí nghiệm

sinh lý học lâm sàng của Đại học bang Michigan cho biết: *“Viết ra cảm xúc bên trong có thể làm cho tâm trí không còn nghĩ về những việc làm bạn căng thẳng, khiến sự lo lắng bị đốt cháy. Kỹ thuật này giúp “đánh lạc hướng” bộ não, giúp nó làm việc tốt hơn”*.

Mỗi người trong chúng ta, sống luôn cần sự sẻ chia và thấu hiểu từ người khác. Mình lấy ví dụ, mỗi ngày các bạn đăng trạng thái lên các trang mạng xã hội (như Facebook, Instagram), có phải chỉ để xem ai đã yêu thích trạng thái của bạn, ai quan tâm đến bạn? Nhưng phải chăng những gì bạn đăng không thật sự đúng với những cảm xúc trong bạn? Một ngày thật buồn nhưng bạn lại chia sẻ những hình ảnh vui nhộn, có lẽ vì không muốn mang những suy nghĩ tiêu cực đến với mọi người. Đúng vậy, hầu hết mỗi người đều muốn chia sẻ những câu chuyện của mình cho người quan trọng nhất, tuy nhiên vẫn sẽ có những câu chuyện chỉ có thể giữ cho riêng bạn và không ai có thể hiểu bạn bằng chính bạn. Vì vậy việc sắm cho mình một quyển nhật ký là rất cần thiết. Bạn có thể thoải mái chia sẻ những gì mình nghĩ, những dự định mà bạn mong muốn trong tương lai, nếu như trước kia bạn chỉ để trong lòng thì giờ đây bạn có thể viết hết tâm tư của mình ra giấy và quyển nhật ký như một người bạn của bạn. Tuy nhiên với nhịp sống hối hả hiện nay, đa phần mọi người không thể dành nhiều thời gian để viết nhật ký theo cách truyền thống vậy nên một ứng dụng viết nhật ký là một giải pháp tuyệt vời cho những người hiện đại.

Từ những lý do trên, mình đã quyết định chọn đề tài ***“Xây dựng ứng dụng viết nhật ký theo phong cách Bullet Journal”*** cho đồ án tốt nghiệp của mình. Tại sao lại là phong cách Bullet Journal? Đây là gì? Có thể hiểu nôm na, Bullet Journal là một phong cách viết hiện đại, sử dụng nhiều hình ảnh, nhãn dán và kí tự làm cho trang nhật ký của bạn trở nên độc đáo và phá cách hơn. Hơn nữa, bạn có thể thỏa sức sáng tạo phong cách viết cho riêng mình. Ở phần này, mình chỉ giới thiệu sơ lược về đề tài, nội dung chi tiết sẽ được trình bày ở các chương sau.

1.2 Tình hình nghiên cứu

Trên thị trường ứng dụng hiện nay, có rất nhiều ứng dụng hỗ trợ viết nhật kí, ghi chú với số lượng người dùng rất lớn. Mặc dù đã có ứng dụng hệ thống dành riêng cho ghi chú tuy nhiên các ứng dụng này vẫn được nhiều người ưa chuộng. Bằng chứng là ứng dụng ColorNote – đứng đầu các ứng dụng ghi chú, nhận được hơn 100 triệu lượt tải về trên Google Play. Điều này nói lên rằng trải nghiệm về tính năng, giao diện và hiệu

năng là thứ được người dùng quan tâm hơn, không đơn thuần chỉ là chức năng ghi chú thông thường. Vì vậy mình đã giành thời gian nghiên cứu rất nhiều ứng dụng để đưa ra hướng phát triển cho ứng dụng của mình, bên cạnh đó mình cũng học hỏi được những điểm hay và những góp ý từ người dùng của họ về ứng dụng.

1.3 Quá trình nghiên cứu

Trong quá trình nghiên cứu, mình nhận thấy mỗi ứng dụng viết nhật ký hoặc ghi chú đều mang một phong cách riêng và có nhiều chức năng hay thú vị phù hợp với nhiều đối tượng khác nhau. Tuy nhiên đa phần các ứng dụng là riêng lẻ, khi người dùng muốn sử dụng phải tải rất nhiều ứng dụng khác nhau. Do đó, mình mong muốn xây dựng ứng dụng của mình là một ứng dụng tất cả trong một (all-in-one) với các chức năng chính là quản lý nhật ký, quản lý ghi chú và quản lý chuyến đi. Chức năng nổi bật trong ứng dụng là viết nhật ký theo phong cách Bullet Journal, một phong cách viết rất khác với những ứng dụng viết nhật ký thông thường. Nhật ký sẽ không còn đơn điệu, nhàm chán thay vào đó là những hình ảnh, nhãn dán, bố cục được trang trí đẹp mắt.

Để hiểu rõ hơn về Bullet Journal, mình đã tham gia vào nhóm “Cộng đồng Bullet Journal Việt Nam” trên Facebook – một cộng đồng rất lớn với hơn 100.000 thành viên, những người yêu thích Bullet Journal. Thành viên trong nhóm chia sẻ nhiều phong cách viết khác nhau, sử dụng cho nhiều mục đích như viết nhật ký, ghi chú, lập thời gian biểu cho công việc hằng ngày, lập kế hoạch cho chuyến đi,... Cách viết phổ biến nhất là trước tiên, sẽ phân bố cục cho bài viết (Viết chữ ở đâu? Hình ảnh để đâu?), đảm bảo bố cục phải cân đối và dễ nhìn. Tiếp theo viết theo bố cục đã sắp xếp, có thể viết chữ theo phong cách riêng, hoặc sử dụng màu chữ khác để làm nổi bật trang nhật ký. Cuối cùng, thêm các nhãn dán ở những chỗ trống hoặc những nơi cần thu hút sự chú ý. Từ đây, mình đã đưa ra được những chức năng cụ thể cần có trong ứng dụng có thể đáp ứng được phần nào nhu cầu thực tế của người dùng cho một ứng dụng viết theo phong cách Bullet Journal. Các chức năng bao gồm:

- Chức năng viết nhật ký: có thể tùy chỉnh thêm hình ảnh, văn bản, biểu tượng cảm xúc và địa chỉ. Người dùng có thể di chuyển các thành phần này theo bố cục mà mình mong muốn.
- Chức năng ghi chú: người sử dụng cũng có thể tùy chọn thêm hình ảnh và ghi âm.

- Chức năng ghi lại hành trình chuyển đi: có thể dùng chức năng này để lập kế hoạch du lịch hoặc lưu lại các chuyến đi đã qua.
- Chức năng ghi lại các hoạt động hàng ngày.

1.4 Cấu trúc báo cáo đồ án tốt nghiệp

Cấu trúc đồ án được chia thành các chương như sau:

Chương 1: Tổng quan đề tài – Giới thiệu tổng quan về đề tài đồ án tốt nghiệp.

Chương 2: Cơ sở lý thuyết

- Giới thiệu về Ionic Framework
- Giới thiệu về Flutter
- Kiến trúc phần mềm
- Giới thiệu về Bullet Journal

Chương 3: Phân tích và thiết kế hệ thống

- Sơ đồ usecase của hệ thống.
- Sơ đồ phân rã chức năng.
- Sơ đồ hoạt động.

Chương 4: Thiết kế và xây dựng ứng dụng

- Hiện trạng.
- Phân tích yêu cầu hệ thống.
- Xây dựng giao diện ứng dụng

Kết luận

- Đưa ra kết quả đạt được, những thứ còn tồn tại và hướng phát triển các chức năng thực tế, hữu ích cho ứng dụng trong tương lai.

Tài liệu tham khảo.

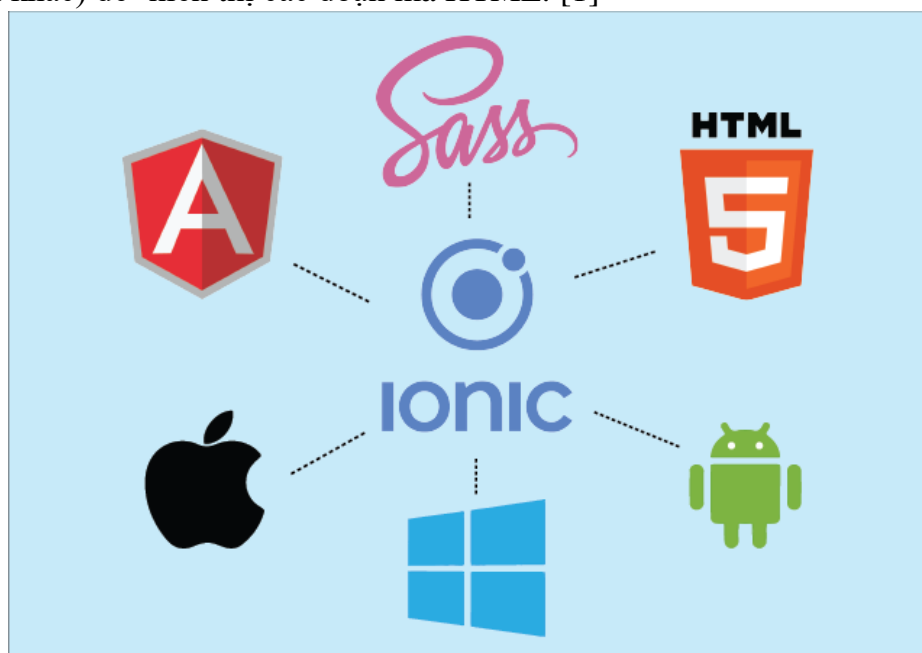
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu về Ionic Framework.

Hiện nay theo thống kê thế giới có khoảng hơn 2.3 tỉ điện thoại thông minh với hai hệ điều hành phổ biến nhất là iOS và Android (hoặc các hệ điều hành tùy biến từ hệ điều hành Android). Doanh thu từ ứng dụng trên hai hệ điều hành iOS và Android xấp xỉ 61 tỉ USD. Vì vậy, hệ sinh thái di động trở thành thị trường màu mỡ mà không một công ty hay các nhà phát triển nào muốn bỏ qua. Việc phát triển một ứng dụng di động cần phải cân nhắc nhiều tiêu chí như (thời gian, chi phí), do đó nhiều doanh nghiệp đã lựa chọn Ionic framework như một giải pháp tối ưu để phát triển ứng dụng của họ.

2.1.1 Khái niệm

Ionic là một framework dùng để phát triển ứng dụng hybrid dựa trên HTML5. Một ứng dụng hybrid là một ứng dụng di động được xây dựng bằng các công nghệ phát triển web như HTML5, CSS, JavaScript và sau đó được đóng gói trong một môi trường hiển thị nội dung có thể hoạt động trên hầu hết các thiết bị di động giống như trình duyệt web (Native container). Các ứng dụng này sử dụng các thành phần hiển thị nội dung website (các trình duyệt ẩn như UIWebView trên iOS, hay Webview trên Android và các thiết bị khác) để ¹hiển thị các đoạn mã HTML. [1]



Hình 2.1 Mô phỏng công nghệ Ionic¹

¹Nguồn ảnh - https://fuzzycloud.in/blog/why_ionic/

2.1.2 Ưu điểm

Ứng dụng hybrid đem lại nhiều lợi thế như khả năng hiển thị nội dung trên tất cả các thiết bị di động, cũng như tận dụng tối đa các tính năng khác của thiết bị di động như GPS, camera, danh sách liên lạc,... Bên cạnh đó, thời gian và chi phí dùng để tạo nên một ứng dụng hybrid cũng thấp hơn so với các ứng dụng di động thông thường.

Bạn có thể coi Ionic như là một bộ khung front-end giúp bạn kiểm soát hình ảnh và trải nghiệm trên ứng dụng của bạn. Giống như “Bootstrap for Native”, nhưng với sự hỗ trợ của một lượng lớn các thành phần di động, hiệu ứng chuyển động mượt mà và thiết kế đẹp. Không như các framework khác, Ionic mang đến cho chúng ta những thành phần giao diện và cách bố trí mang phong cách rất giống với như phong cách thiết kế mặc định trên thiết bị di động.

Lợi ích đối với doanh nghiệp

- Ionic Frameworks sẽ giúp doanh nghiệp cắt giảm chi phí sản xuất. Thay vì phải thuê hai đội ngũ phát triển trên nền tảng Android và IOS doanh nghiệp chỉ cần một nhóm phát triển ionic app.
- Một lợi thế khác cho các chủ doanh nghiệp là hiệu quả về thời gian. Nếu doanh nghiệp đang cần một ứng dụng khẩn cấp, có thể xem xét Ionic Framework. Các ứng dụng được xây dựng trên nó không tốn nhiều thời gian.

2.1.3 Nhược điểm

Hiệu suất của ứng dụng Ionic không tốt bằng so với các ứng dụng di động gốc. Tuy nhiên, khoảng cách hiệu suất là không đáng chú ý đối với hầu hết người dùng bình thường. [2]

Ứng dụng được phát triển bằng Ionic framework có thể không cung cấp nhiều bảo mật như các ứng dụng gốc. Do đó ionic app không phù hợp với các ứng dụng đòi hỏi tính bảo mật cao như tài chính, ngân hàng,...

Có một số chức năng gốc có thể không khả dụng trong Ionic framework. Trong trường hợp như vậy, bạn cần tự phát triển plugin để hoàn thành chức năng đó. Tuy nhiên,

có rất nhiều plugin có sẵn để bao gồm hầu hết các chức năng gốc. Ionic framework không hoàn hảo cho các ứng dụng hoặc trò chơi điện tử phụ thuộc vào đồ họa cao.

Công nghệ Ionic vẫn đang được phát triển. Nó thường xuyên thay đổi. Các thư viện của nó có thể được viết lại hoàn toàn bất cứ lúc nào. Do đó mà việc phát triển ứng dụng của bạn có thể bị gián đoạn.

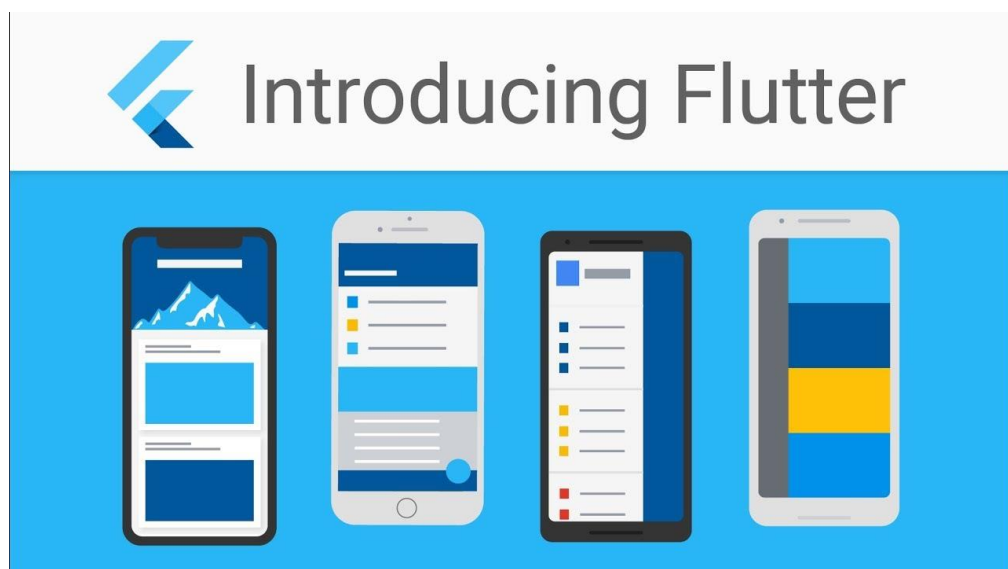
Rủi ro đối với doanh nghiệp

- Tính bảo mật của ứng dụng không tốt vì vậy dễ xảy ra nguy cơ bị đánh cắp dữ liệu
- Hiệu năng của ứng dụng thường sẽ không cao bằng các ứng dụng gốc dẫn đến trải nghiệm người dùng sẽ không tốt.

2.2 Giới thiệu về Flutter

2.2.1 Khái niệm

Flutter là UI Framework mã nguồn mở dành cho di động, miễn phí được tạo bởi Google và phát hành vào tháng 5 năm 2017. Flutter nó cho phép bạn tạo một ứng dụng di động gốc (native app) với một codebase. Điều này có nghĩa là bạn có thể sử dụng một ngôn ngữ lập trình và một codebase để tạo hai ứng dụng khác nhau (cho iOS và Android). [3]



Hình 2.2 Giới thiệu về Flutter².

