

TRƯỜNG ĐẠI HỌC NGOẠI NGỮ TIN HỌC HUFLIT
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN HỌC PHẦN

Các công nghệ mới trong phát triển phần mềm

Đề tài:

ỨNG DỤNG ĐẶT ĐỒ ĂN VỚI FLUTTER VÀ DART

GVHD: TS. Thái Thanh Thảo

Nhóm Sinh viên:

1. Nguyễn Minh Đức Khôi 17DH111108
2. Đỗ Duy Khang 17DH111102

Thành phố Hồ Chí Minh – 2021

LỜI CẢM ƠN

Để hoàn thành được niên luận với đề tài “Ứng dụng bán đồ ăn với Flutter và Dart”, bên cạnh sự nỗ lực của bản thân đã vận dụng những kiến thức tiếp thu được, tìm tòi học hỏi em luôn nhận được sự giúp đỡ tận tình từ phía gia đình, nhà trường, thầy cô, bạn bè. Trên thực tế không có sự thành công nào không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp. Lời đầu tiên em xin gửi lời cảm ơn chân thành đến gia đình, người thân, những người đã luôn ở bên cạnh em ủng hộ, giúp đỡ em có thời gian nghiên cứu đề tài và hết lòng hỗ trợ em trong suốt thời gian thực hiện đề tài.

Bên cạnh đó, em cũng xin gửi lời cảm ơn đến trường Đại học HUFLIT, khoa Công Nghệ Thông Tin, nơi đã tạo mọi điều kiện tốt nhất để em được học tập, rèn luyện, trao dồi bản thân. Cuối cùng, em xin gửi lời cảm ơn đến toàn thể quý thầy cô bộ môn cũng như quý thầy cô trong khoa đã tận tình chỉ bảo, giảng dạy để em có những kiến thức vững chắc làm nền tảng hoàn thành thật tốt đề tài.

Với lòng biết ơn sâu sắc, em xin gửi lời cảm ơn đến cô Thảo người đã trực tiếp hướng dẫn, chỉ bảo và giúp đỡ em trong suốt thời gian thực hiện đề tài này. Bước đầu đi vào thực tế của em còn hạn chế và nhiều ngỡ ngàng. Do vậy, không tránh khỏi những thiếu sót, em rất mong nhận được những ý kiến đóng góp của cô.

Em xin chân thành cảm ơn!

TP.HCM, Ngày 19 tháng 06 năm 2021

Nguyễn Minh Đức Khôi

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and extend across the width of the page. There are no margins, text, or other markings on the paper.

Giáo viên hướng dẫn

MỤC LỤC

LỜI CẢM ƠN.....	1
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	2
MỤC LỤC.....	3
TÓM TẮT	1
CHƯƠNG 1: TỔNG QUAN	2
1.1. LÝ DO CHỌN ĐỀ TÀI.....	2
1.2. MỤC TIÊU CỦA ĐỀ TÀI	3
1.3. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU CỦA ĐỀ TÀI.....	4
Đối tượng ứng dụng của đề tài.....	4
Phạm vi nghiên cứu của đề tài	4
Về mặt lý thuyết.....	4
Về mặt lập trình.....	4
1.4. Ý NGHĨA THỰC TIỄN CỦA ĐỀ TÀI.....	6
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	8
2.1. SƠ LƯỢC VỀ ANDROID STUDIO	8
2.2. FLUTTER.	8
Khái niệm.	8
Flutter hoạt động như thế nào ?	9
2.3. NGÔN NGỮ DART.....	9
Khái niệm.	9
2.4. Mô hình bloc pattern.....	10
2.4 .GOOGLE FIREBASE	14
Khái niệm	14
Tại sao lại là Firebase?	14
2.5. Net 5 API (SWAGGER_JWT_OPEN API).....	14

JWT Authentication and Swagger with .NET	14
1. OpenAPI là gì.....	14
2. Swagger là gì	15
3. Cấu trúc cơ bản.....	15
Web API trong asp.net core	17
2.6. ChatBot.....	19
CHƯƠNG 3: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN	22
3.0. THU VIỆN SỬ DỤNG	22
3.1. THIẾT KẾ CƠ SỞ DỮ LIỆU	23
3.2. Mô hình hoá yêu cầu.....	25
3.2.1. Xác định actor:	25
3.2.2. Xác định Use Case.....	25
1.1.2. Use Case Tổng quát:.....	27
3.2.3. Phân tích – Thiết kế chức năng.....	29
Đăng nhập	29
Đăng ký.....	31
Xem danh sách món ăn	33
Thêm giỏ hàng.....	34
Quản lý giỏ hàng.....	35
Thanh Toán	37
Quản lý món ăn.....	38
Xem danh sách hoá đơn	41
Xem Thống Kê	42
3.2.4. Cài đặt demo và giải thích một số chức năng chính	43
a. Cài đặt.....	43
b. Demo giải thích các chức năng chính	43
CHƯƠNG 4: KẾT LUẬN	59

4.1.	CÁC KẾT QUẢ ĐẠT ĐƯỢC.....	59
4.2.	HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN.....	59
	Hạn chế.....	59
	Hướng phát triển.....	59
	TÀI LIỆU THAM KHẢO	60

TÓM TẮT

Đề tài “Ứng dụng bán đồ ăn trên nền tảng di động” là một ứng chương trình phục vụ nhu cầu mua sắm một cách nhanh chóng, an toàn và tiện lợi, đồng thời đề tài còn hướng đến sự tiện lợi trong công tác quản lý bán hàng bằng việc xây dựng 1 app song song cho việc quản lý trên website, và điều đặc biệt nhất là bằng 1 ngôn ngữ lập trình có thể tạo ra ứng dụng chạy trên cả 2 nền tảng di động phổ biến nhất hiện nay chính là Android và IOS. Để giải quyết vấn đề nêu trên, đề tài phải tập trung giải quyết 3 vấn đề lớn như sau : Nghiên cứu, thiết kế cơ sở dữ liệu hoàn chỉnh, xây dựng ứng dụng cho khách hàng kết hợp với ứng dụng quản lý để thực hiện truy xuất cơ sở dữ liệu đảm bảo các tính năng cần thiết cho việc quản lý. Xuyên suốt đề tài người thực hiện sử dụng ngôn ngữ lập trình Dart với Flutter mobile UI Framework, dịch vụ cơ sở dữ liệu qua API của của hàng cùng với các thư viện Pub package cho việc hoàn thành toàn bộ dự án cụ thể như sau:

- Phân tích đặt tả hệ thống, đặt tả chức năng, xây dựng các mô hình cơ sở dữ liệu. Kết quả của bước này là cấu trúc một cơ sở dữ liệu hoàn chỉnh.
- Xây dựng ứng dụng quản lý thực hiện việc nhập xuất hình thành cấu trúc cơ sở dữ liệu như mô hình đã đề cập trên API.
- Tiến hành thiết kế giao diện cho toàn bộ ứng dụng, truy xuất cơ sở dữ liệu trên API để hiển thị chúng. Kết quả của bước này là một app bán hàng hoàn chỉnh với các tính năng cơ bản và một số tính năng của một app bán hàng hiện đại.