² Nguồn ảnh - <https://flutter.dev/>

Flutter gồm 2 thành phần quan trọng:

- Một SDK (Software Development Kit): Một bộ sưu tập các công cụ sẽ giúp bạn phát triển các ứng dụng của mình. Điều này bao gồm các công cụ để biên dịch mã của bạn thành mã máy gốc (mã cho iOS và Android).
- Một Framework (UI Library based on widgets): Một tập hợp các thành phần UI có thể tái sử dụng (button, text inputs, slider, v.v.) giúp bạn có thể cá nhân hóa tùy theo nhu cầu của riêng mình.

2.2.2 Ngôn lập trình – Dart

Để lập trình với Flutter, bạn sẽ sử dụng một ngôn ngữ lập trình gọi là Dart. Dart là ngôn ngữ lập trình đa mục đích ban đầu được phát triển bởi Google và sau đó được Ecma (ECMA-408) phê chuẩn làm tiêu chuẩn. Nó được sử dụng để xây dựng các ứng dụng web, server, máy tính để bàn và thiết bị di động. [4]

Dart là một ngôn ngữ hướng đối tượng, được xác định theo lớp, với cơ chế garbage-collected, sử dụng cú pháp kiểu C để dịch mã tùy ý sang JavaScript. Nó hỗ trợ interface, mixin, abstract, generic, static typing và sound type (2 cái cuối có thể hiểu là type-safe).

Ưu điểm của ngôn ngữ Dart:

- Cú pháp Dart rõ ràng và súc tích, công cụ của nó đơn giản nhưng mạnh mẽ. Type-safe giúp bạn xác định sớm các lỗi tinh tế. Dart có các thư viện cốt lõi và một hệ sinh thái gồm hàng ngàn package.
- Dart cung cấp tối ưu hóa việc biên dịch trước thời hạn để có được dự đoán hiệu suất cao và khởi động nhanh trên các thiết bị di động và web.
- Dart biên dịch thành mã ARM và x86, để các ứng dụng di động của Dart có thể chạy tự nhiên trên iOS, Android và hơn thế nữa. Đối với các ứng dụng web, chuyển mã từ Dart sang JavaScript.
- Dart quen thuộc với nhiều nhà phát triển hiện có, nhờ vào cú pháp và cấu trúc hướng đối tượng
- Dart rất phù hợp với lập trình Reactive, với sự hỗ trợ để quản lý các đối tượng tồn tại trong thời gian ngắn, chẳng hạn như các widget UI, thông

qua phân bổ đối tượng nhanh và GC. Dart hỗ trợ lập trình không đồng bộ thông qua các tính năng ngôn ngữ và API sử dụng các đối tượng Future và Stream.

2.2.3 Ưu điểm

a) Thời gian phát triển nhanh.

Giống như nhiều cross-platform framework khác, Flutter cũng giúp phát triển ứng dụng nhanh hơn. Sử dụng single code-base của Flutter, bạn có thể khởi chạy ứng dụng của mình trên nhiều nền tảng. [5]

Điều này có nghĩa là bạn chỉ cần viết code một lần thay vì viết code riêng cho từng hệ điều hành. Do đó, phát triển Flutter tiết kiệm nỗ lực phát triển và cực kỳ hiệu quả về chi phí. Việc sửa và theo dõi các lỗi trong mã nguồn cũng đỡ vất vả hơn, vì các thay đổi được phản ánh trên mọi nền tảng cùng một lúc.

b) Tự do tùy chỉnh và kết xuất nhanh hơn.

Kiến trúc phân lớp của Flutter có thể giúp bạn tự do tùy chỉnh các thành phần giao diện người dùng đến mức bạn cảm thấy như mình đang kiểm soát mọi pixel trên màn hình. Flutter là một công cụ hỗ trợ tuyệt vời cho thư viện hình ảnh và đó cũng là lý do vì sao mình chọn Flutter cho đồ án của mình.

c) Tính năng hot reload.

Sử dụng hot reload, các nhà phát triển hiện có thể thay đổi mã và ngay lập tức nhìn thấy chúng trong thời gian chạy trong bản xem trước ứng dụng. Không cần phải biên dịch lại code mỗi lần. Điều này cho phép các nhà phát triển tự do thử nghiệm với các giao diện người dùng khác nhau trong khi xây dựng ứng dụng.

d) Tương đương với Hiệu suất ứng dụng gốc.

Các ứng dụng di động được phát triển bằng Flutter không thể phân biệt được với các ứng dụng được tạo bằng mã gốc. Trên thực tế, các ứng dụng Flutter có thể hoạt động tốt hơn trong các tình huống giao diện người dùng phức tạp.

Lý do chính đằng sau điều này là Flutter không dựa trên bất kỳ code trung gian nào hoặc cầu nối JavaScript. Thay vào đó, nó được tích hợp trực tiếp vào mã máy. Điều này giúp loại bỏ bất kỳ lỗi hiệu suất có thể xảy ra thường gặp trong quá trình giải thích.

2.2.4 Nhược điểm

a) Chưa ổn định

Flutter là một trong những framework khá non trẻ và vẫn đang tiếp tục phát triển. Nhiều thư viện của nó vẫn đang trong giai đoạn tiền alpha và cần kiểm tra thêm trước khi trở nên ổn định. Do đó, Flutter cần thêm thời gian để thực sự khai thác tiềm năng của một hệ điều hành.

b) Kích thước tệp lớn

Một nhược điểm khác của việc sử dụng Flutter là kích thước tệp ứng dụng. Các ứng dụng được tạo bằng Flutter rất nặng và mất nhiều thời gian để khởi động hoặc tải hơn. Điều này có thể làm hỏng trải nghiệm của người dùng và là dấu hiệu về hiệu suất kém.

c) Thường xuyên thay đổi và cập nhật

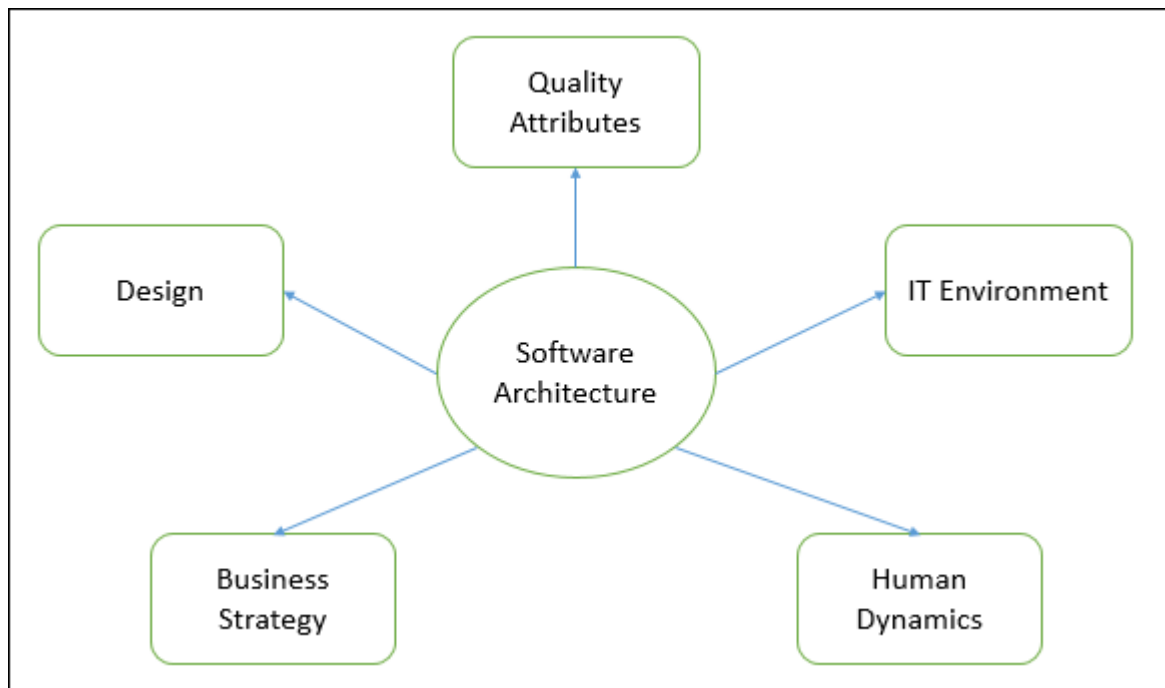
Vì Flutter là một framework mới nên nó thường xuyên trải qua các thay đổi và cập nhật có thể ảnh hưởng đến việc phát triển. Duy trì code có thể là một thách thức trong một môi trường thay đổi nhanh chóng như vậy.

2.3 Kiến trúc phần mềm

2.3.1 Kiến trúc phần mềm là gì?

Kiến trúc phần mềm (*Software architecture*) của hệ thống mô tả các thành phần (component) chính, mối quan hệ của chúng, và cách chúng tương tác với nhau. [6]

Nó được xem như một bản thiết kế chi tiết, cung cấp mô hình để quản lý hệ thống và thiết lập cũng như điều phối giao tiếp giữa các component. Kiến trúc xác định một giải pháp để đáp ứng các yêu cầu kỹ thuật và vận hành. với mục tiêu chung là hiệu năng và bảo mật.



Hình 2.3 Software architecture³.

Software architecture được dùng để xác định phần khung và các component cấp cao của hệ thống, và cách chúng làm việc với nhau. Chẳng hạn, bạn cần một kiến trúc serverless chia nhỏ ứng dụng thành 2 component: BaaS (backend-as-a-service) và FaaS (functions-as-a-service), hoặc bạn cần một kiến trúc microservice mà các chức năng được chia thành các modules riêng biệt.

2.3.2 Mẫu kiến trúc

Mẫu kiến trúc (*Architecture patterns*) là một giải pháp tổng thể cho các vấn đề chung trong thiết kế kiến trúc phần mềm.

Architecture patterns giúp bạn đơn giản hóa kiến trúc phức tạp đằng sau các ứng dụng. Các tầng này được tách riêng theo debugging, testing vì vậy việc maintenance ứng dụng trở nên dễ dàng hơn. Nó hỗ trợ giải quyết các vấn đề khác nhau trong kỹ thuật phần mềm, chẳng hạn như tăng hiệu năng ứng dụng và giảm thiểu rủi ro trong kinh doanh.

³ Software architecture: Kiến trúc phần mềm - <https://viblo.asia/p/thiet-ke-ung-dung-web-software-architecture-101-4dbZNLaq5YM>

Một vài ví dụ về Architectural Patterns: [7]

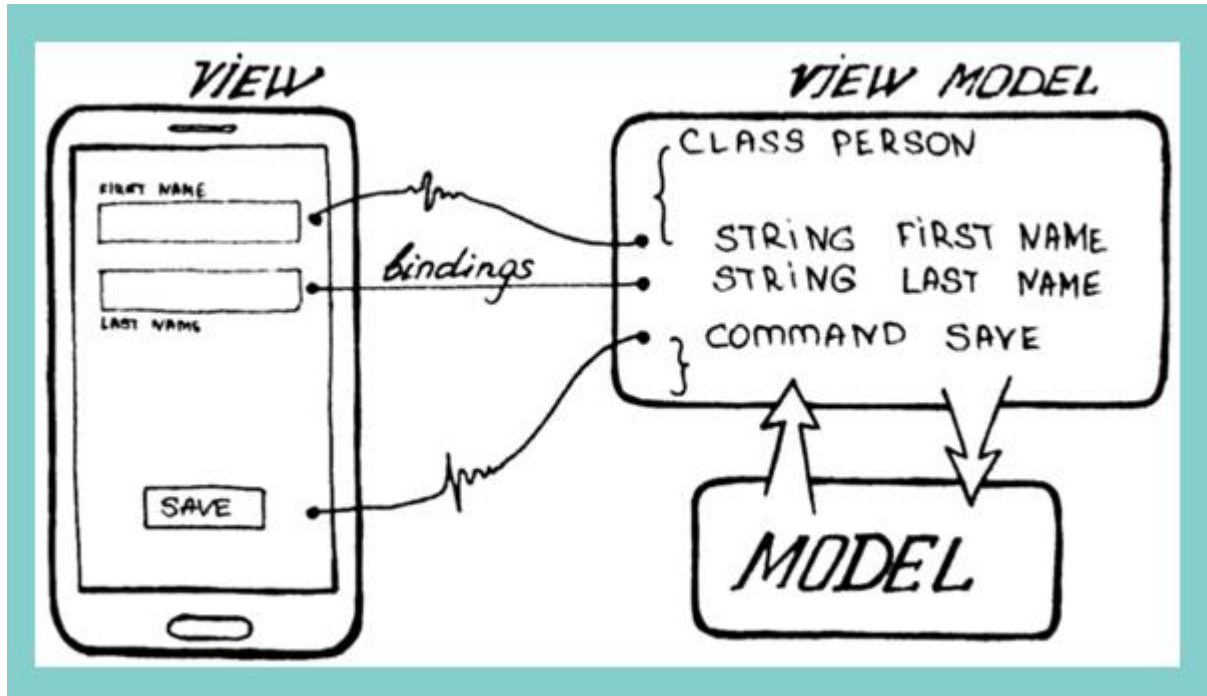
- Three – tier
- Microkernel
- Model – View – Controller (MVC)
- Model – View – ViewModel (MVVM)

2.3.3 Mô hình kiến trúc Model – View – ViewModel (MVVM)

Kể từ khi Microsoft cho ra mắt hai nền tảng phát triển ứng dụng mới đó là WPF (Windows Presentation Foundation) và Silverlight. Trên nền tảng đó đã có nhiều thay đổi trong việc xử lý sự kiện và binding dữ liệu, giữa các tầng của ứng dụng với nhau. Vì vậy, nảy sinh ra nhu cầu phải có một mô hình phát triển ứng dụng mới phù hợp hơn. Do vậy Model – View – ViewModel (MVVM) pattern ra đời và ngày càng trở nên phổ biến và mình đã lựa chọn mô hình này áp dụng vào đồ án.

a) Khái niệm

MVVM (Model – View – ViewModel) là mô hình hỗ trợ two-way data binding (ràng buộc dữ liệu) giữa View và ViewModel.



Hình 2.4 Mô hình MVVM⁴

⁴ Nguồn ảnh - <https://wiki.matbao.net/mvc-mvp-mvvm-la-gi-thong-tin-can-biet-ve-cac-mo-hinh-lap-trinh/>

Mô hình bao gồm các thành phần chính sau:

- **Model:** là bộ phận đại diện cho các tầng business logic, định nghĩa business rules cho dữ liệu.
- **View:** là nơi cho phép người dùng tương tác, đưa ra yêu cầu cho ứng dụng.
- **View Model:** là lớp trung gian giữa View và Model. ViewModel là bộ phận có chức năng ghi nhận và điều hướng các yêu cầu (request) mà người dùng đưa ra tại View. Sau khi tiếp nhận, Controller sẽ xử lý các dữ liệu từ Model và cuối cùng trả kết quả, phản hồi tại View và ngược lại.

b) Ưu điểm

MVVM sẽ tạo sự tương tác hiệu quả giữa designer và developer.

Tăng khả năng sử dụng lại các thành phần hay việc thay đổi giao diện chương trình mà không cần thiết phải viết lại code quá nhiều.

Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì,...

c) Nhược điểm

Đối với dự án nhỏ việc áp dụng mô hình MVVM gây cồng kềnh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các thành phần.

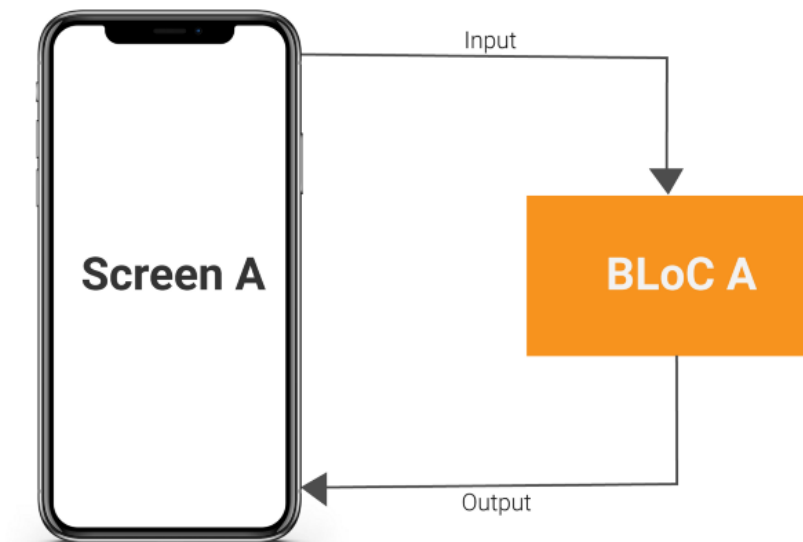
Đối với dự án lớn hơn, nó gây khó khăn và mất thời gian để thiết kế các ViewModel.

Việc liên kết dữ liệu cho tất cả các thành phần gây khó khăn trong việc debug khi cơ sở dữ liệu phức tạp.

1.3.4 Bloc Pattern trong Flutter

a) Khái niệm

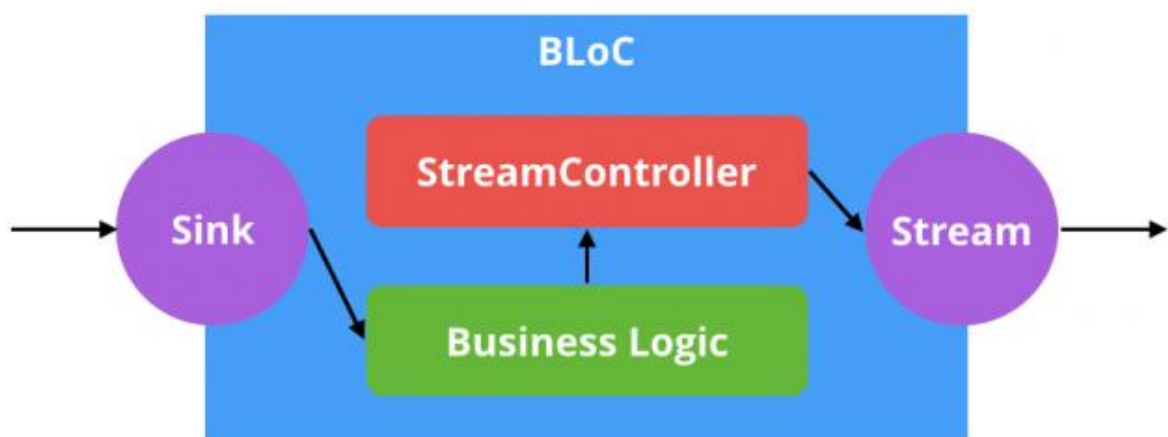
BLoC Parttern gần như hoàn toàn tương đồng với kiến trúc MVVM. Bloc là 1 hệ thống quản lý state cho Fultter được phát triển bởi Google developers. Nó giúp lập trình viên quản lý state và luồng dữ liệu trong ứng dụng. [8]



Hình 2.5 Bloc Parttern⁵.

b) Áp dụng

Để triển khai Bloc Parttern, người ta sử dụng StreamController.



Hình 2.6 Bloc Parttern for Flutter⁶.

Dữ liệu sẽ được thêm vào sink (tạm gọi là thùng chứa dữ liệu) của StreamController và phía UI sẽ sử dụng Stream để lắng nghe nhận state mỗi khi có event được add vào sink. Tức là, Stream nhận nhiệm vụ lắng nghe sự kiện từ sink, mỗi khi dữ liệu trong trong sink thay đổi, Stream sẽ thông báo đến UI và UI lập tức cập nhật lại giao diện với dữ liệu mới.

⁵ Bloc Pattern: Mẫu thiết kế Bloc - <https://viblo.asia/p/hoc-bloc-pattern-theo-cach-de-hieu-nhat-maGK7JYO5j2>

⁶ Nguồn ảnh - <https://viblo.asia/p/hoc-bloc-pattern-theo-cach-de-hieu-nhat-maGK7JYO5j2>

1.4 Giới thiệu về Bullet Journal

1.4.1 Bullet Journal là gì?



Hình 2.7 Ví dụ về Bullet Journal⁷.

Bullet Journal là một cách ghi chép công việc và kế hoạch rất hiệu quả do Ryder Carroll, một nhà thiết kế tại Brooklyn sáng tạo ra, nó sử dụng phương pháp rapid logging để quản lý và theo dõi công việc hằng ngày của cá nhân bằng một quyển sổ và một cây bút. [9]

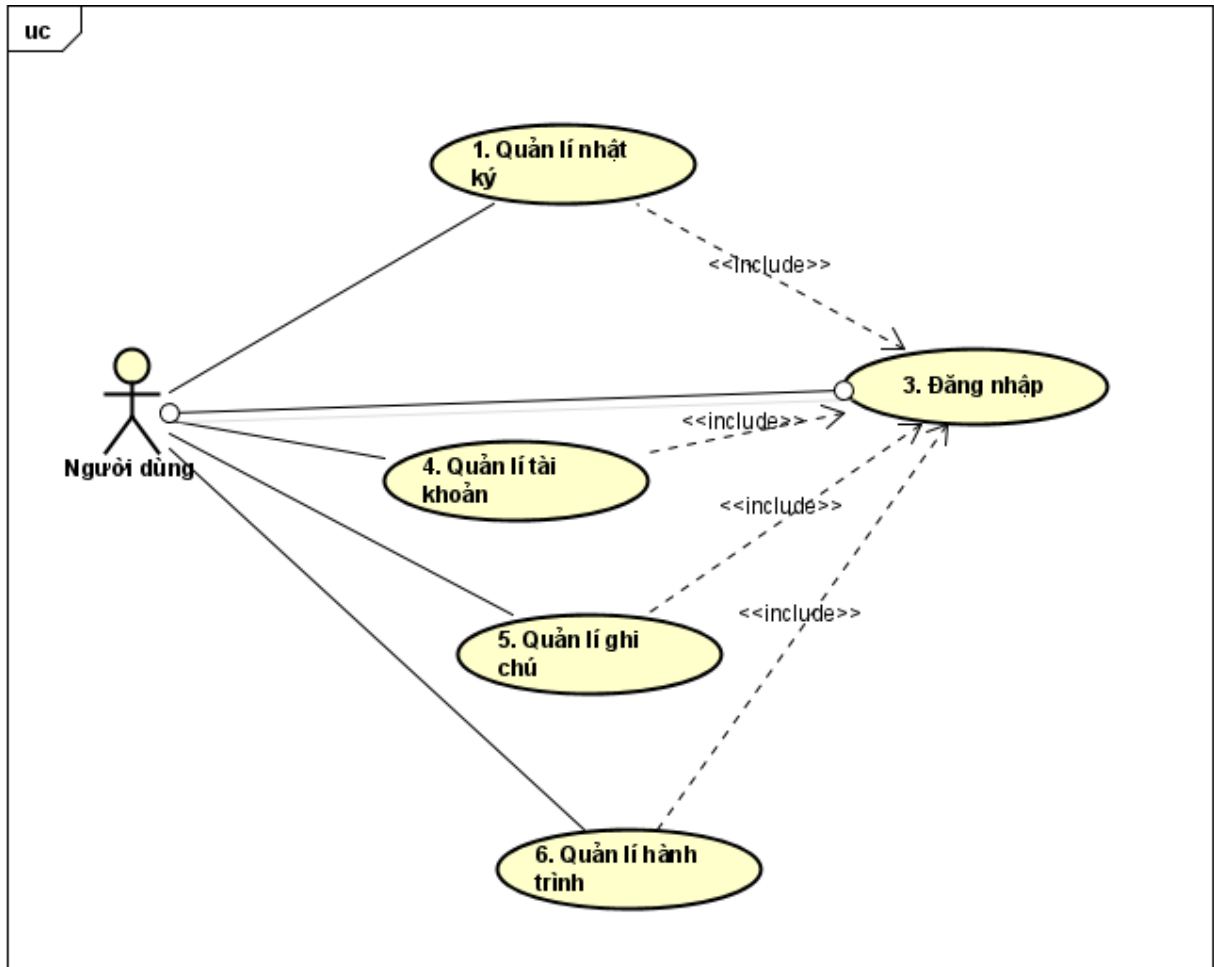
Nếu bạn là một người thích viết lách và ghi chú, chắc chắn bạn sẽ bị thu hút bởi loại nhật ký mới mẻ mang tên Bullet Journal. Bullet Journal được định nghĩa là ⁷một hình thức ghi chép sổ tay mà trong đó bạn sẽ ghi lại các mục tiêu ngắn và dài hạn của bản thân, danh sách các việc cần làm trong ngày theo một cách khoa học và bố cục được trình bày theo sở thích. Bạn có thể thỏa sức sáng tạo với những hình ảnh, nhãn dán, kí hiệu để tạo ra một quyển Bullet Journal độc đáo cho riêng bạn.

⁷ Nguồn ảnh: Fanpage Cộng đồng Bullet Journal Việt Nam - <https://www.facebook.com/groups/290658024746040>

Tùy vào sự thay đổi của bạn mà cách sử dụng Bullet Journal cũng sẽ thay đổi theo. Nếu tháng tới bạn du lịch, Bullet Journal sẽ hỗ trợ bạn ghi chép lại những phụ kiện, vật dụng du lịch bạn cần đem theo. Hoặc, nếu bạn muốn lên kế hoạch cho các bữa tiệc chính chu hơn, bạn cũng có thể ghi chúng vào Bullet Journal.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

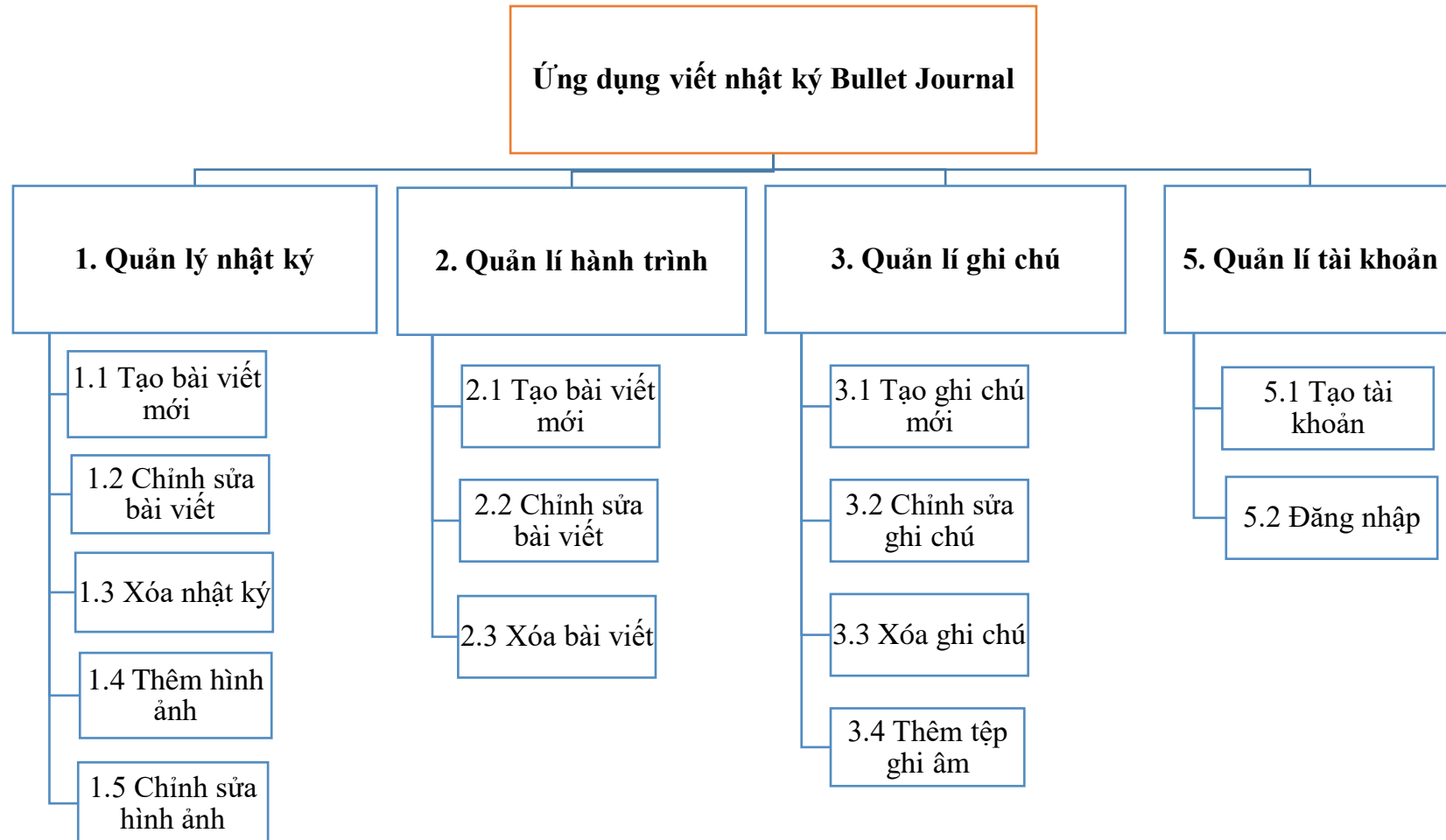
3.1 Sơ đồ usecase của hệ thống



Hình 3.1 Sơ đồ usecase tổng quát

3.2 Sơ đồ phân rã chức năng.

3.2.1 Sơ đồ phân rã chức năng tổng quát.



Hình 3.2 Sơ đồ phân rã chức năng tổng quát.

3.2.2 Mô tả chức năng.

a) Chức năng quản lý nhật ký.

Chức năng quản lý nhật ký cho phép người sử dụng quản lý nhật ký với các chức năng chính:

- Xem thông tin bài viết.
- Tìm kiếm bài viết.
- Tạo mới bài viết.
- Chỉnh sửa bài viết.
- Xóa bài viết.

b) Chức năng quản lý ghi chú.

Chức năng quản lý cho phép người dùng quản lý ghi chú với các chức năng:

- Xem thông tin ghi chú.
- Tìm kiếm ghi chú.
- Tạo mới ghi chú.
- Chỉnh sửa ghi chú.
- Xóa ghi chú.

c) Chức năng quản lý chuyến đi.

Chức năng cho phép người dùng quản lý chuyến đi với các chức năng:

- Xem thông tin chuyến đi.
- Tìm kiếm chuyến đi.
- Tạo mới chuyến đi.
- Chỉnh sửa chuyến đi.
- Xóa chuyến đi.

d) Chức năng quản lý công việc.

Chức năng cho phép người dùng quản lý công việc với các chức năng:

- Hiển thị lịch và đánh dấu ngày có công việc.
- Hiển thị danh sách công việc theo ngày.
- Tìm kiếm.

e) Chức năng quản lý tài khoản.

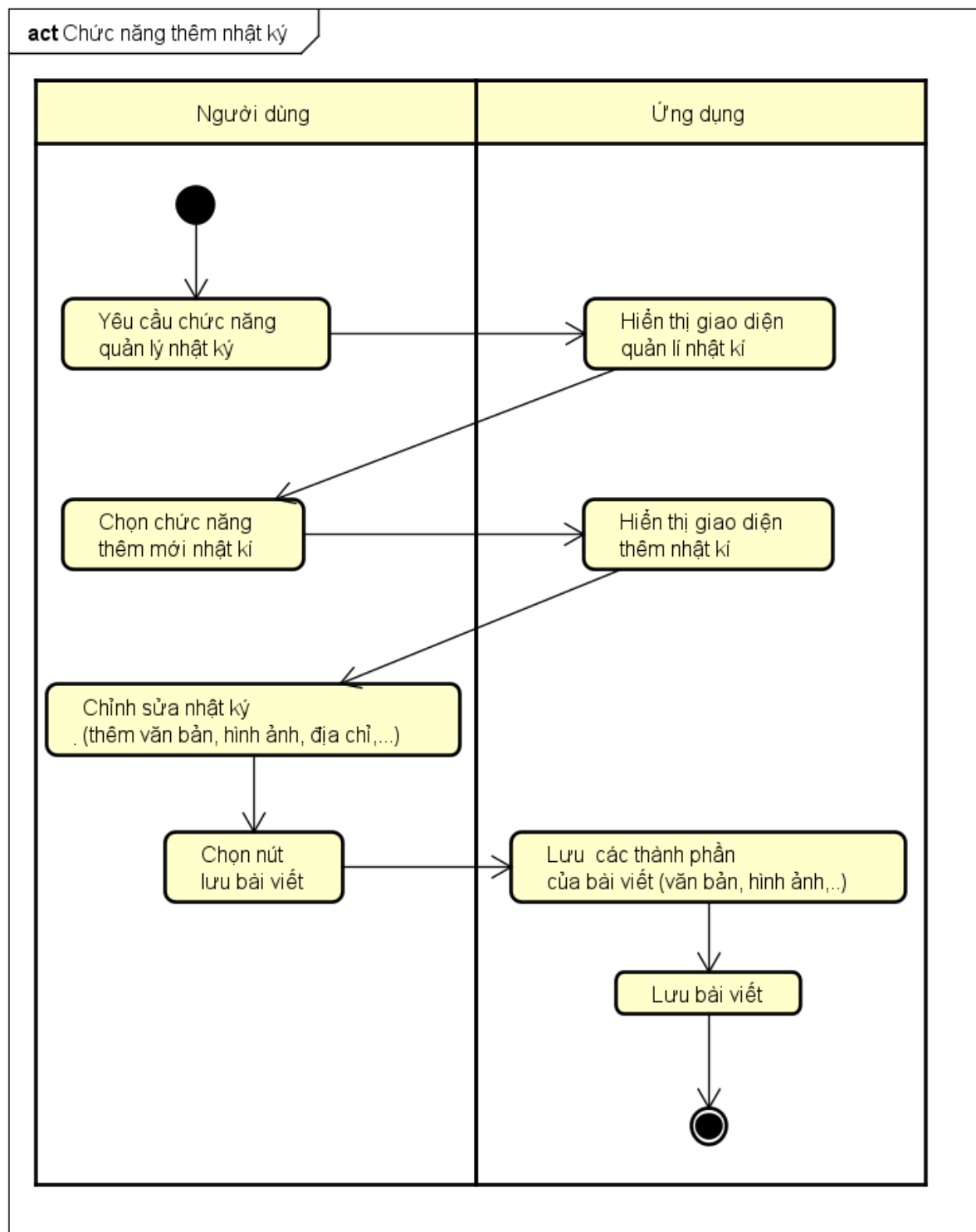
Chức năng cho phép người dùng:

- Tạo tài khoản.
- Đăng nhập.