CHƯƠNG 1: TỔNG QUAN

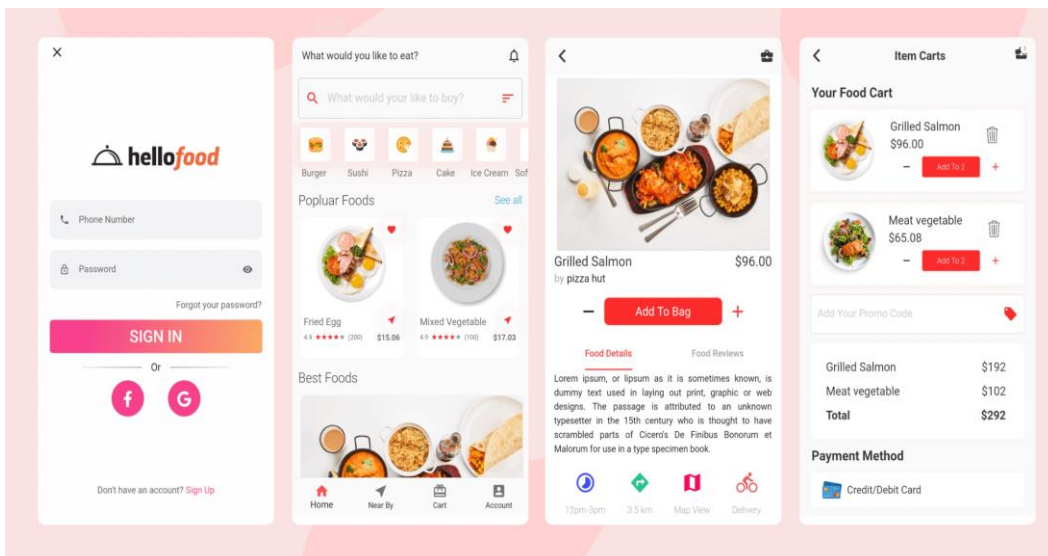
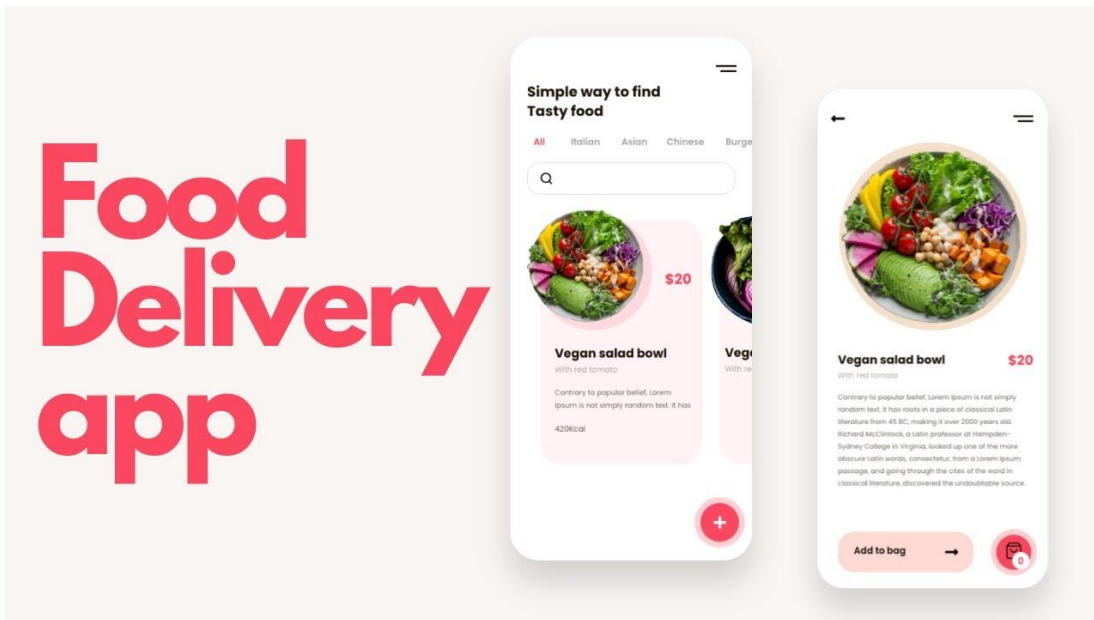
1.1. LÝ DO CHỌN ĐỀ TÀI

Trong những năm gần đây nền kinh tế cũng như tốc độ tăng trưởng kinh tế của Việt Nam dần hội nhập với nền kinh tế của các nước trong khu vực và thế giới, thu nhập bình quân của người dân dần được cải thiện và nâng cao. Với những điều kiện thuận lợi như trên thì nhu cầu về tiêu dùng, hình thức phục vụ, các dịch vụ tiện ích an toàn cùng được mọi người ngày càng chú ý đến. Cùng với xu thế đó, bán hàng online là một giải pháp hoàn toàn phù hợp với nhu cầu hiện tại của người tiêu dùng. Sự cạnh tranh giữa các nhà cung cấp, cửa hàng ngày càng trở nên gay gắt. Với mục đích đáp ứng nhu cầu của khách hàng, cung cấp dịch vụ nhằm phát triển kinh doanh, hình thức bán hàng trực tuyến trở nên phát triển, thu hút lượng khách lớn, phạm vi phục vụ rộng rãi, hình thức quảng cáo đơn giản, tiện sử dụng, dễ dàng cập nhật thông tin, và có thể đáp ứng nhu cầu khách hàng vào bất cứ thời gian truy cập nào. Bên cạnh đó, công tác quản lý cũng hết sức cần thiết đối với nhà bán hàng. Có thể quản lý hàng hóa, hóa đơn, xuất nhập kho một cách thuận tiện dễ dàng, chính xác là điều mà mọi nhà bán hàng đều hướng tới. Đây cũng chính là lý do chính em đã chọn đề tài “Ứng dụng bán hàng trên di động” làm cho việc mua sắm cũng như quản lý trở nên đơn giản, chính xác, tiết kiệm thời gian của người mua hàng/bán hàng.

Cùng với đó là sự phát triển của các hệ điều hành lớn như Android và IOS, việc tạo một ứng dụng di động phải đáp ứng cho các nền tảng điều hành lớn này để đạt tối đa hiệu quả của ứng dụng tốt nhất. Một trong những công cụ đáp ứng được nhu cầu trên là Flutter. Công cụ này cho phép tạo ứng dụng di động gốc (Native App) với một codebase, nghĩa là chỉ sử dụng một ngôn ngữ lập trình và một codebase để tạo hai ứng dụng khác nhau (cho Android và cả IOS). Vì thế, Flutter lựa chọn tốt để phát triển ứng dụng di động.

1.2. MỤC TIÊU CỦA ĐỀ TÀI

Chương trình được tạo ra cho phép người tiêu dùng mua hàng, đặt hàng thông qua ứng dụng được cài đặt trên điện thoại di động một cách dễ dàng, có thể thực hiện ở bất kì đâu, bất cứ lúc nào. Ngoài ra, chương trình còn đáp ứng nhu cầu quản lý hàng hóa, đơn hàng, xuất nhập hàng hóa. Ứng dụng quản lý cũng được thiết kế trên nền tảng di động nên dễ quản lý ở mọi nơi để truy vấn đến dịch vụ cơ sở dữ liệu API thay đổi, chỉnh sửa chúng. Một số hình ảnh App design về mô hình food order



1.3. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU CỦA ĐỀ TÀI

Đối tượng ứng dụng của đề tài

Tất cả các cá nhân, doanh nghiệp có nhu cầu kinh doanh online, mở rộng tập khách hàng, thay đổi phương thức phục vụ kinh doanh, phục vụ quản lý.

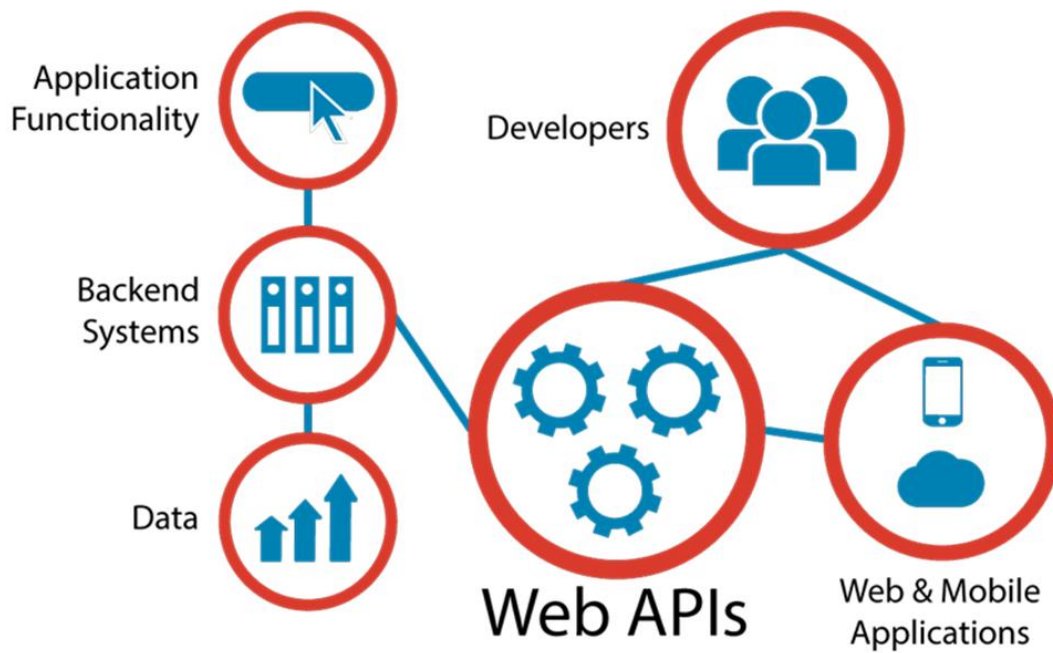
Phạm vi nghiên cứu của đề tài

Về mặt lý thuyết

Tìm hiểu cách sử dụng API qua các thư viện như HTTP, DIO,..., tìm hiểu dịch vụ cơ sở dữ liệu Firebase của Google, kỹ thuật lập trình Flutter và các thư viện hỗ trợ trong Pub package, tìm hiểu nghiên cứu các thuộc tính đối tượng trong Flutter, ngôn ngữ lập trình Dart.

Về mặt lập trình

- Tiến hành xây dựng giao diện website quản lý cho Admin, xử lý các logic, tuần tự, và truy vấn, cập nhật, xử lý cơ sở dữ liệu lên API để tạo và hình thành hoàn chỉnh cơ sở dữ liệu online nếu API được deploy, hiển thị dữ liệu lên giao diện để dễ quản lý đã. .
- Xây dựng giao diện mua hàng hóa cho app mua hàng dành cho client xử lý các tính năng quản lý như thêm, sửa, xóa hàng hóa, xác nhận hóa đơn, nhắn tin tư vấn khách hàng, lịch sử mua hàng, địa chỉ giao hàng,...