3.3 Sơ đồ hoạt động

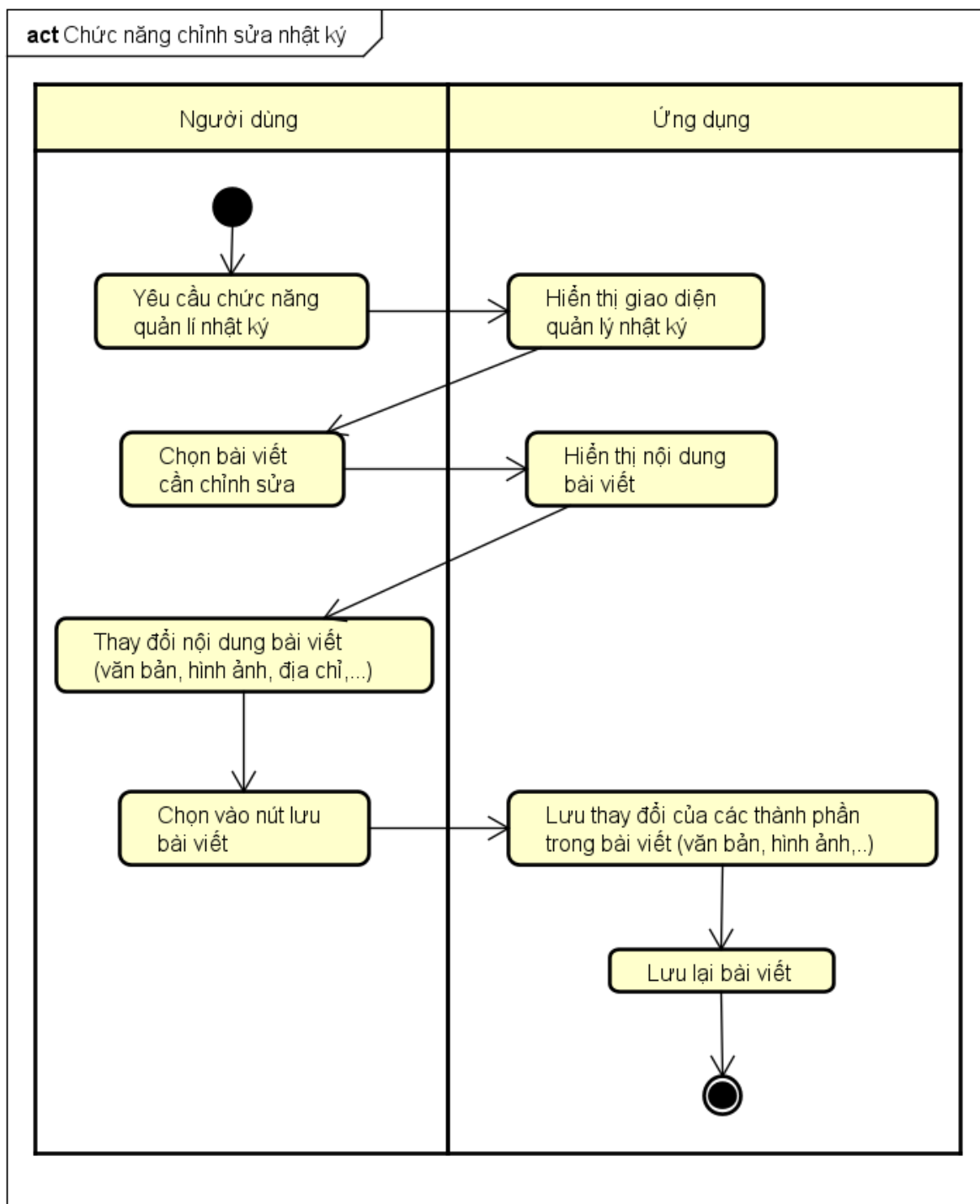
3.3.1 Chức năng quản lý nhật ký.

a) Chức năng thêm nhật ký



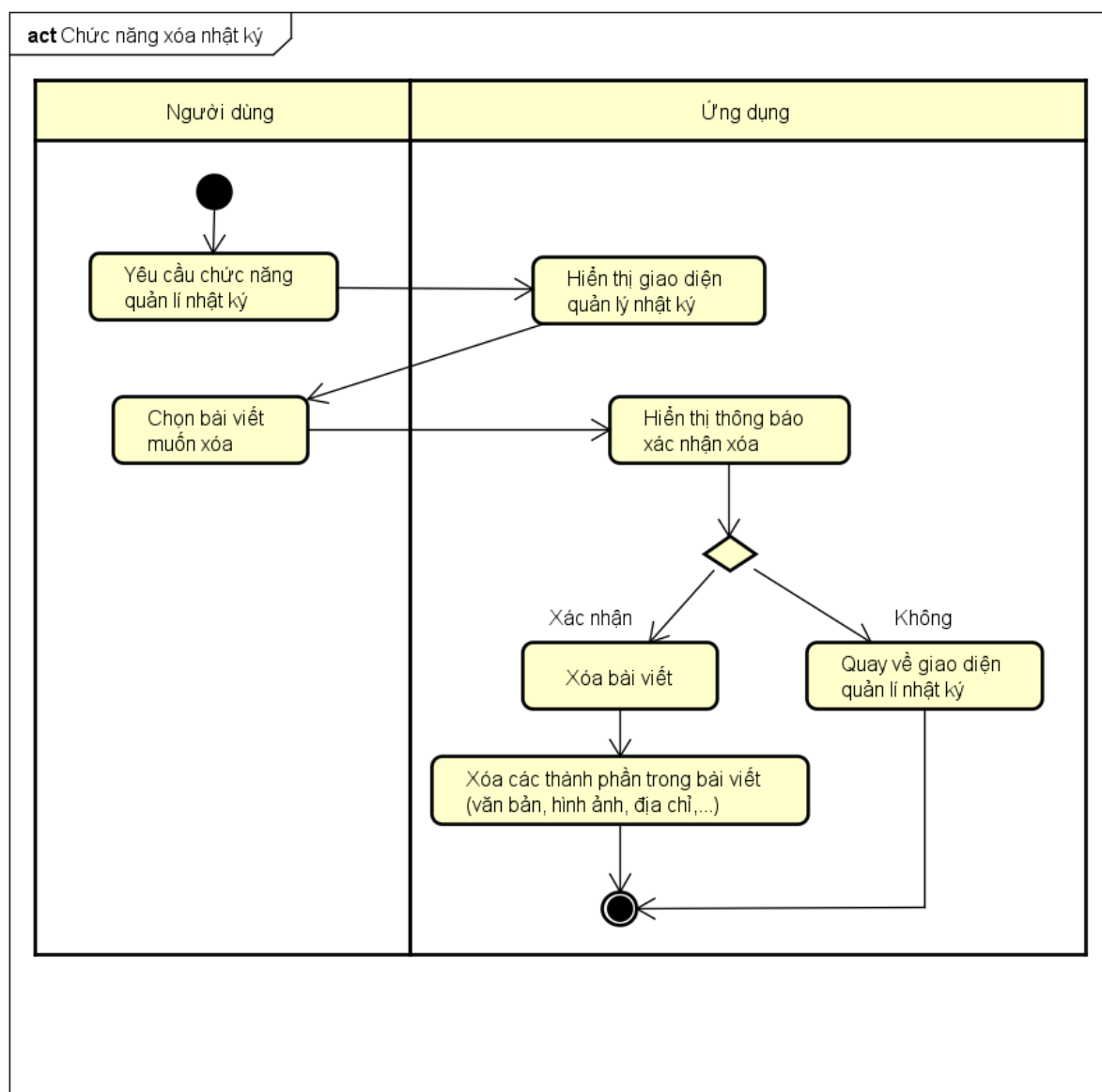
Hình 3.3 Chức năng thêm nhật ký.

b) Chức năng chỉnh sửa nhật ký



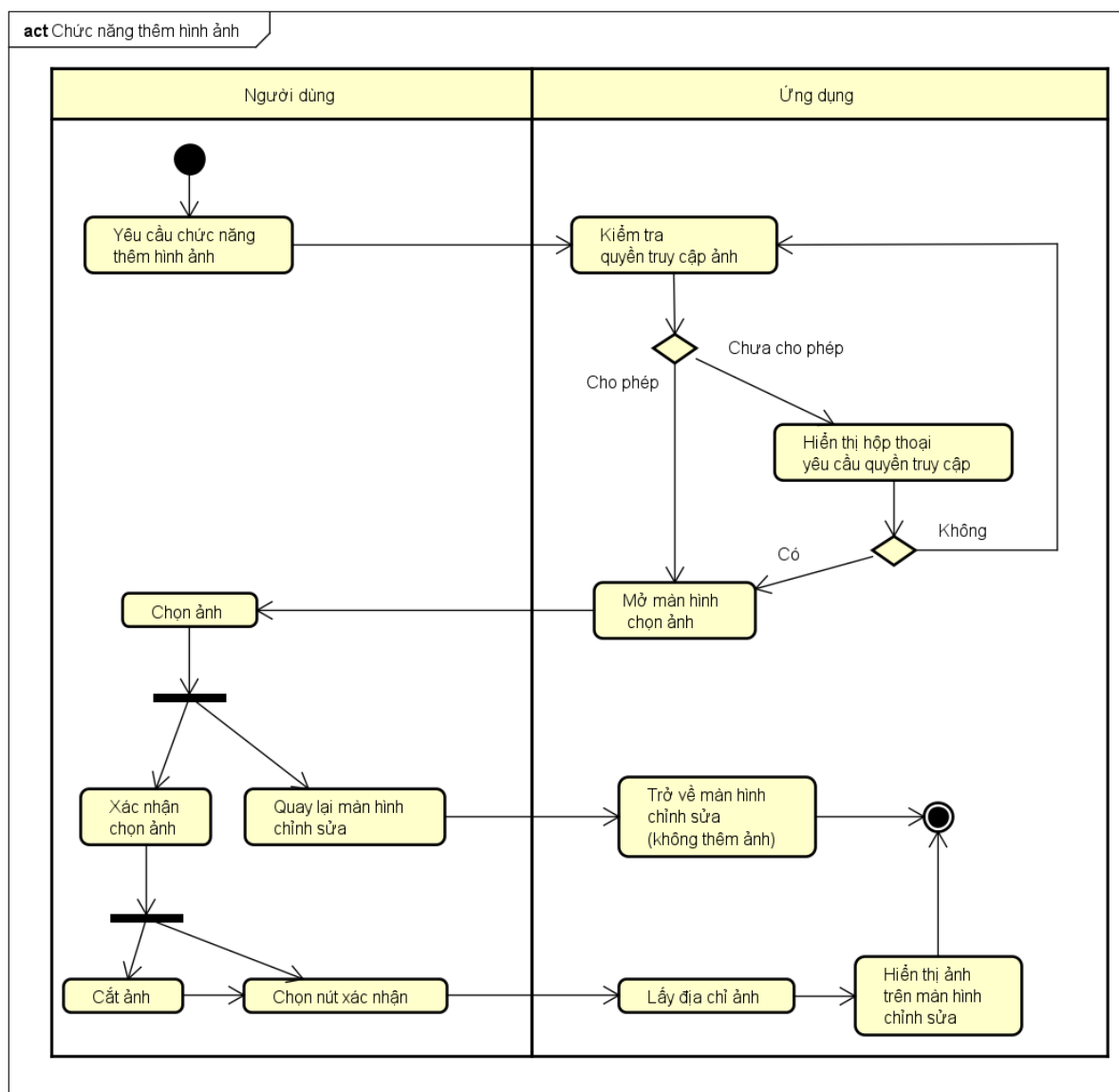
Hình 3.4 Chức năng chỉnh sửa nhật ký.

c) Chức năng xóa nhật kí



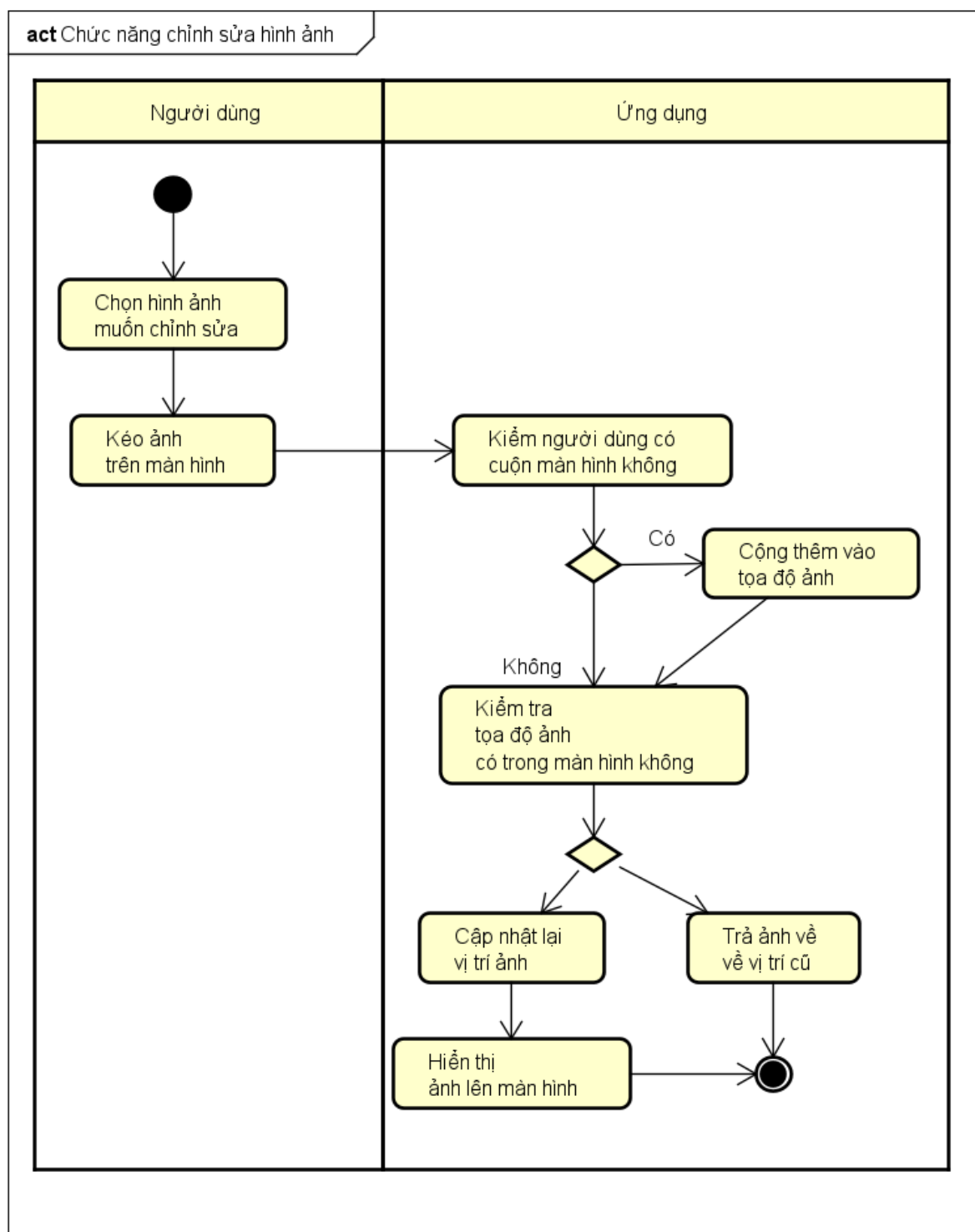
Hình 3.5 Chức năng xóa nhật ký.

d) Chức năng thêm hình ảnh



Hình 3.6 Chức năng thêm hình ảnh.