Hình 1.1: Xử lý dữ liệu từ Client lên Firebase so với xử lý dữ liệu truyền thống.

1.4. Ý NGHĨA THỰC TIỄN CỦA ĐỀ TÀI

Trong xã hội hiện đại và phát triển như ngày nay thì nhu cầu tiêu dùng mua sắm nhất là order đồ ăn tăng cao cũng như công tác quản lý tăng cao và đòi hỏi càng khắc khe hơn về các dịch vụ tiện ích hỗ trợ. Do đó, sự ra đời của “Ứng dụng bán hàng online” là hết sức cần thiết và đã mang lại những ý nghĩa thực tiễn trong đời sống như sau:

- Trong việc phục vụ tiêu dùng mua sắm của khách hàng (website):
 - ❖ Ở bất cứ đâu, bất kể lúc nào nếu có mạng internet khách hàng hoàn toàn có thể đặt đồ ăn cho bản thân mà không cần đích thân ra ngoài. Chỉ cần click chọn sản phẩm và đặt hàng là sản phẩm sẽ được chuyển đến khách hàng một cách nhanh nhất.
 - ❖ Khách hàng có thể lựa chọn món ăn, xem thông tin món ăn, giá cả món ăn một cách dễ dàng bởi việc món ăn được sắp xếp theo loại và trình bày một cách rõ ràng giúp khách hàng tìm kiếm món ăn rất thuận tiện, ngoài ra công cụ tìm kiếm hỗ trợ mạnh mẽ việc tìm kiếm những món ăn theo yêu cầu của khách hàng.
 - ❖ Khách hàng hoàn toàn có thể chủ động kiểm tra toàn bộ quá trình từ đặt hàng, gói hàng, vận chuyển cho đến lúc thanh toán bằng tính năng kiểm tra đơn hàng. Ở đây trạng thái đơn hàng sẽ được cập nhật liên tục giúp khách hàng có thể kiểm soát được đơn hàng của mình qua lịch sử đơn hàng.
- Trong công tác quản lý của cá nhân, doanh nghiệp kinh doanh:
 - ❖ Thông tin hàng hóa, đơn hàng hoàn toàn được bảo mật bởi việc xác thực mật khẩu mỗi khi đăng nhập vào website quản trị. Giúp hạn chế tình trạng lộ thông tin khách hàng, hàng hóa và thu chi của doanh nghiệp cá nhân.
 - ❖ Cá nhân, doanh nghiệp có thể quản lý toàn bộ sản phẩm hàng hóa, khách hàng, đơn đặt hàng, khuyến mãi, xuất nhập hàng,... rất thuận tiện. Ứng dụng quản trị cung cấp đầy đủ các tính năng cơ bản để quản lý như đăng nhập, xem, cập nhật, thêm sửa xóa dữ liệu, tìm kiếm, thống kê.



CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. SƠ LƯỢC VỀ ANDROID STUDIO

Android Studio là một phần mềm bao gồm các bộ công cụ khác nhau dùng để phát triển ứng dụng chạy trên thiết bị sử dụng hệ điều hành Android như các loại điện thoại smartphone, các tablet... Android Studio được đóng gói với một bộ code editor, debugger, các công cụ performance tool và một hệ thống build/deploy (trong đó có trình giả lập simulator để giả lập môi trường của thiết bị điện thoại hoặc tablet trên máy tính) cho phép các lập trình viên có thể nhanh chóng phát triển các ứng dụng từ đơn giản tới phức tạp.

Việc xây dựng một ứng dụng mobile (hoặc tablet) bao gồm rất nhiều các công đoạn khác nhau. Đầu tiên chúng ta sẽ viết code ứng dụng sử dụng máy tính cá nhân hoặc laptop. Sau đó chúng ta cần build ứng dụng để tạo file cài đặt. Sau khi build xong thì chúng ta cần copy file cài đặt này vào thiết bị mobile (hoặc table) để tiến hành cài đặt ứng dụng và chạy kiểm thử (testing). Bạn thử tưởng tượng nếu với mỗi lần viết một dòng code bạn lại phải build ứng dụng, cài đặt trên điện thoại hoặc tablet và sau đó chạy thử thì sẽ vô cùng tốn thời gian và công sức. Android Studio được phát triển để giải quyết các vấn đề này. Với Android Studio tất cả các công đoạn trên được thực hiện trên cùng một máy tính và các quy trình được tinh gọn tới mức tối giản nhất.

2.2. FLUTTER.

Khái niệm.

Flutter là mobile UI framework của Google để tạo ra các giao diện chất lượng cao trên iOS và Android trong khoảng thời gian ngắn. Flutter hoạt động với những code sẵn có được sử dụng bởi các lập trình viên, các tổ chức.

Flutter hoàn toàn miễn phí và cũng là mã nguồn mở.

Flutter có các thành phần UI của riêng nó, cùng với một cơ chế để kết xuất chúng trên nền tảng Android và iOS. Hầu hết các thành phần giao diện người dùng, đều sẵn dùng, phù hợp với các nguyên tắc của Material Design.

Flutter hoạt động như thế nào ?

Các ứng dụng Flutter không biên dịch trực tiếp với các ứng dụng native của Android và iOS. Thay vào đó, chúng chạy trên engine render Flutter (được viết bằng C++) và Flutter Framework (được viết bằng Dart, cũng như các ứng dụng Flutter), cả hai bộ này đều được đóng gói cùng với mọi ứng dụng. Sau đó bộ SDK sẽ đóng gói lại vào trong một ứng dụng để sẵn sàng chạy trên mỗi nền tảng. Bạn tạo ra ứng dụng của mình, một engine mới sẽ chạy các đoạn code của Flutter, và các đoạn code native vừa đủ để nền tảng Flutter chạy trên cả Android và iOS.

How Flutter works



<https://flutter.io/technical-overview/>

5

Hình 2.1: Cách Flutter hoạt động.

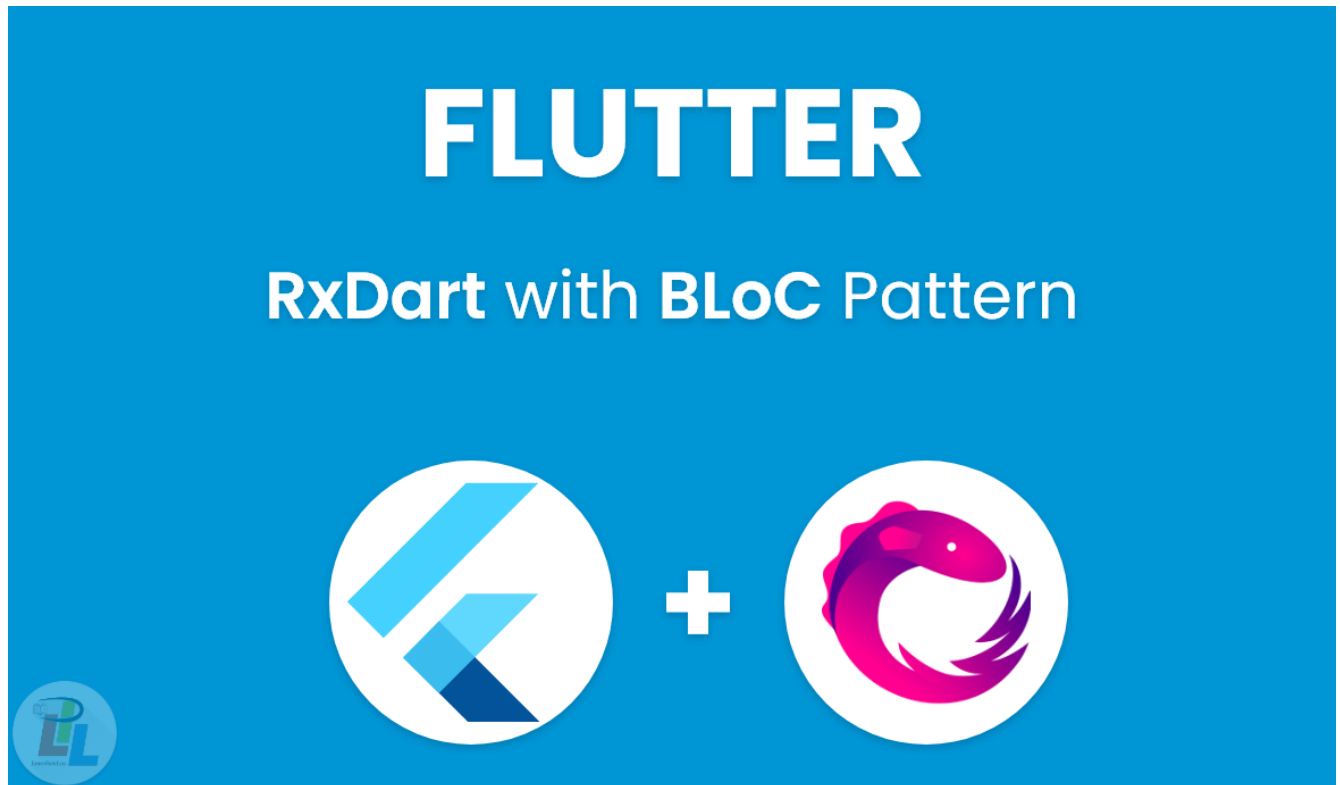
2.3. NGÔN NGỮ DART.

Khái niệm.

Dart là ngôn ngữ lập trình đa mục đích ban đầu được phát triển bởi Google, Nó

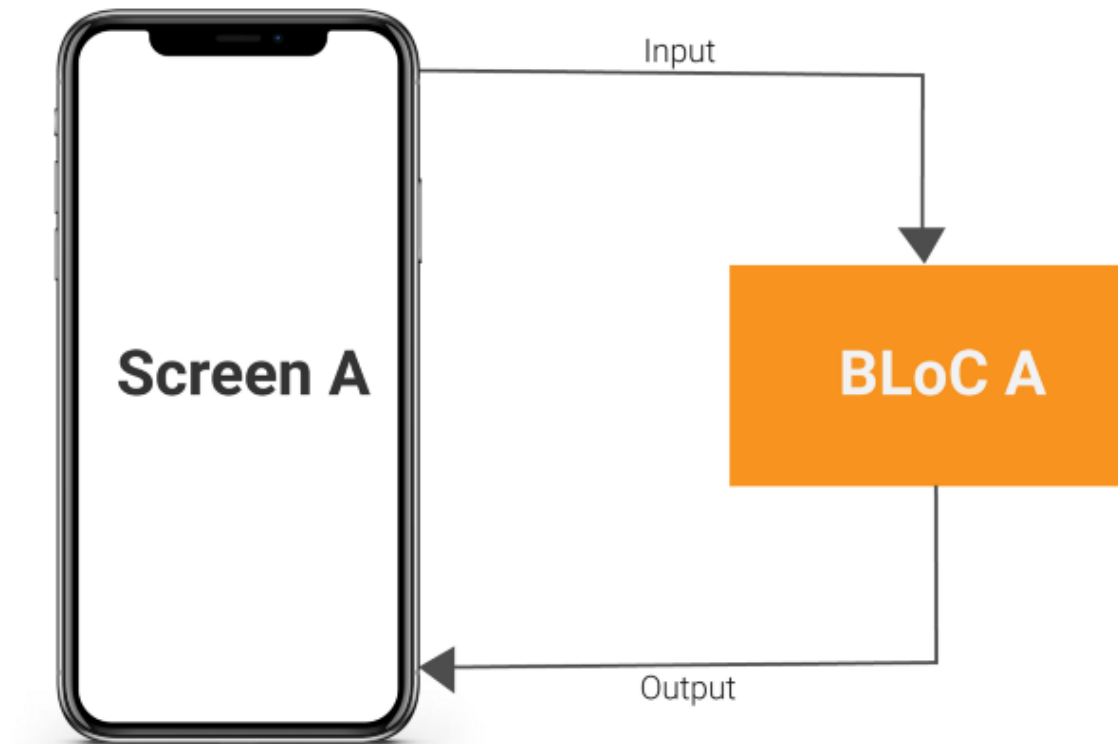
được sử dụng để xây dựng các ứng dụng web, server, máy tính để bàn và thiết bị di động. Dart là một ngôn ngữ hướng đối tượng, được xác định theo lớp, sử dụng cú pháp kiểu C để dịch mã tùy ý sang JavaScript. Dart là ngôn ngữ mã nguồn mở và miễn phí.

2.4. Mô hình bloc pattern

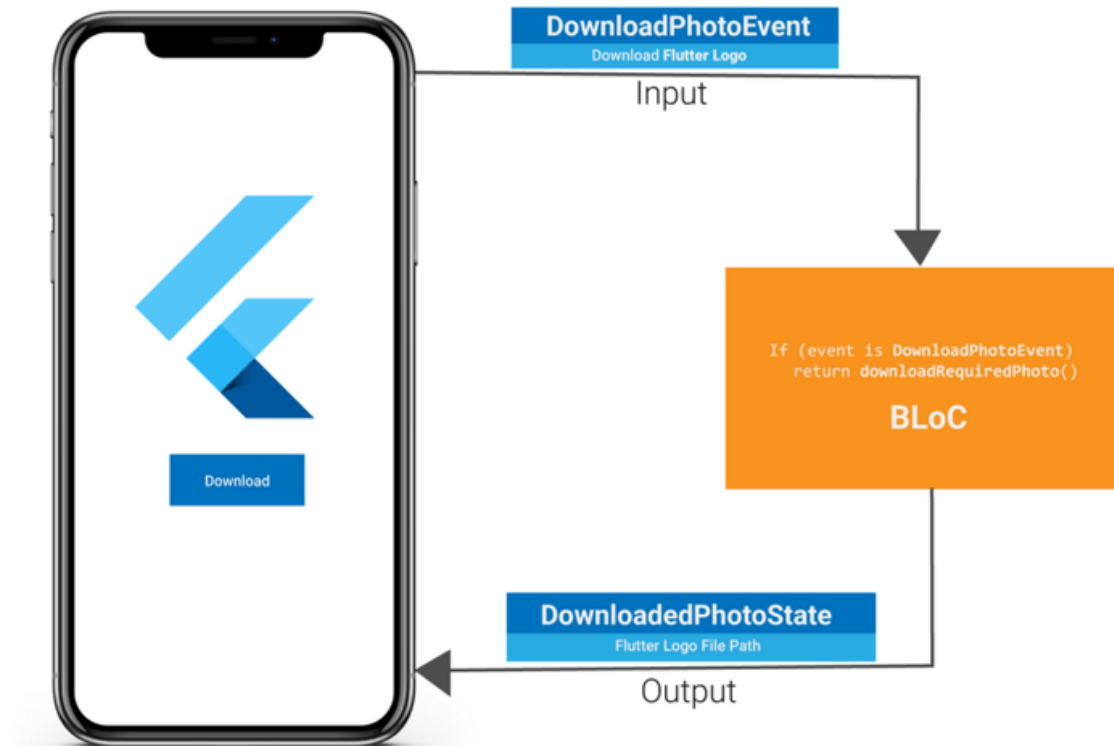


1. Bloc Pattern là gì

Tất nhiên nó là một Pattern rồi, mục đích của Pattern này là tách code business logic ra khỏi UI thay vì code gộp chung cả logic và UI vào cùng 1 file, để sau này spec mới có yêu cầu sửa code business logic hay sửa UI sẽ dễ dàng sửa hơn. Code business logic được tách ra đó người ta đặt tên là Bloc (Business Logic Component). Bên cạnh đó, nó còn giúp chúng ta quản lý state của 1 màn hình tốt hơn vì các state sẽ được quản ở Bloc tách biệt với UI. Chính vì vậy, mỗi màn hình trong app flutter chúng ta nên tạo ra 1 bloc để xử lý logic của màn hình đó và quản lý state của cả màn hình đó.



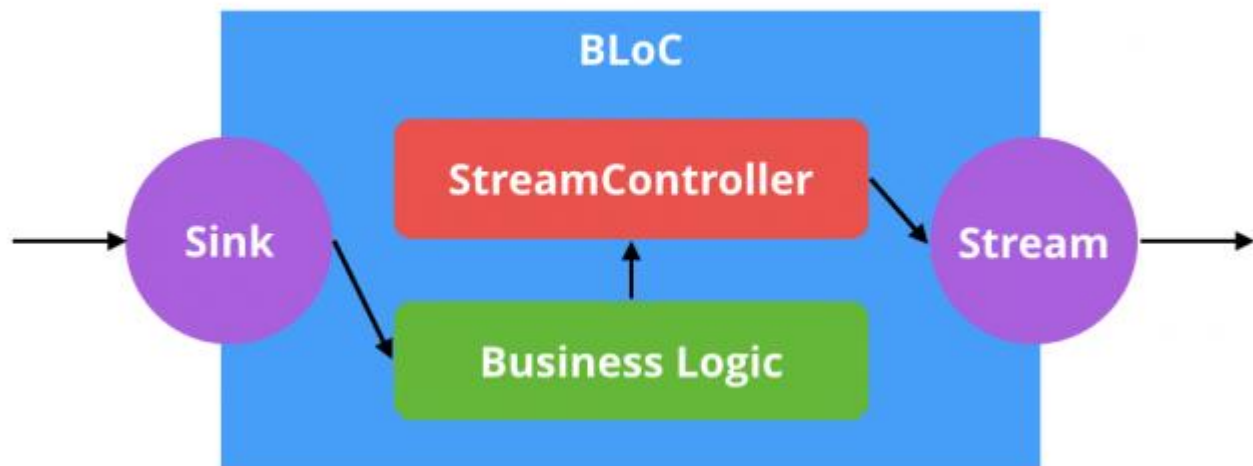
Lấy ví dụ trong dự án thực tế lun nha, khi user click vào button download tức là user gửi cái link URL của ảnh vào Bloc, Bloc sẽ nhận link URL đó và xử lý download thành cái ảnh và truyền cho UI để UI hiển thị lên màn hình.