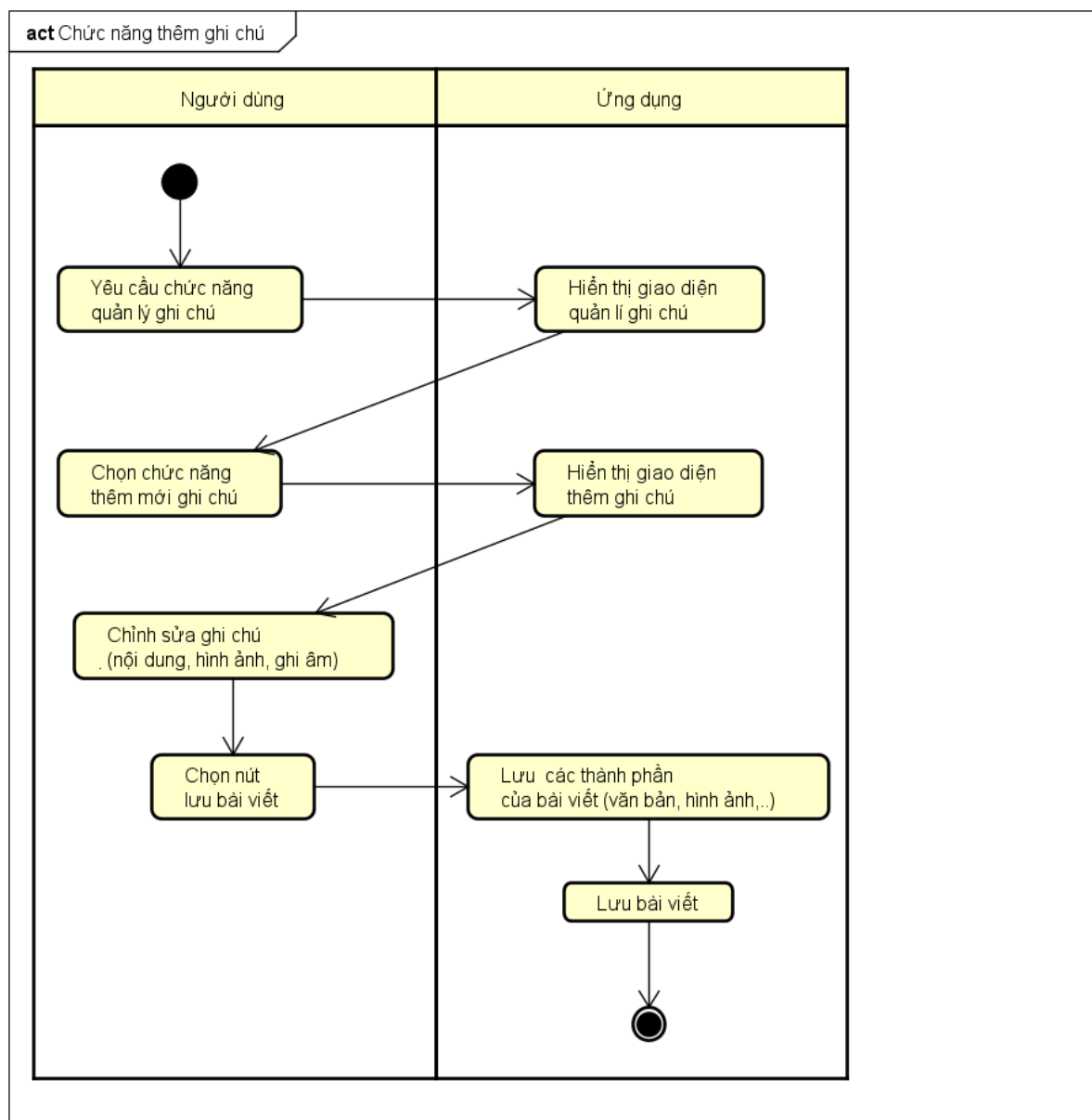
e) Chức năng chỉnh sửa hình ảnh



Hình 3.7 Chức năng chỉnh sửa hình ảnh.

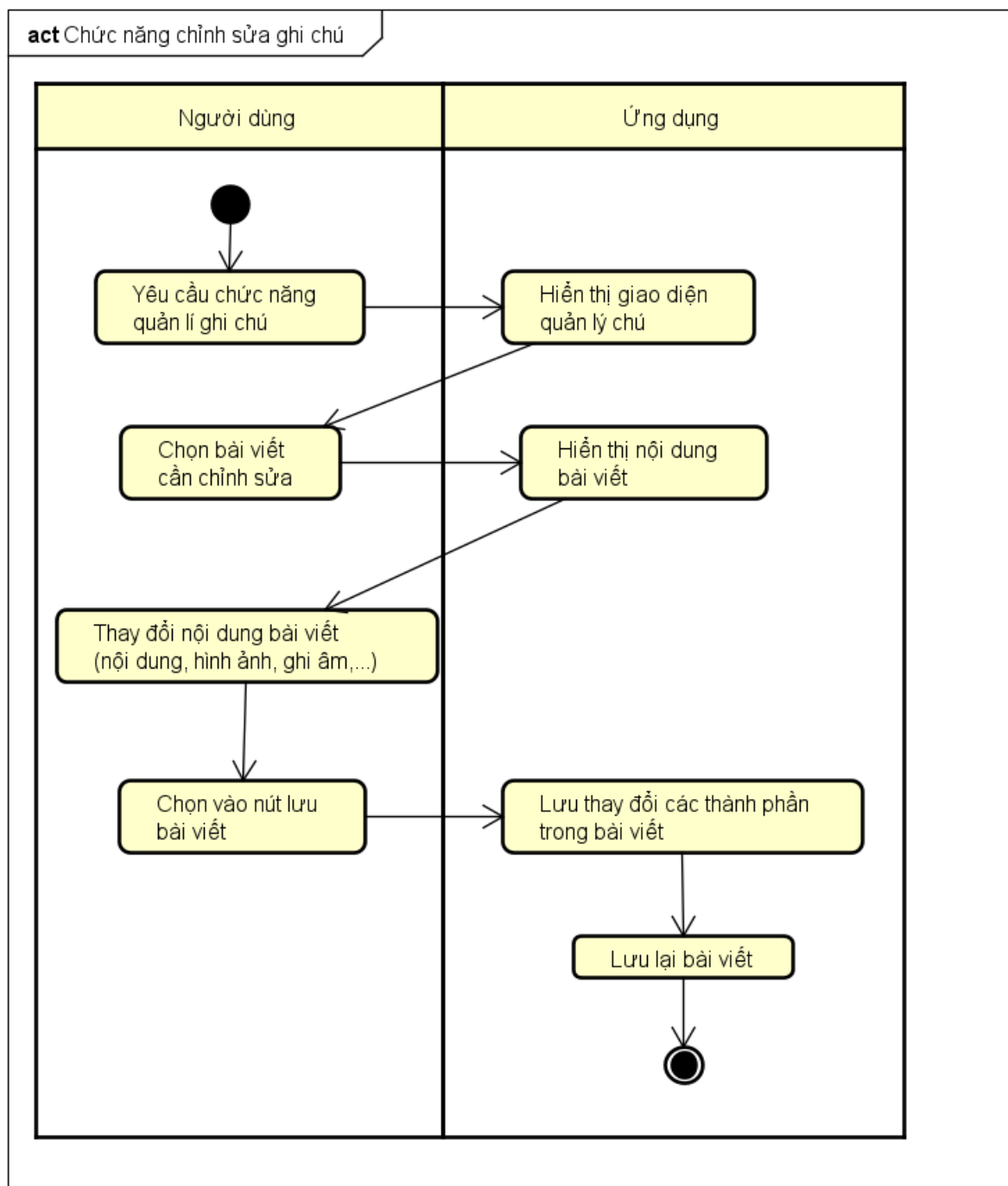
3.3.2 Chức năng quản lý ghi chú.

a) Chức năng thêm ghi chú



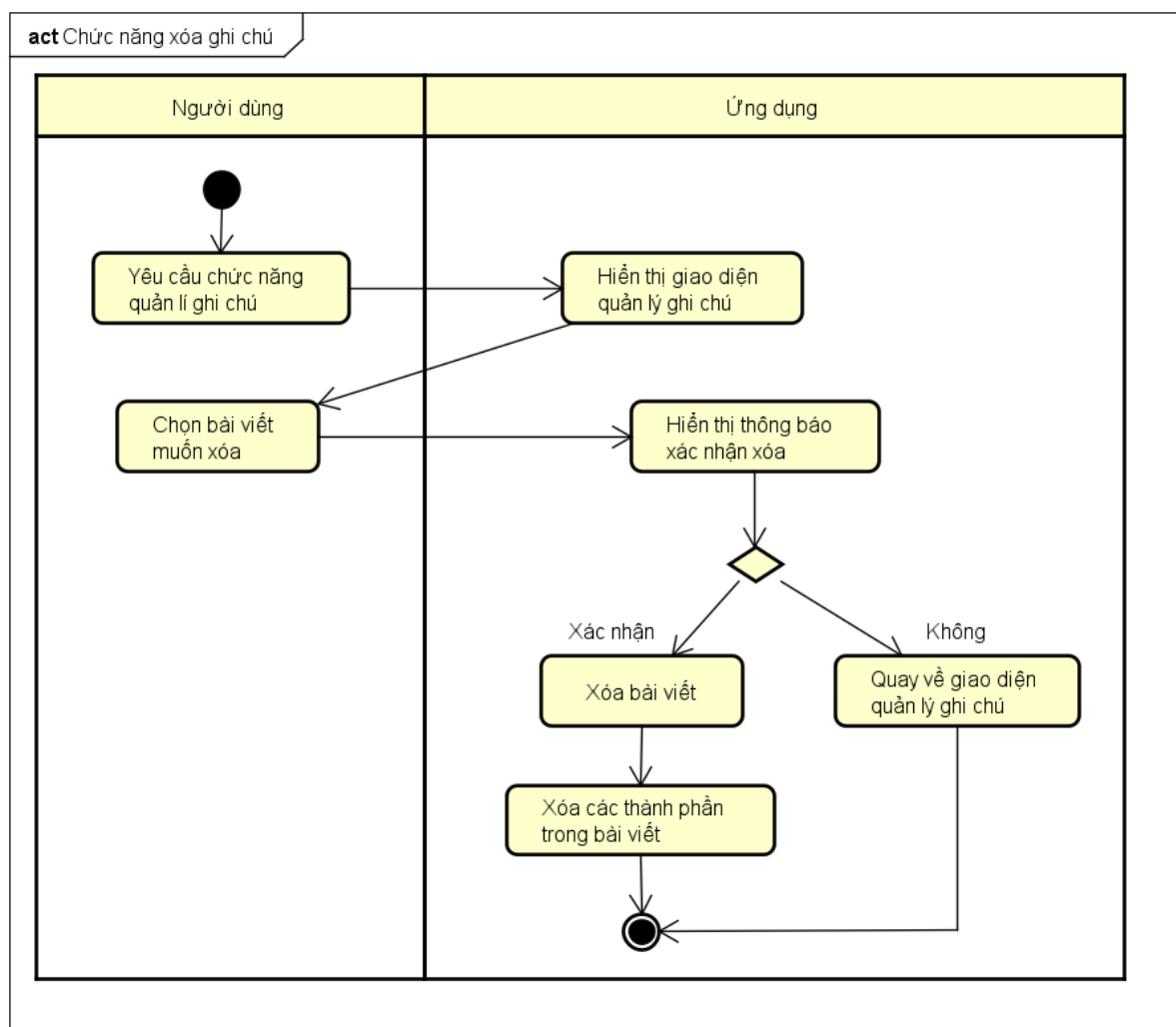
Hình 3.8 Chức năng thêm ghi chú.

b) Chức năng chỉnh sửa ghi chú



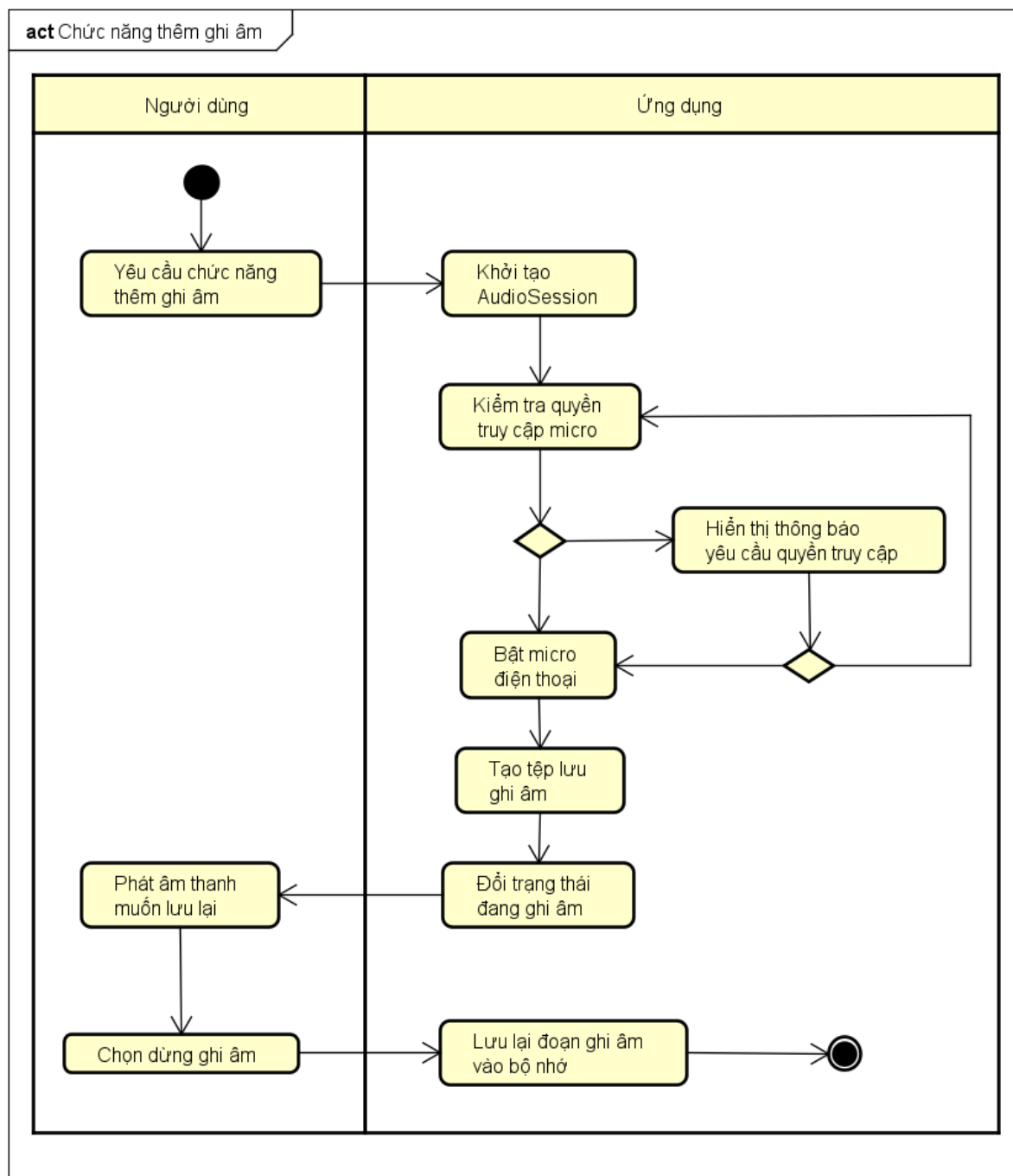
Hình 3.9 Chức năng chỉnh sửa ghi chú.

c) Chức năng xóa ghi chú



Hình 3.10 Chức năng xóa ghi chú.