Tóm lại, mô hình của Bloc trông sẽ thế này. Truyền event vào bloc, bloc xử lý và cho ra output là State của UI.



Ok, ý tưởng của Bloc chỉ đơn giản vậy thôi. Ý tưởng đó nó khá là phù hợp với Stream Controller. Vì vậy để triển khai ý tưởng đó, người ta sử dụng đến StreamController



Input sẽ được thêm vào sink của StreamController và phía UI sẽ sử dụng stream để lắng nghe nhận state mỗi khi có event được add vào sink.

2.4 .GOOGLE FIREBASE

Khái niệm

Google firebase là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình phát triển nhanh các ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

Tại sao lại là Firebase?

Sử dụng Firebase bạn sẽ có được hưởng các lợi ích sau:

- Xây dựng ứng dụng nhanh chóng mà không tốn thời gian, nhân lực để quản lý hệ thống và cơ sở hạ tầng phía sau: Firebase cung cấp cho bạn chức năng như phân tích, cơ sở dữ liệu, báo cáo hoạt động và báo cáo các sự cố lỗi để bạn có thể dễ dàng phát triển, định hướng ứng dụng của mình vào người sử dụng nhằm đem lại các trải nghiệm tốt nhất cho họ.
- Uy tín chất lượng đảm bảo từ Google: Firebase được google hỗ trợ và cung cấp trên nền tảng phần cứng với quy mô rộng khắp thế giới, được các tập đoàn lớn và các ứng dụng với triệu lượt sử dụng từ người dùng.
- Quản lý cấu hình và trải nghiệm các ứng dụng của Firebase tập trung trong một giao diện website đơn giản, các ứng dụng này hoạt động độc lập nhưng liên kết dữ liệu phân tích chặt chẽ.

Vì những lợi ích trên thì việc quyết định chọn Firebase để phát triển cơ sở dữ liệu cho ứng dụng là dễ dàng.

2.5. Net 5 API (SWAGGER_JWT_OPEN API)

JWT Authentication and Swagger with .NET

1. *OpenAPI* là gì

OpenAPI Specification là một định dạng mô tả API dành cho REST APIs. Một file OpenAPI cho phép bạn mô tả toàn bộ API bao gồm cả

- Cho phép những endpoints (/users) và cách thức hoạt động của mỗi endpoint (GET /users, POST /users)
- Các tham số đầu vào & đầu ra của từng hoạt động
- Phương thức xác thực
- Thông tin liên lạc, chứng chỉ, điều khoản sử dụng và những thông tin khác
- API specifications có thể được viết bằng YAML hoặc JSON. Định dạng này dễ đọc, dễ hiểu cho cả người dùng lẫn ngôn ngữ máy tính

2. Swagger là gì

Swagger là một bộ công cụ mã nguồn mở để xây dựng OpenAPI specifications giúp bạn có thể thiết kế, xây dựng tài liệu và sử dụng REST APIs

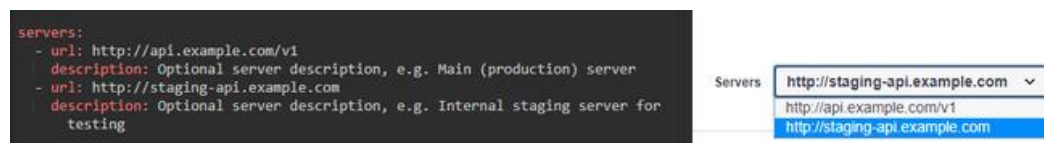
3. Cấu trúc cơ bản

3.1: Metadata

Mỗi OpenAPI specifications sẽ bắt đầu với từ khóa openapi để khai báo phiên bản (VD: openapi: 3.0.0). Phiên bản này sẽ định nghĩa toàn bộ cấu trúc của API. Phần info sẽ chứa những thông tin của API như: title, description (tùy chọn), version title là tên API của bạn description là thông tin mở rộng về API của bạn. Bạn có thể viết thành nhiều dòng & hỗ trợ cú pháp Markdown info cũng hỗ trợ những từ khóa về thông tin liên lạc, chứng chỉ, điều khoản sử dụng và những thông tin khác

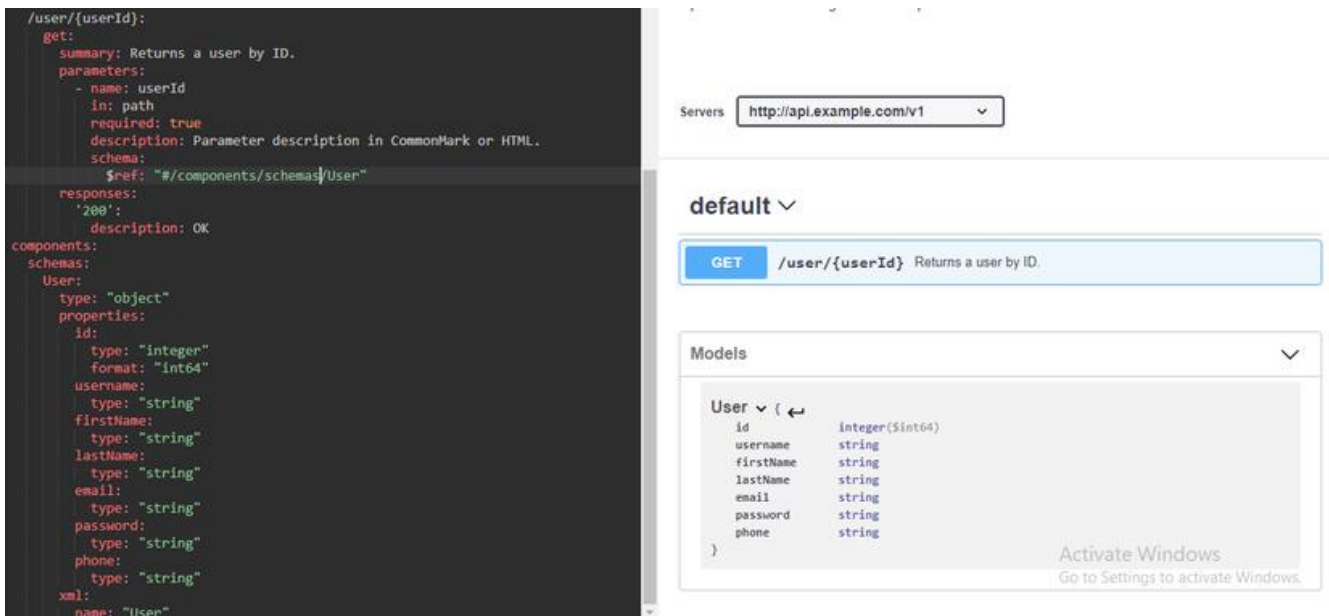
3.2: Servers

Đây là phần sẽ chỉ định đường dẫn của server để ta có thể test được API. Bạn có thể định nghĩa một hoặc nhiều server. Tất cả đường dẫn API sẽ là đường dẫn tương đối của URL mà bạn định nghĩa. Ảnh bên phải là phần UI sẽ hiển thị ra



3.3: Paths

Đây là phần trọng tâm của API. Ở phần này bạn sẽ định nghĩa những paths trong API của bạn cũng như phương thức, tham số trong API

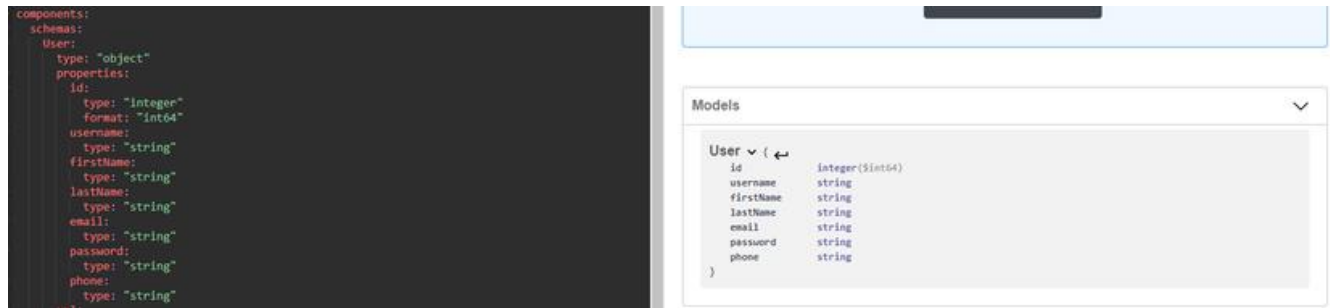


- Phần này sẽ bắt đầu bằng từ khóa paths
- Sau đó là đến những path trong API (`/user/{userId}`)
- Tiếp đến là phương thức của API (GET, POST, DELETE, PUT ...)
- summary là phần mô tả tóm tắt của API
- parameters: sẽ là những tham số truyền vào API. Bạn có thể set tham số required hay không, mô tả nó (description) hoặc validate. Đặc biệt trong phần này. bạn có thể chỉ định 1 schema (hiểu nôm na là 1 Model) để có thể định nghĩa cho phần tham số thông qua schema & \$ref
- response là phần trả về của server. Bạn có thể định nghĩa những HTTP code: 200, 404, 500 ... với những mô tả cho từng trường hợp

3.4: Schema

- Bạn có thể hiểu nôm na đây là 1 Model. Phần này được khai báo qua từ khóa component & schemas (Lưu ý: những chỗ gọi đến schema này phải chỉ định chính xác đường dẫn VD \$ref: `"#/components/schemas/User"`)

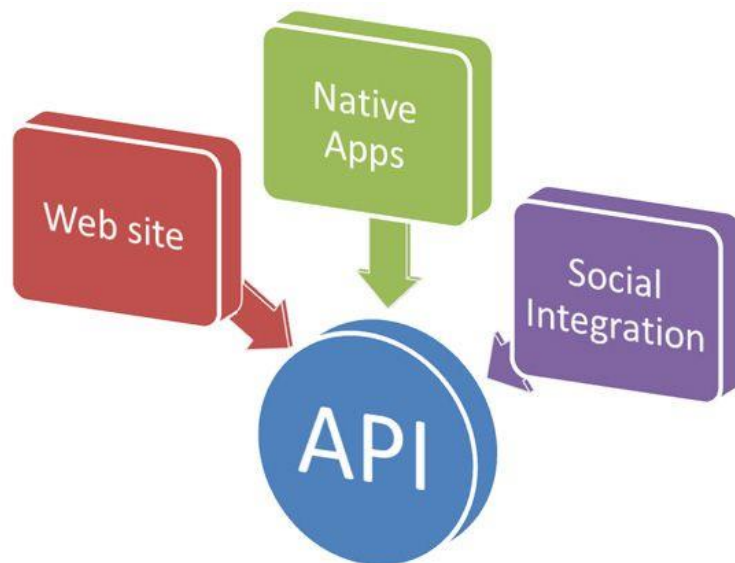
- Tham số đầu tiên là tên của Model (User)
- Tiếp đó sẽ là phân kiểu định dạng (object)
- Sau đó là phân thuộc tính của Model này
- Trên đây mình đã hướng dẫn sơ qua về những tính năng của swagger mà mình đã sử dụng trong dự án Các bạn có thể vào link để có thể viết API 1 cách tiện nhất. Nó sẽ render UI ngay lập tức cho bạn



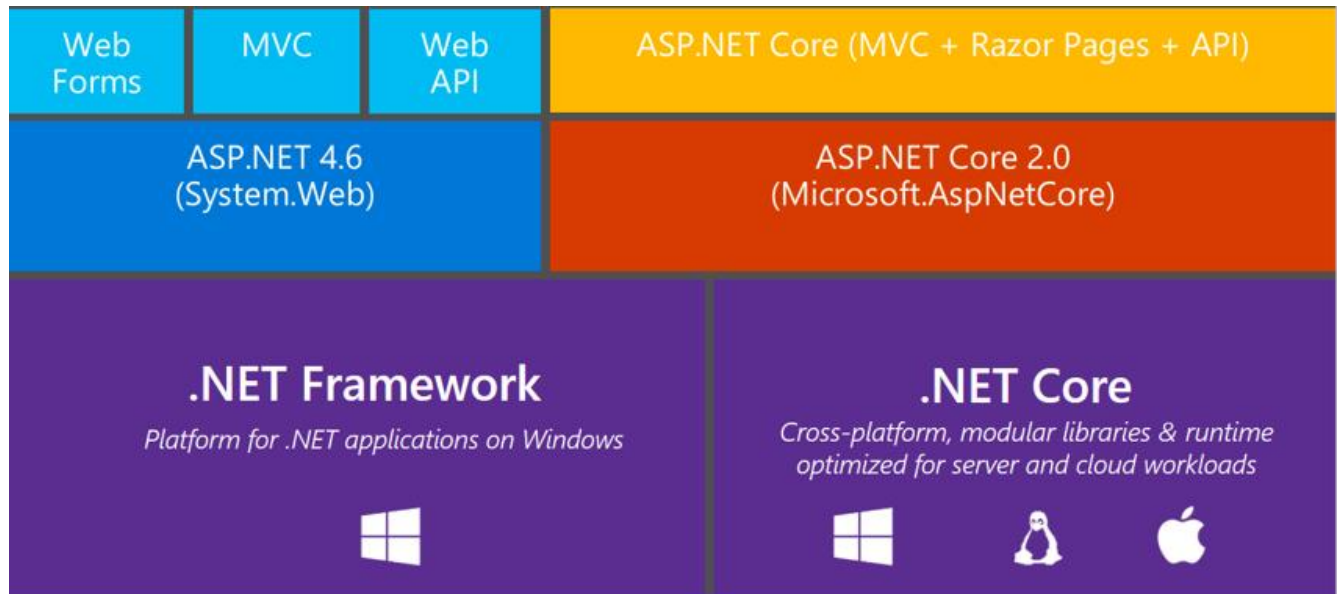
Web API trong asp.net core

Các khái niệm cơ bản gắn liền với API

API là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng.



Restful api là gì ?



Restful API được hiểu là một tiêu chuẩn được sử dụng để thiết kế API cho các ứng dụng website. Nhằm để quản lý các resource web một cách dễ dàng hơn. Có thể nói Restful là một trong những kiểu thiết kế API được các lập trình viên sử dụng khá phổ biến.

Theo đó Rest quy định cách sử dụng http method như GET, POST, PUT, DELETE... và cách định dạng URL cho các ứng dụng web.

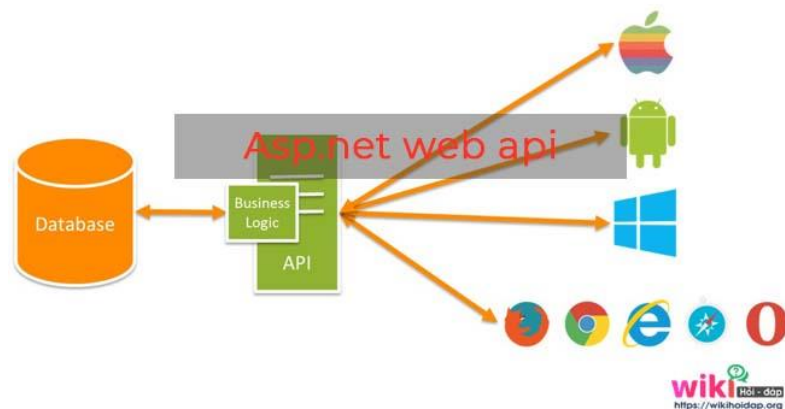
API cũng có thể liên quan đến khung phần mềm : Khung có thể dựa trên một số thư viện triển khai một số API. Nhưng không giống như việc sử dụng API thông thường, quyền truy cập vào hành vi được xây dựng trong khung trung gian bằng cách mở rộng nội dung của nó với các lớp mới cắm vào khung chính nó.

Web API là một trong những công nghệ mới của Microsoft dùng để xây dựng dịch vụ thành phần phân tán. Web API là mô hình dùng để hỗ trợ MVC bao gồm: routing, controller, action result, filter, loc container, model binder, unit test, injection. Bên cạnh đó nó còn hỗ trợ restful đầy đủ các phương thức: Get/Post/put/delete dữ liệu.

Ưu điểm của API

- Kết nối mọi lúc nhờ vào Internet.
- Giao tiếp hai chiều phải được xác nhận trong các giao dịch.

- Vì giao tiếp là API hai chiều nên thông tin rất đáng tin cậy.
- Cung cấp trải nghiệm thân thiện với người.
- Cung cấp giải pháp phát triển khi các nhà phát triển tìm thấy cách sử dụng mới để trao đổi API.
- Cấu hình đơn giản khi được so sánh với WCF.
- Mã nguồn mở.
- Hỗ trợ chức năng RESTful một cách đầy đủ.
- Hỗ trợ đầy đủ các thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, dependency injection, unit test.
- Khả năng trình diễn cao.
- Đối với các website thông thường, chúng ta hoàn toàn có thể tạo ra mà không cần đến 1 API nào cả nhưng nhu cầu hiện nay thì tất cả mọi dịch vụ được tạo ra người dùng không chỉ muốn truy cập trên website mà còn cả nhiều trên nhiều thiết bị của nhiều hệ điều hành khác nhau, để giải quyết các vấn đề này API ngày nhu cầu sử dụng API ngày càng trở nên phổ biến.



2.6. ChatBot

Trước tiên, ta phải tìm hiểu về “Bot”. Một “Bot” là một phần mềm thực hiện các nhiệm vụ, công việc con người yêu cầu một cách tự động. Hoặc “Bot” cũng có thể là một chương trình máy tính được thiết kế để “giao tiếp” với người dùng thông qua kết nối Internet.

“Chat” có lẽ là từ quá quen thuộc mà ai cũng biết, nghĩa là trò chuyện, giao tiếp qua lại giữa hai người. Chatbot chính là hệ thống các Bot ở trong trạng thái trực tuyến, trên các website hoặc trên các nền tảng, giao diện chat khác của social media để “chat tự động” với người dùng.