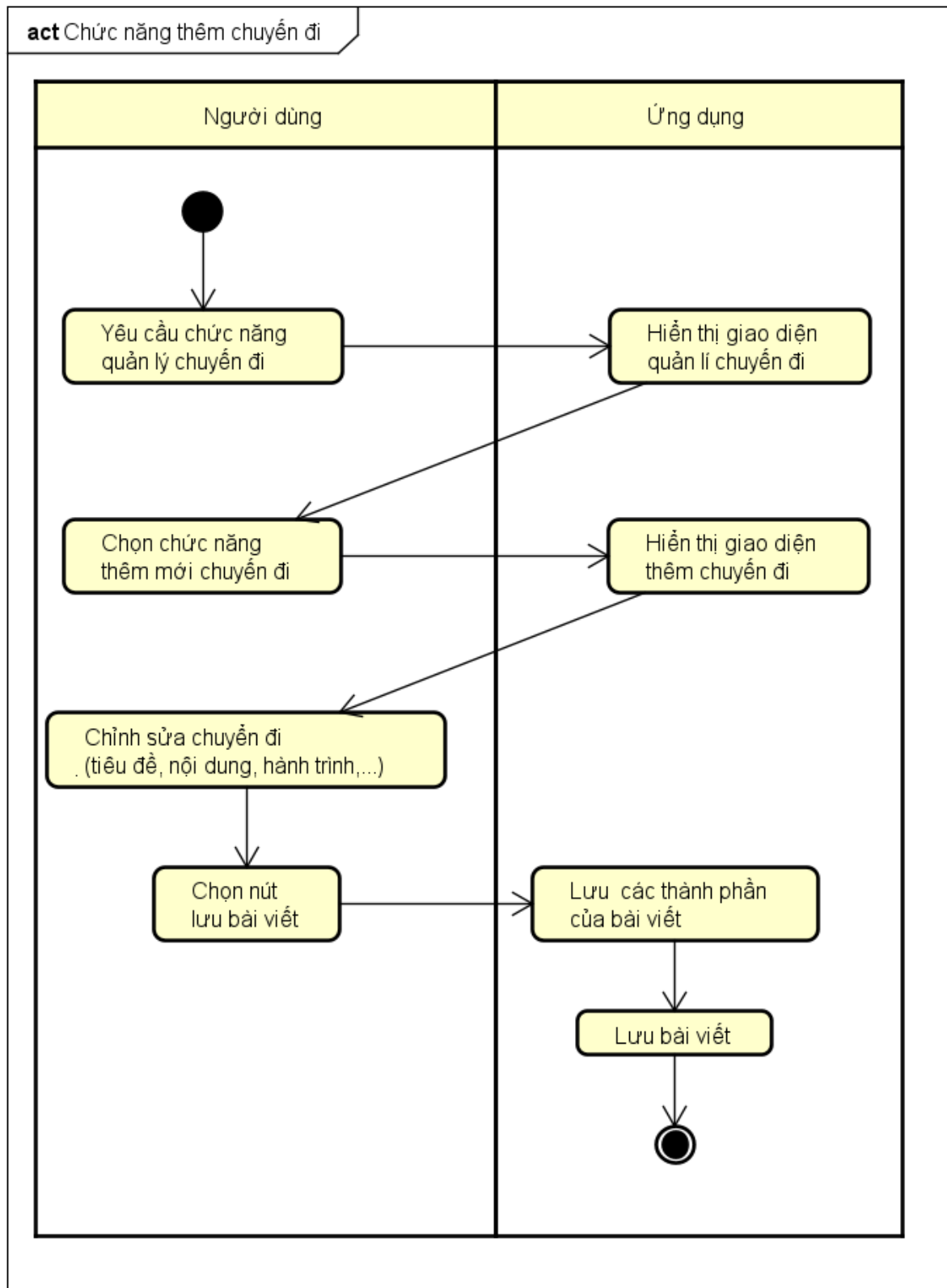
d) Chức năng thêm tệp ghi âm



Hình 3.11 Chức năng thêm tệp ghi âm.

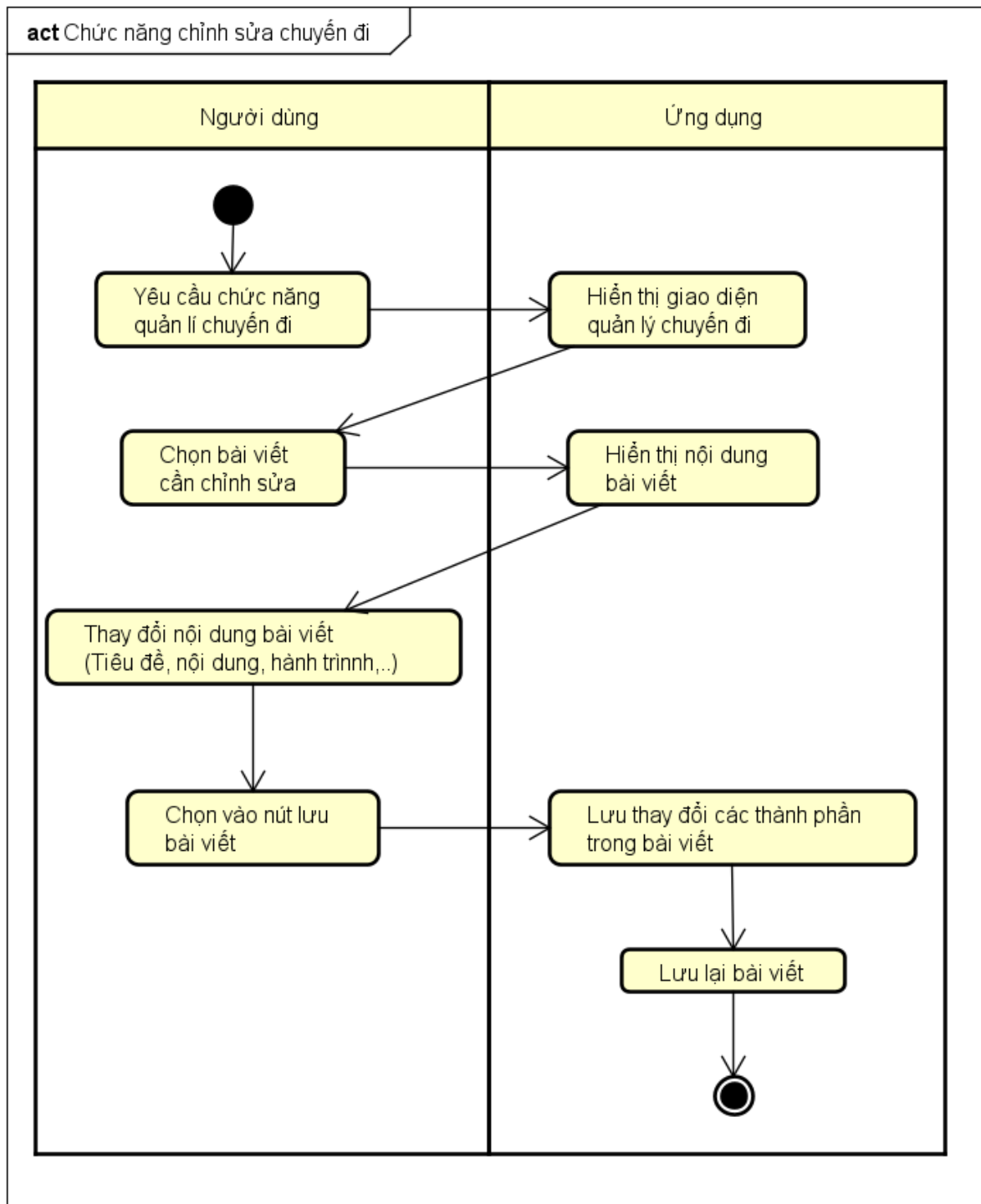
2.3.3 Chức năng quản lý chuyến đi.

a) Chức năng thêm chuyến đi



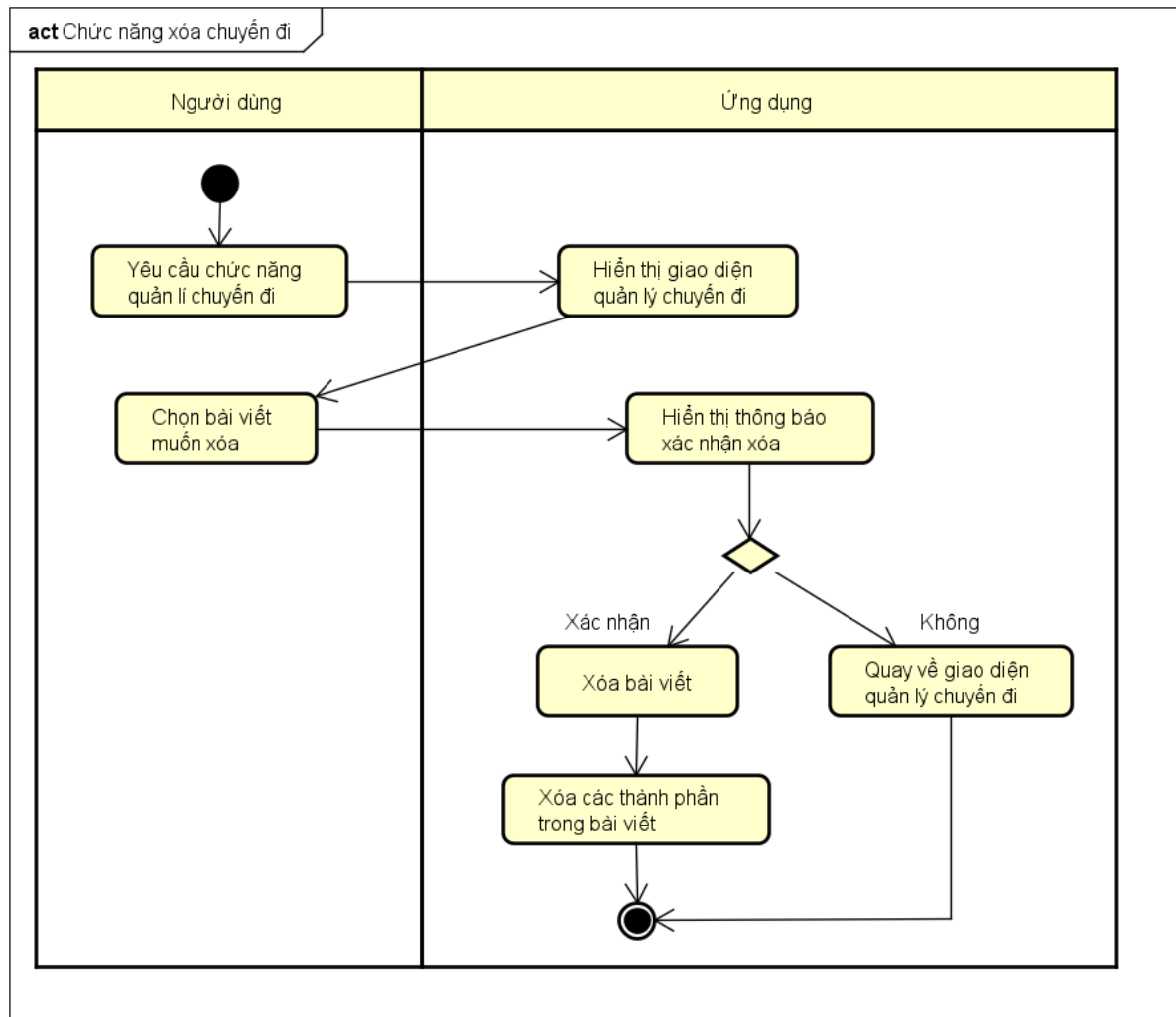
Hình 3.12 Chức năng quản lý chuyến đi.

b) Chức năng chỉnh sửa chuyển đi



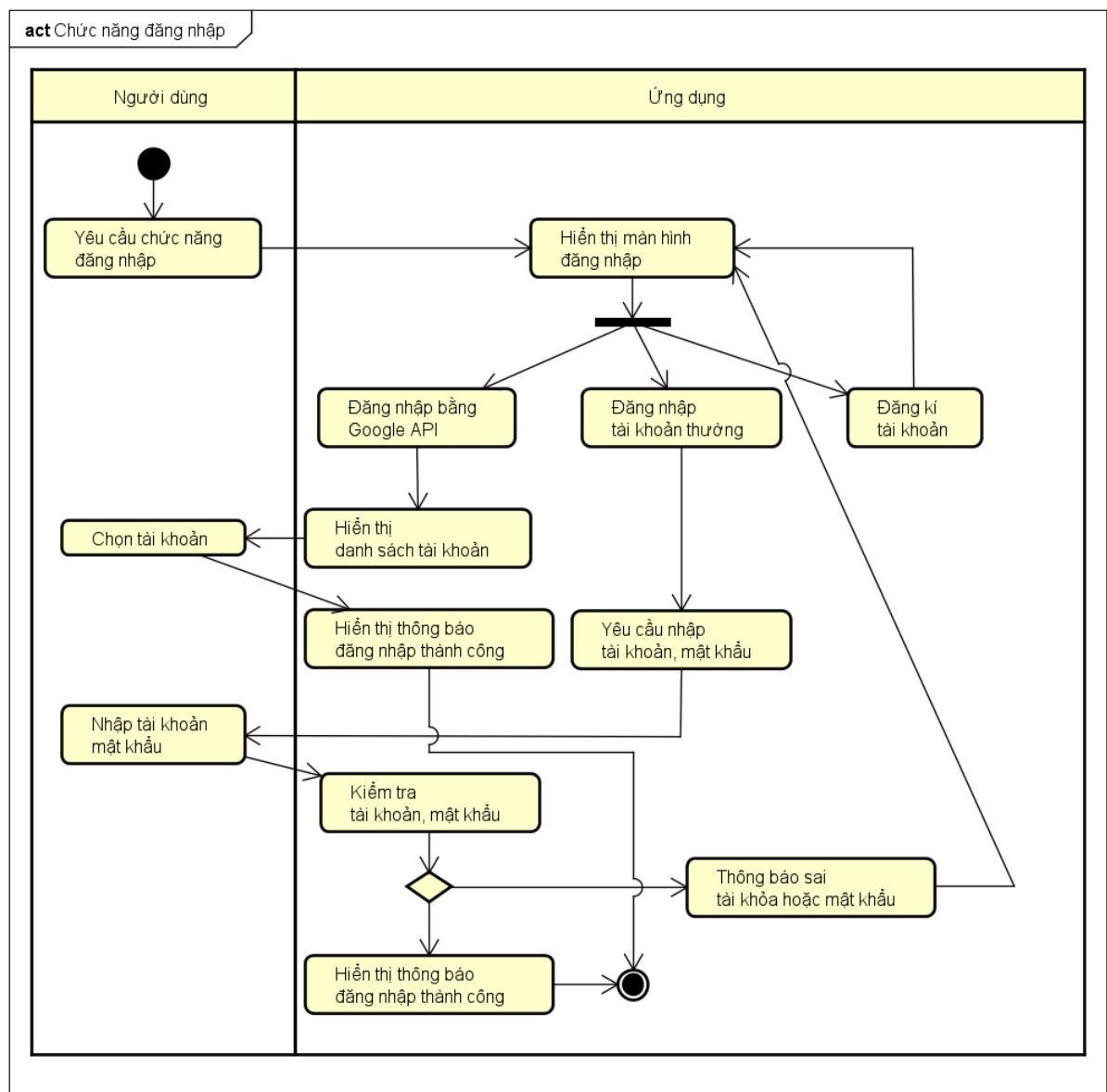
Hình 3.13 Chức năng chỉnh sửa chuyển đi.

c) Chức năng xóa chuyển đi



Hình 3.14 Chức năng xóa chuyển đi.

2.2.4 Chức năng đăng nhập.



Hình 3.15 Chức năng đăng nhập.

CHƯƠNG 4. THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

4.1 Hiện trạng

Hiện tại có rất nhiều ứng dụng viết nhật ký tuy nhiên một ứng dụng viết nhật ký theo phong cách Bullet Journal là rất ít và những ứng dụng này thường ít tùy chỉnh cho bản miễn phí, đa phần đòi hỏi người dùng phải trả phí để sử dụng hết các tính năng ứng dụng.

Các ứng dụng thường là riêng lẻ, vì vậy người dùng muốn sử dụng chức năng nào thì phải tải ứng dụng đó. Điều này rất dễ gây ra sự nhầm lẫn về chức năng ứng, các ứng dụng cũng không đồng nhất với nhau nên sẽ tạo ra nhiều bất cập khi sử dụng nhiều ứng dụng cùng lúc.

4.2 Yêu cầu hệ thống

4.2.1 Yêu cầu chức năng

- Chức năng viết nhật ký: có thể tùy chỉnh thêm hình ảnh, văn bản, biểu tượng cảm xúc và địa chỉ. Người dùng có thể di chuyển các thành phần này theo bố cục mà mình mong muốn.
- Chức năng ghi chú: người sử dụng cũng có thể tùy chọn thêm hình ảnh và ghi âm.
- Chức năng ghi lại hành trình chuyến đi: có thể dùng chức năng này để lập kế hoạch du lịch hoặc lưu lại các chuyến đi đã qua.
- Chức năng ghi lại các hoạt động hàng ngày.
- Chức năng đăng kí, đăng nhập.