Chatbot được hiểu thông thường như một “cái máy” có thể đối thoại một cách tự nhiên với con người. Ví dụ: bất kỳ người dùng nào cũng có thể hỏi chatbot một câu hỏi hoặc một câu lệnh bất kỳ và chatbot sẽ trả lời hoặc thực hiện một hoạt động phù hợp đáp lại người dùng.

Chatbot tương tác với chúng ta như một hệ thống trả lời tin nhắn nhanh chóng, tự động. Bằng cách xây dựng, giả lập các mô hình tương tác, kích bản tương tác như của con người sử dụng phương pháp trong Machine Learning, hệ thống Chatbot có thể “tự học”, “tự hiểu” các câu hỏi, nhu cầu của người dùng, khách hàng và thực hiện, đưa ra các phản hồi sao cho phù hợp.

Chatbot sau khi được lập trình và huấn luyện nó sẽ tự động làm việc một cách độc lập như một con người. Chỉ những câu hỏi, tin nhắn của khách hàng đã được cấu trúc lại thành các câu, ý định ngắn gọn với ngôn ngữ tự nhiên (natural language) và thêm vào hệ thống kèm theo các kịch bản đối thoại tương ứng đã xây dựng trước đó thì Chatbot mới có khả năng đưa ra phản hồi.

Chatbot sẽ sử dụng database – cơ sở dữ liệu, nơi lưu trữ các câu hỏi, câu đối thoại đã được “huấn luyện” cho Chatbot – để phản hồi lại người dùng tại bất kỳ thời điểm nào. Trong trường hợp Chatbot không hiểu câu hỏi của người dùng, có thể do Chatbot chưa được “huấn luyện” kỹ thì Chatbot sẽ phản hồi sai thông tin, không phù hợp với mong muốn của người dùng. Tuy nhiên, mỗi Chatbot mặc dù làm việc độc lập nhưng vẫn được vận hành và quản lý bởi người xây dựng hệ thống. Do đó, Chatbot sẽ chuyển thông tin đến người quản lý khi không hiểu ý định của người dùng hay khách hàng.

Chatbot được “huấn luyện” và hoàn thiện trong thời gian dài sẽ tăng khả năng “tự học”, tự phát triển về phạm vi hiểu biết các ý định của người dùng và đạt được độ chính xác, độ tin cậy cao trong các phản hồi đưa ra.

Chatbot thường trao đổi với người dùng qua hình thức tin nhắn (Textual) hoặc âm thanh (Auditory).

CHƯƠNG 3: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN

3.0. THƯ VIỆN SỬ DỤNG

ASP.NET CORE WEB API LIB

```
<ItemGroup>
  <PackageReference Include="AutoMapper" Version="10.1.1" />
  <PackageReference Include="AutoMapper.Extensions.Microsoft.DependencyInjection" Version="8.1.1" />
  <PackageReference Include="FluentValidation.AspNetCore" Version="10.1.0" />
  <PackageReference Include="IdentityServer4.AccessTokenValidation" Version="3.0.1" />
  <PackageReference Include="Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore" Version="5.0.4" />
  <PackageReference Include="Microsoft.AspNetCore.Identity.EntityFrameworkCore" Version="5.0.4" />
  <PackageReference Include="Microsoft.AspNetCore.Mvc.Abstractions" Version="2.2.0" />
  <PackageReference Include="Microsoft.AspNetCore.Mvc.NewtonsoftJson" Version="5.0.4" />
  <PackageReference Include="Microsoft.EntityFrameworkCore.Design" Version="5.0.6">
    <PrivateAssets>all</PrivateAssets>
    <IncludeAssets>runtime; build; native; contentfiles; analyzers; buildtransitive</IncludeAssets>
  </PackageReference>
  <PackageReference Include="Microsoft.EntityFrameworkCore.SqlServer" Version="5.0.4" />
  <PackageReference Include="Microsoft.EntityFrameworkCore.Tools" Version="5.0.6">
    <PrivateAssets>all</PrivateAssets>
    <IncludeAssets>runtime; build; native; contentfiles; analyzers; buildtransitive</IncludeAssets>
  </PackageReference>
  <PackageReference Include="Swashbuckle.AspNetCore" Version="5.6.3" />
  <PackageReference Include="System.Configuration.ConfigurationManager" Version="5.0.0" />
</ItemGroup>
```

FLUTTER LIB

```
dependencies:
  sound_stream: ^0.2.0
  dialogflow_grpc: ^0.2.9

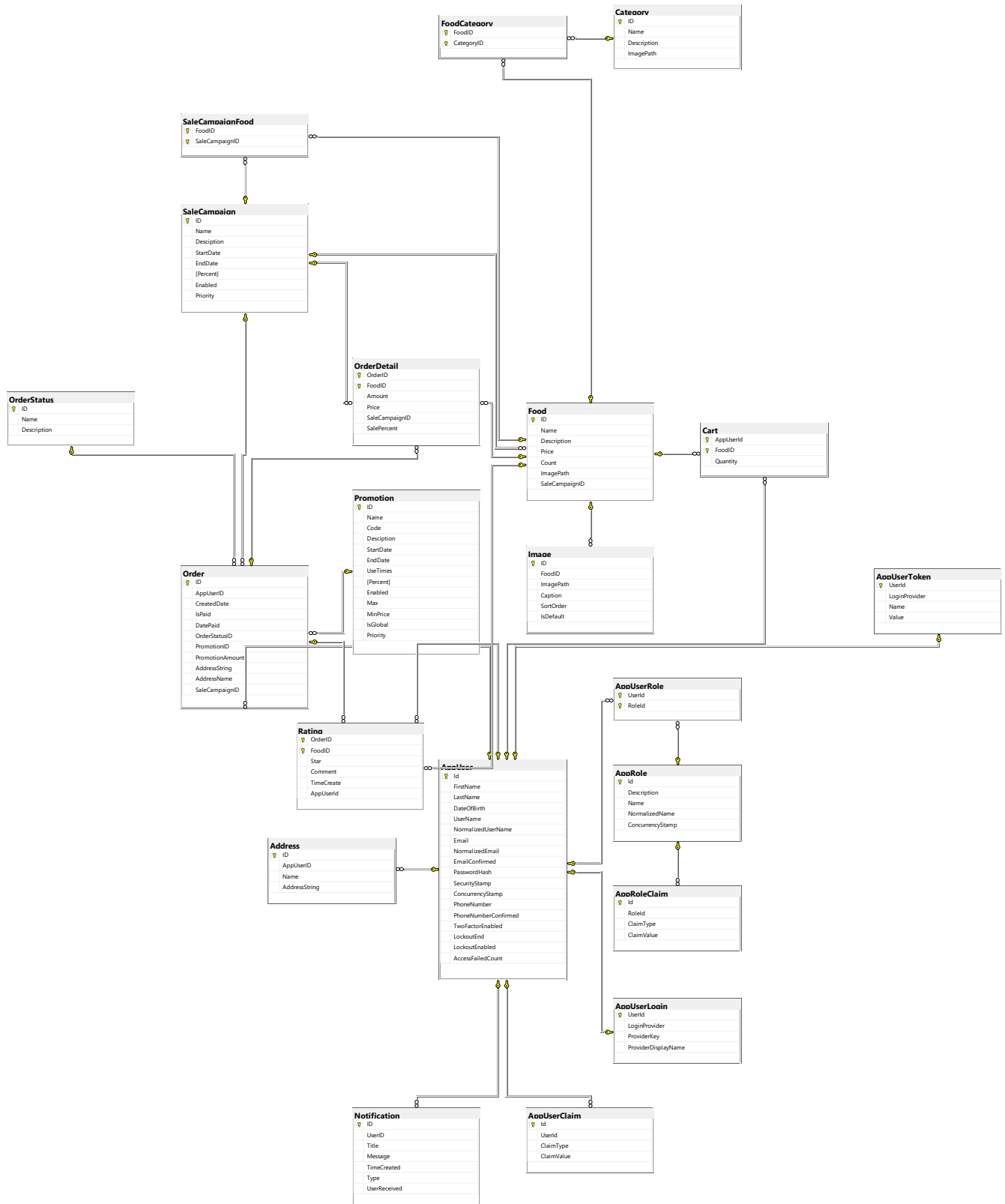
  flutter:
    sdk: flutter

  #carousel_pro: ^0.0.13
  http: ^0.13.3

  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^0.1.3
  json_annotation: ^4.0.1
  path_provider: ^2.0.1
  provider: ^5.0.0
  cached_network_image: ^3.0.0
  intl: ^0.17.0
  #google_fonts: ^2.1.0
  carousel_slider: ^4.0.0-nullsafety.0
  flutter_bloc: ^7.0.0
  equatable: ^2.0.2
  rxdart: ^0.27.0
```