4.2.2 Yêu cầu phi chức năng

Ứng dụng cần sử dụng các thư viện có tính ổn định cao để đảm bảo ứng dụng hoạt động tốt.

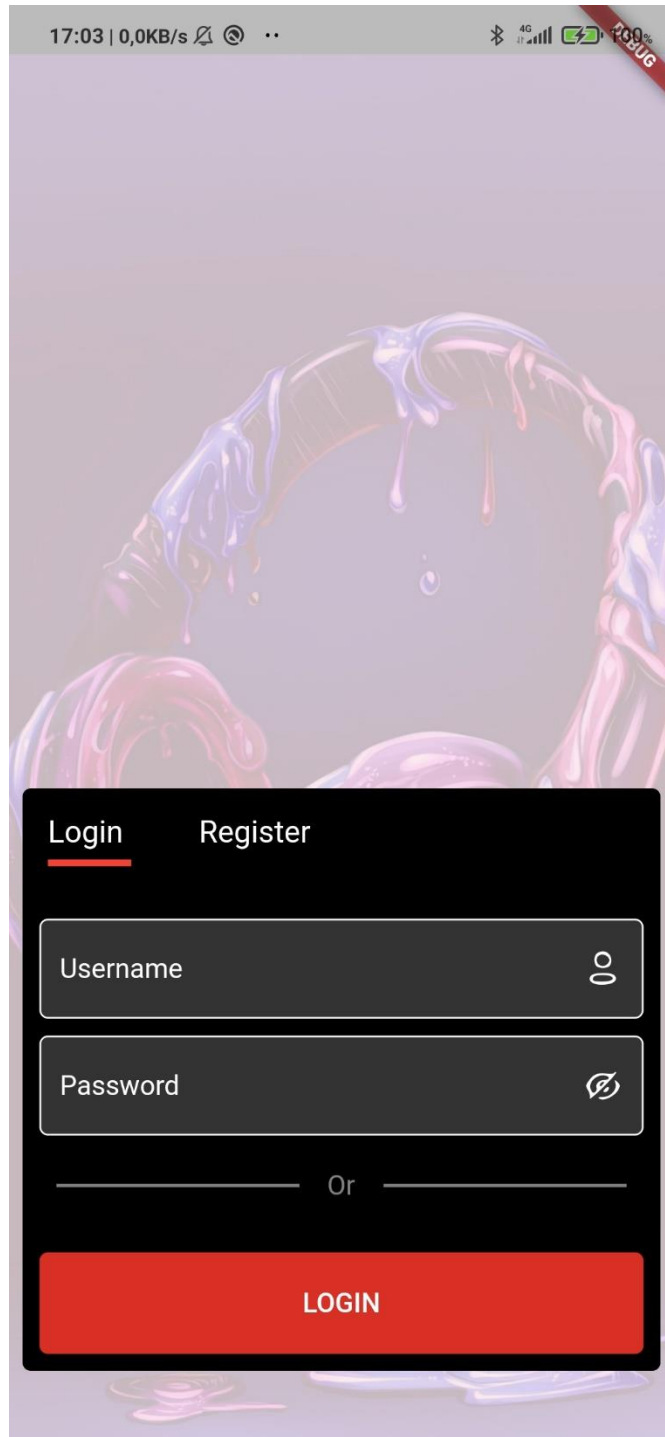
Giao diện đẹp, đơn giản, thân thiện với người dùng và dễ sử dụng. Cho phép người dùng dễ dàng tùy chỉnh các thành phần trong giao diện.

Màu sắc ứng dụng: sử dụng ram màu sáng tối phù hợp làm nổi bật lên nội dung.

Tốc độ tải, tùy chỉnh hình ảnh và lưu bài viết phải nhanh để tăng trải nghiệm người dùng.

4.3 Xây dựng giao diện ứng dụng

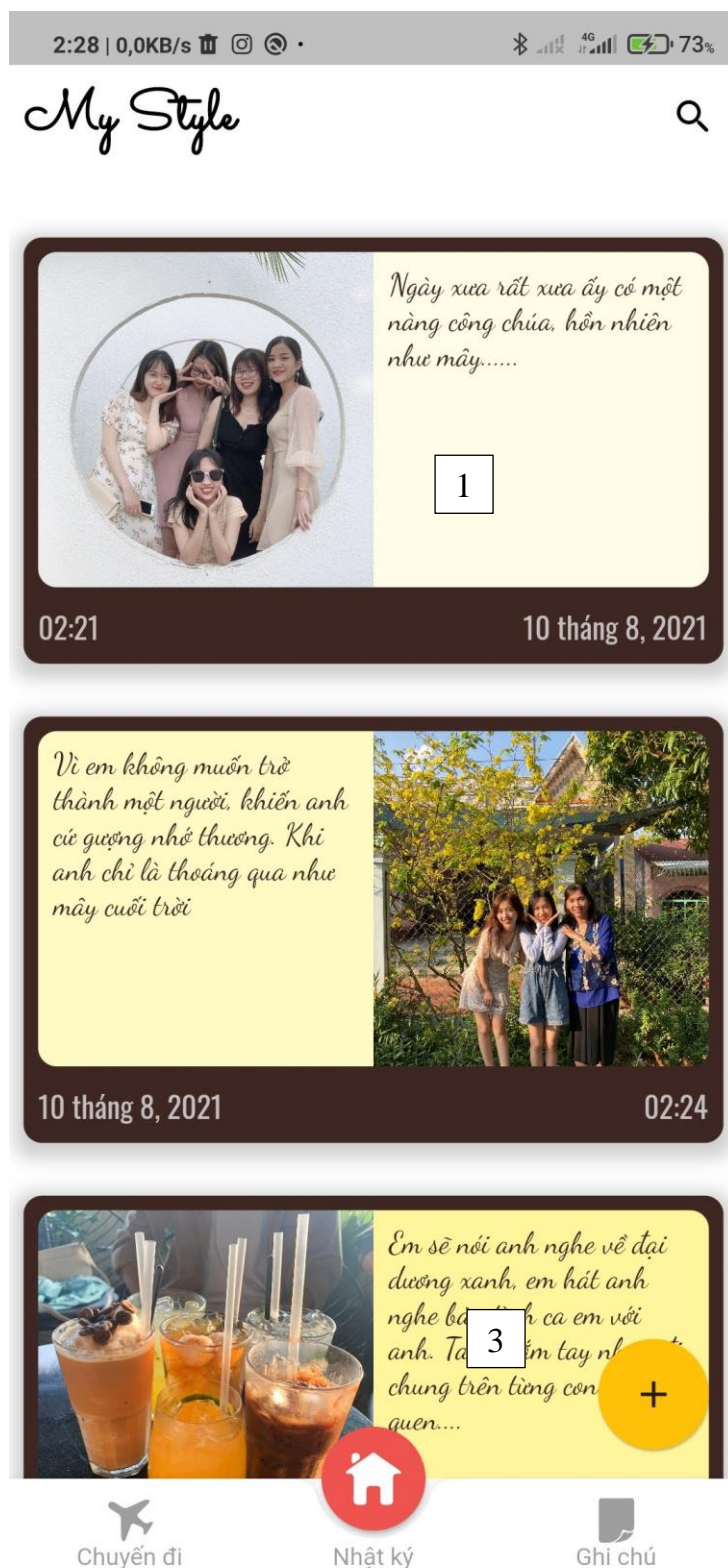
4.3.1 Màn hình đăng nhập



Hình 4.1 Màn hình đăng nhập

Giao diện cho phép người dùng đăng nhập vào hệ thống.

4.3.2 Màn hình quản lý nhật ký.



Hình 4.2 Màn hình quản lý nhật ký.

Giao diện cho phép người sử dụng quản lý nhật ký với các chức năng chính

1. Xem thông tin bài viết.
2. Tìm kiếm bài viết.

3. Tạo mới bài viết.

4. Xóa bài viết.

Người dùng nhấn giữ lâu vào bài viết muốn xóa, hộp thoại xác nhận sẽ xuất hiện, chọn đồng ý để xóa bài viết.

4.3.3 Màn hình chỉnh sửa nhật ký.



Hình 4.3 Màn hình chỉnh sửa nhật ký.

Giao diện cho phép người sử dụng chỉnh sửa nhật ký với các chức năng chính

1. Thêm hình ảnh.
2. Thêm văn bản.
3. Thêm biểu tượng cảm xúc.

4. Thêm địa chỉ.
5. Chỉnh sửa vị trí các thành phần của bài viết.

a) Chức năng thêm hình ảnh



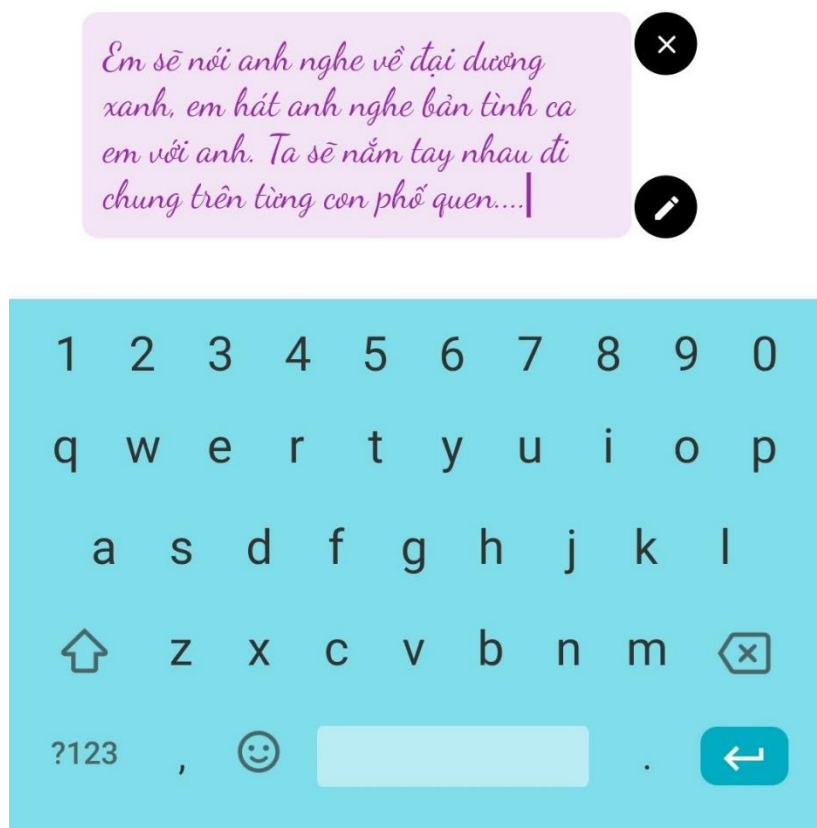
Hình 4.4 Màn hình thêm hình ảnh

b) Chức năng thêm biểu tượng cảm xúc



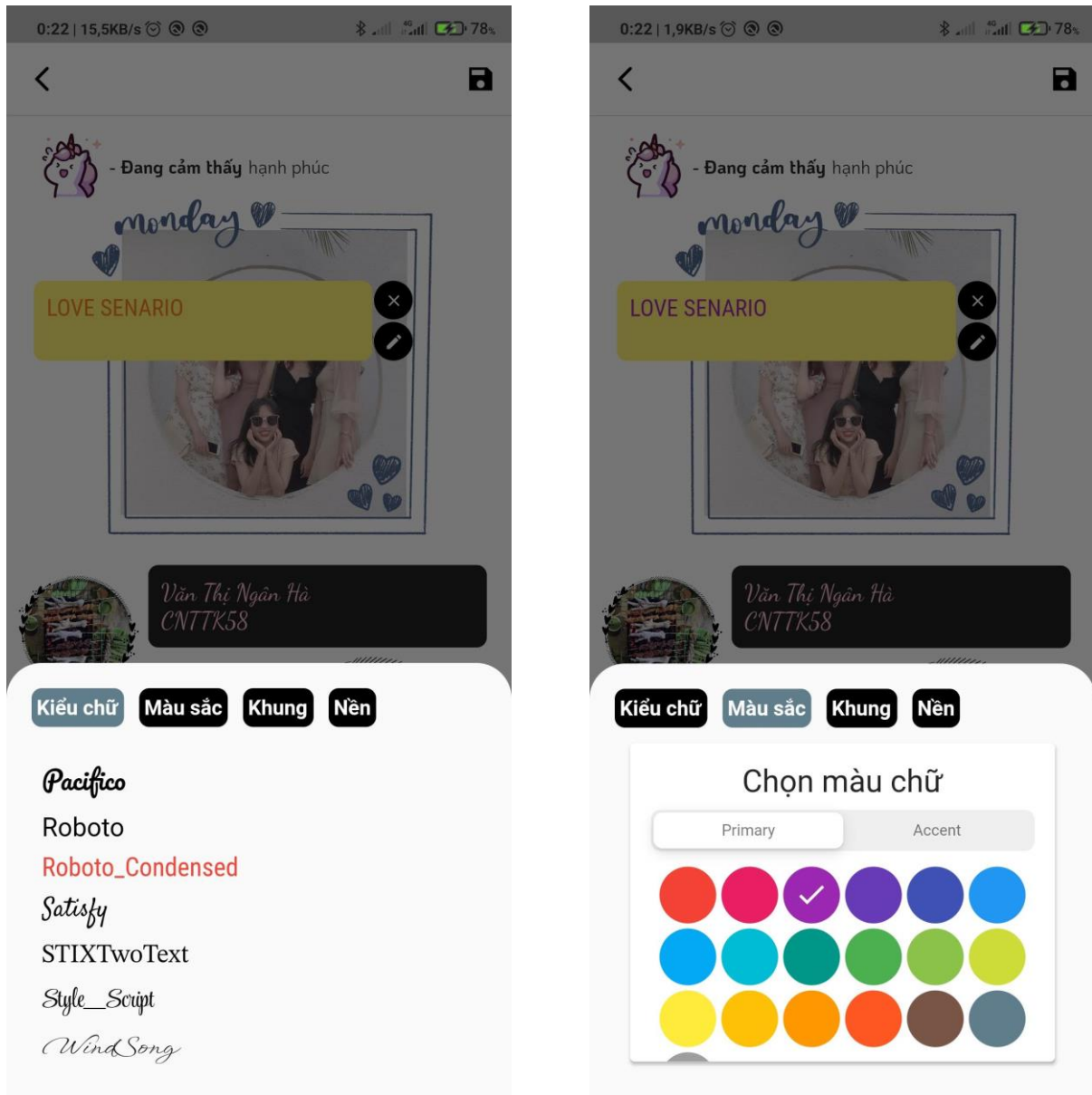
Hình 4.5 Giao diện thêm biểu tượng cảm xúc

c) Chức năng thêm văn bản.



Hình 4.6 Giao diện thêm văn bản

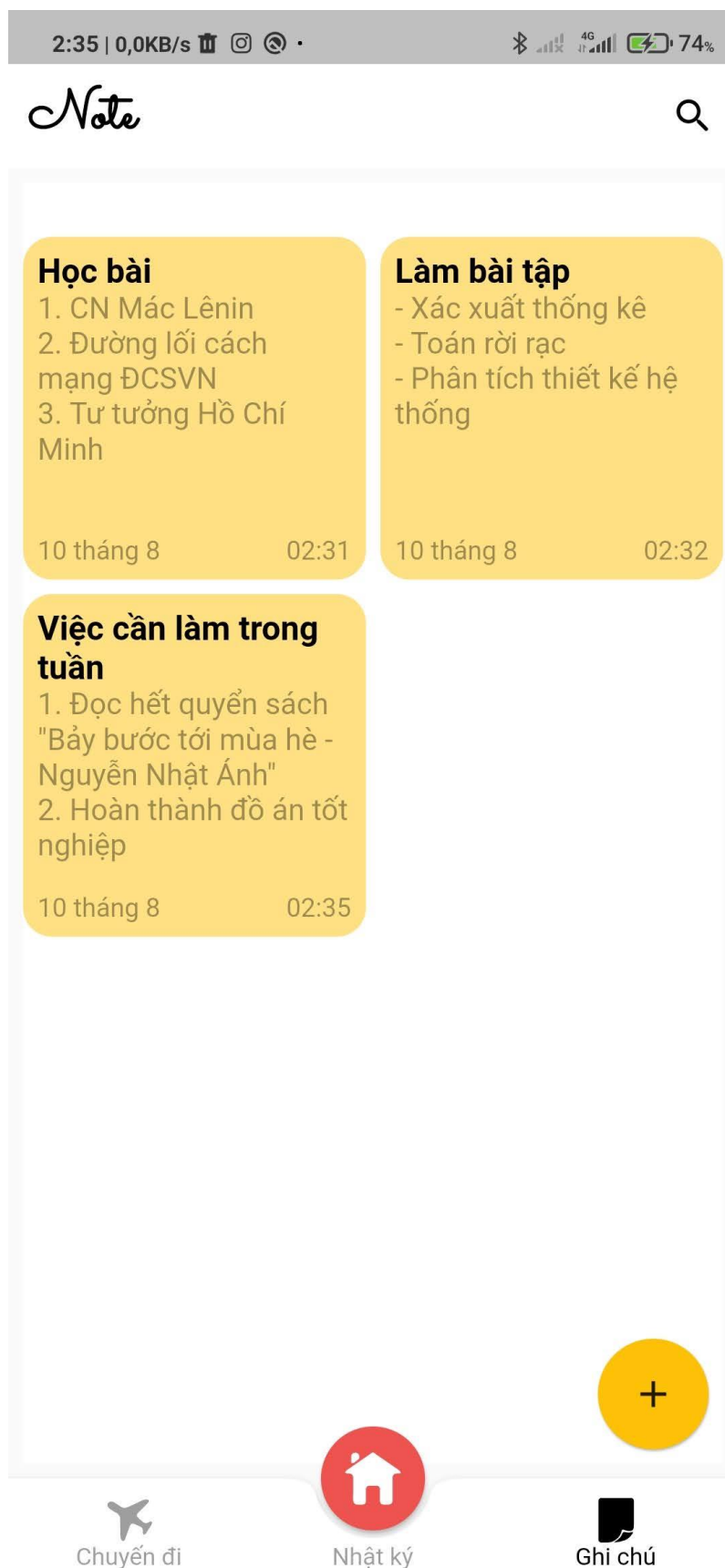
d) Chức năng chỉnh sửa phông chữ, màu sắc.



Hình 4.7 Giao diện thay đổi phông chữ, màu sắc.

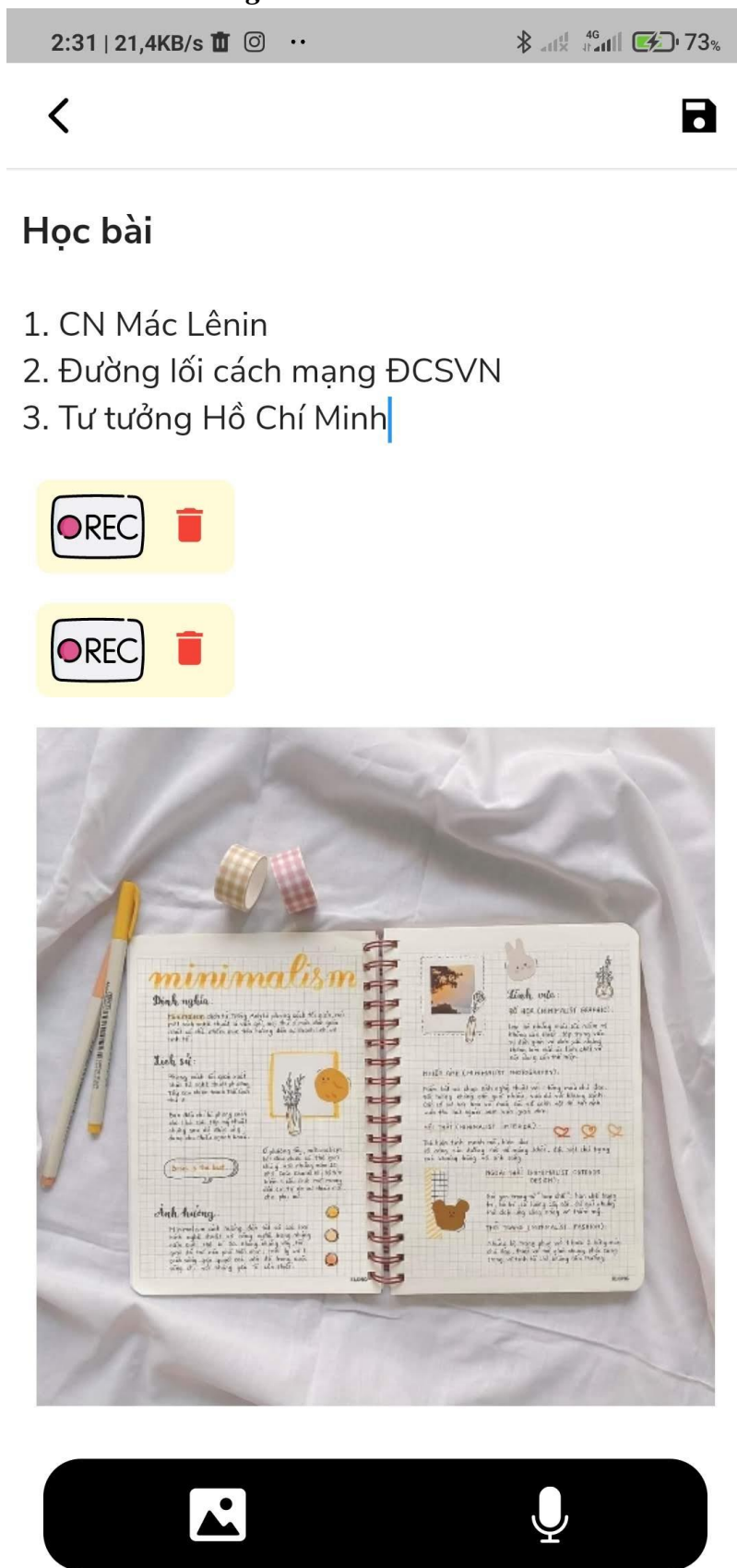
Giao diện cho phép người dùng tùy chỉnh phông chữ, màu chữ, màu nền theo ý thích.

4.3.4 Màn hình quản lý ghi chú



Hình 4.8 Màn hình quản lý ghi chú

4.3.5 Màn hình chỉnh sửa ghi chú



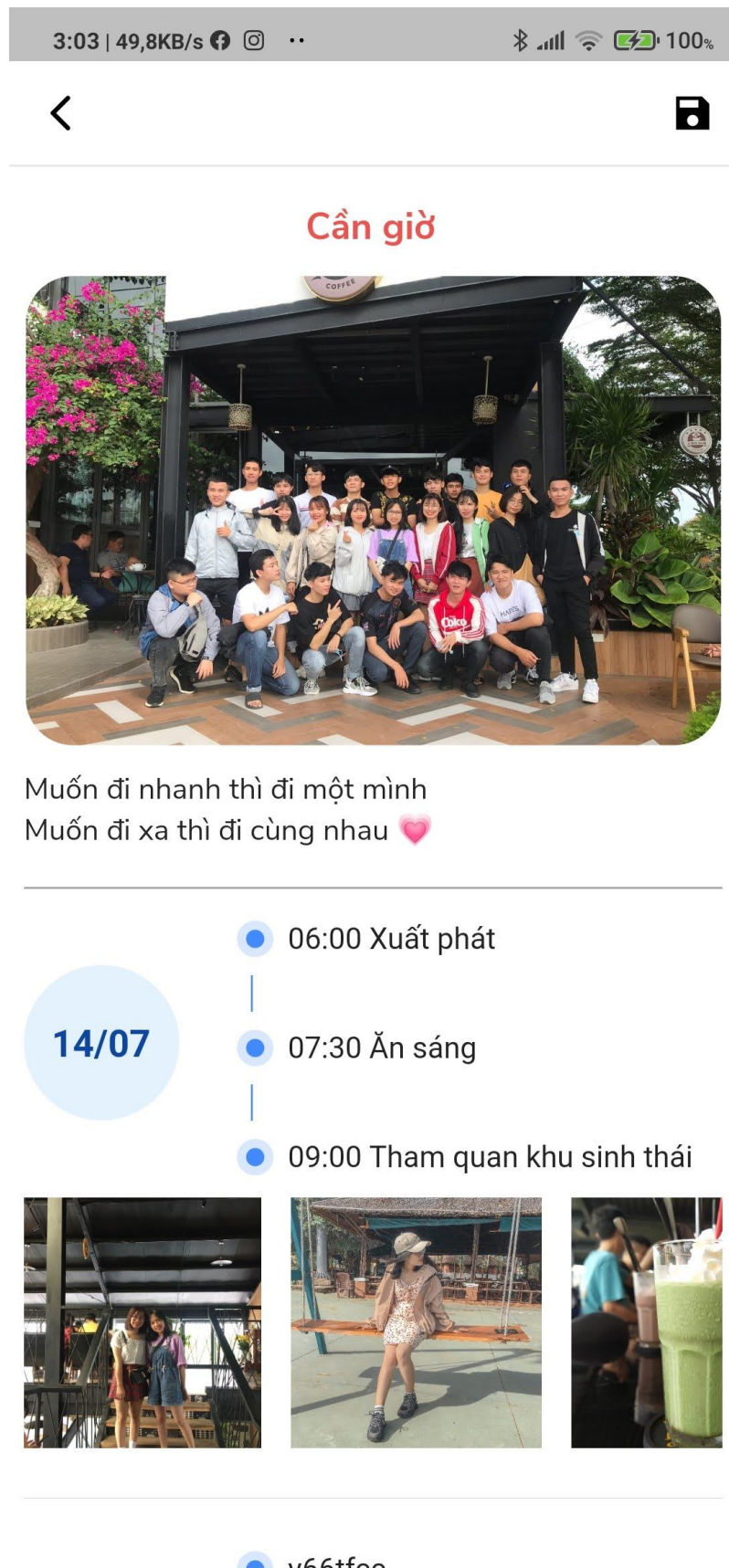
Hình 4.9 Màn hình chỉnh sửa ghi chú.

4.3.6 Màn hình quản lý chuyến đi

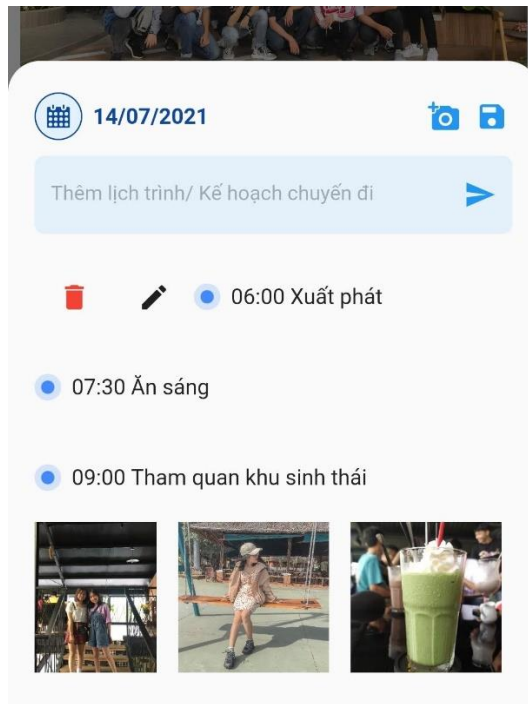


Hình 4.10 Màn hình quản lý chuyến đi.

4.3.7 Màn hình chỉnh sửa chuyến đi.



Hình 4.11 Màn hình chỉnh sửa chuyến đi.

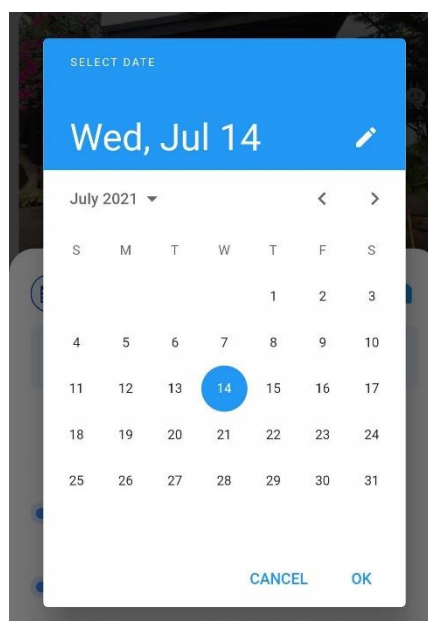


Hình 4.12 Giao diện chỉnh sửa lịch trình.

Giao diện hỗ trợ người dùng chỉnh sửa lịch trình với các chức năng:

- Chỉnh sửa ngày diễn ra lịch trình.
- Chỉnh sửa các lịch trình trong ngày.
- Chỉnh sửa hình ảnh.

Khi người dùng chọn vào biểu tượng lịch, ứng dụng sẽ hiển thị của sổ giúp người dùng dễ dàng trong việc chọn ngày.



Hình 4.13 Cửa sổ chọn ngày.

KẾT LUẬN

Kết quả đạt được

Sau quá trình tìm hiểu, nguyên cứu, thực hiện xây dựng ứng dụng, đề tài đã được những yêu cầu đã đặt ra ở mục tiêu đề ra:

- Xây dựng được ứng dụng viết nhật ký theo phong cách Bullet Journal với các chức năng:
 - Chức năng viết nhật kí: có thể tùy chỉnh thêm hình ảnh, văn bản, biểu tượng cảm xúc và địa chỉ. Người dùng có thể di chuyển các thành phần này theo bố cục mà mình mong muốn.
 - Chức năng ghi chú: người sử dụng cũng có thể tùy chọn thêm hình ảnh và ghi âm.
 - Chức năng ghi lại hành trình chuyến đi: có thể dùng chức năng này để lập kế hoạch du lịch hoặc lưu lại các chuyến đi đã qua.
- Ứng dụng chạy ổn định trên các nền tảng với giao diện thân thiện, dễ sử dụng.

Nhược điểm

- + Chức năng chỉnh sửa nhật kí chưa nhiều tùy biến để có thể tạo một bài viết có bố cục đẹp và nổi bật giống trong thực tế.
- + Dung lượng ứng dụng còn khá lớn.
- + Chức năng thêm ảnh trong ứng dụng còn phụ thuộc nhiều vào thư viện chưa thể tùy chỉnh theo mong muốn.

Hướng phát triển

- Khắc phục những nhược điểm, hạn chế để chức năng ứng dụng được hoàn thiện hơn và phù hợp hơn với nhu cầu thực tế.
- Thêm nhiều tùy chỉnh để ứng dụng có thể tạo những trang nhật ký như một quyển Bullet Journal thực thụ.
- Thêm tính năng chia sẻ và bình luận để người dùng có thể chia sẻ bài viết của mình cho người khác.

TÀI LIỆU THAM KHẢO

- [1] V. V. Chuyên, "VIBLO," 26 03 2017. [Online]. Available: <https://viblo.asia/p/tim-hieu-ionic-framework-jvElaLq6Zkw>. [Accessed 15 06 2021].
- [2] Akkie, "it-swarm-vi.com," 07 03 2014. [Online]. Available: <https://www.it-swarm-vi.com/vi/angularjs/nhung-nhuoc-diem-khi-su-dung-ionic-framework-hoac-onsen-ui-chi-la-ung-dung-web-duoc-luu-tru/1044913074/>. [Accessed 15 06 2017].
- [3] "TOPDev," [Online]. Available: <https://topdev.vn/blog/flutter-la-gi/>. [Accessed 16 06 2021].
- [4] V. V. Anh, "VIBLO," 21 04 2019. [Online]. Available: <https://viblo.asia/p/tim-hieu-ve-ngon-ngu-dart-phan-i-bJzKmykwK9N>. [Accessed 16 06 2021].
- [5] Hannah, "CodeLearn," 30 12 2020. [Online]. Available: <https://codelearn.io/sharing/why-flutter-mobile-app-2021>. [Accessed 17 06 2021].
- [6] Nghĩa, "VIBLO," 22 02 2020. [Online]. Available: <https://viblo.asia/p/thiet-ke-ung-dung-web-software-architecture-101-4dbZNLaq5YM>. [Accessed 17 06 2021].
- [7] R. J. ., J. V. Richard Helm, "Design Patterns," in *Design Patterns: Elements of Reusable Object-Oriented Software*, United States, Addison-Wesley, 1994, p. 395.
- [8] N. T. Minh, "VIBLO," 31 08 2020. [Online]. Available: <https://viblo.asia/p/hoc-bloc-pattern-theo-cach-de-hieu-nhat-maGK7JYO5j2>. [Accessed 17 06 2021].
- [9] B. t. E. Team, "ELLE," 06 03 2021. [Online]. Available: <https://www.elle.vn/bi-quyet-song/phuong-phap-ghi-bullet-journal>. [Accessed 18 06 2021].