3.1. THIẾT KẾ CƠ SỞ DỮ LIỆU

Đối với một hệ thống bán hàng nói chung và ứng dụng bán hàng online nói riêng thì cơ sở dữ liệu là phần hết sức quan trọng, đóng vai trò mấu chốt trong sự thành công của toàn ứng dụng. Cơ sở dữ liệu được xây dựng như sau:



Hình 3.1.Sơ đồ CSDL

3.2. Mô hình hoá yêu cầu

3.2.1. Xác định actor:

a) Khách hàng

- Là người có nhu cầu đặt món ăn trên App mobile

b) Nhân viên quản lý

- Là người trực tiếp quản lý các món ăn, đơn hàng trong website

3.2.2. Xác định Use Case

c) Yêu cầu chức năng nghiệp vụ - Use Case:

1. Khách hàng:

STT	Công việc	Loại công việc	Mô tả chức năng, quy định và công thức liên quan	Ghi chú
1	Đăng ký tài khoản	Lưu trữ	Mỗi khách hàng có 1 username duy nhất. Khi đăng ký, hệ thống yêu cầu Khách hàng điền đầy đủ các thông tin: - Họ tên - Username - Password - SĐT - Địa chỉ	
2	Đăng nhập	Lưu trữ	- Thông tin tài khoản phải tồn tại trên hệ thống	
3	Xem danh sách món ăn	Tra cứu	- Khách hàng được phép xem thông tin (số lượng/giá bán/mô tả) các món ăn đang tồn tại trong hệ thống	
4	Thêm món ăn vào giỏ hàng	Lưu trữ	- Khách hàng có thể đưa món ăn vào giỏ hàng chờ trước khi thanh toán	
5	Quản lý giỏ hàng	Lưu trữ	- Thêm/Giảm số lượng món ăn có trong giỏ hàng - Xóa món ăn khỏi giỏ hàng	
6	Thanh toán	Lưu trữ	- Sau khi chắc chắn về những món ăn mà mình muốn đặt, khách hàng nhấn nút Thanh	

			toán ngay trong Giỏ hàng để hệ thống kết xuất hoá đơn.	
7	Quản lý địa chỉ	Lưu trữ	Thêm xoá sửa địa chỉ giao hàng	
8	Quản lý Profile	Lưu trữ	Quản lý thông tin cá nhân và mật khẩu	

2. Nhân viên quản lý (Admin, Nhân viên bán hàng)

- Nhân viên quản lý có thể thực hiện các use case của khách hàng và thực hiện các use case khác:

STT	Công việc	Loại công việc	Quy định/công thức liên quan	Ghi chú
1	Quản lý món ăn	Lưu trữ	Có thể thực hiện thêm/cập nhật thông tin/xoá món ăn trong hệ thống	
2	Xem danh sách hoá đơn	Tra cứu	Có thể xem toàn bộ hoá đơn trong hệ thống	
3	Xem thống kê	Tra cứu	Xem số lượng món ăn còn trong hệ thống	
4	Quản lý các thông tin phụ	Lưu trữ	Địa chỉ, Khuyến mãi, thông báo, Thông tin đăng nhập nhân viên (Admin)	

d) Yêu cầu chức năng hệ thống:

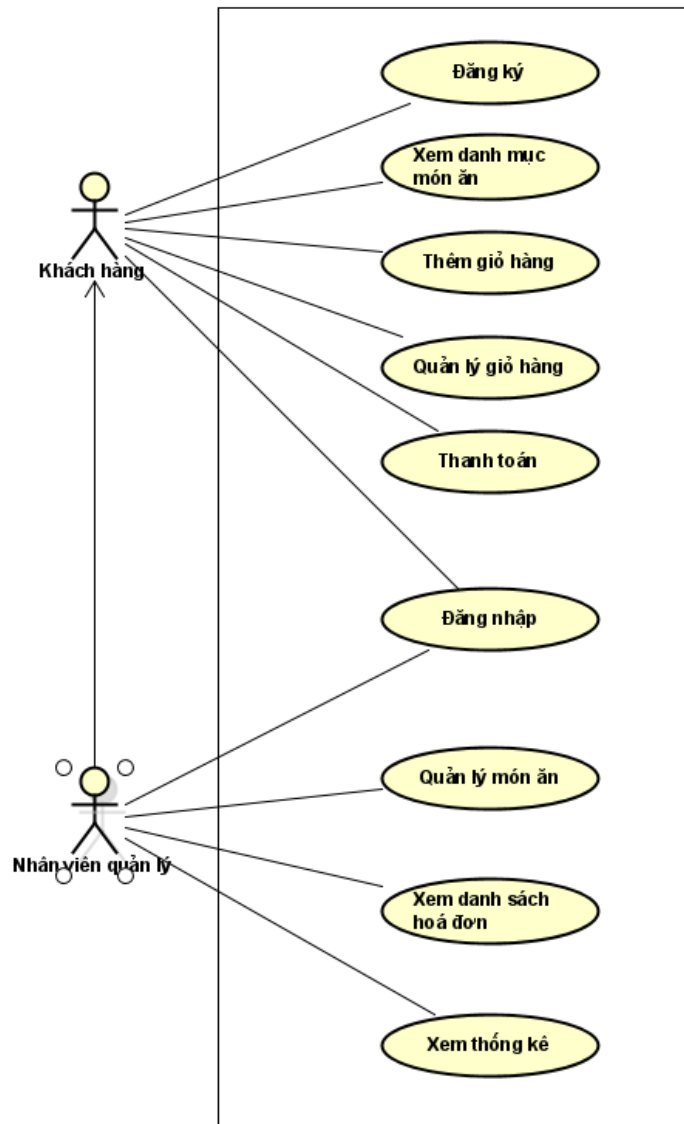
STT	Nội dung	Mô tả chi tiết	Ghi chú
1	Sao lưu, back up, phục hồi thông tin	Các thông tin trong hệ thống không thật sự mất đi khi xoá, chỉ chuyển đổi trạng thái từ còn khả dụng (status=1) thành không khả dụng (status=0)	
2	Phân quyền sử dụng	Khách hàng: Chỉ được truy cập tài khoản vào các app xem món ăn, giỏ hàng.	

		Nhân viên quản lý: Có thể truy cập được tất cả các trang trong hệ thống.	
--	--	--	--

e) Yêu cầu phi chức năng:

STT	Nội dung	Tiêu chuẩn	Mô tả chi tiết
1	Tiết kiệm được thời gian, thu hẹp không gian lưu trữ, tránh thất lạc dữ liệu.	Hiệu quả	Tiết kiệm thời gian so với quản lý thủ công. Dữ liệu được sao lưu trên máy, có thể dễ dàng phục hồi
2	Giao diện đơn giản, dễ thao tác	Tiện dụng	

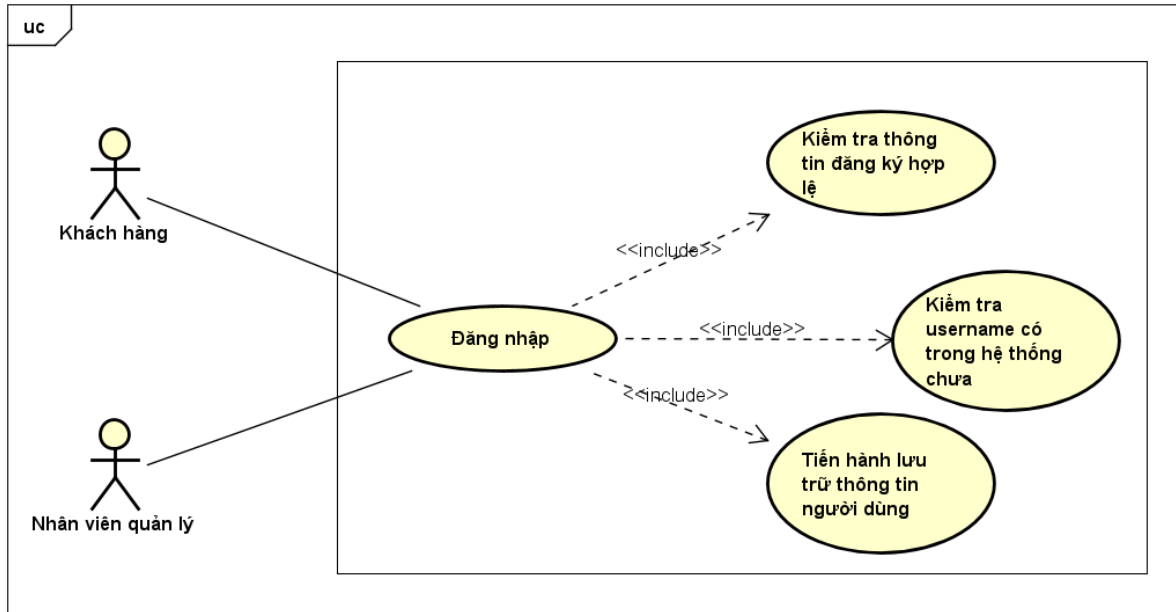
1.1.2. Use Case Tổng quát:



- Ghi chú: Quản lý có toàn quyền sử dụng các chức năng của Nhân viên bán hàng.

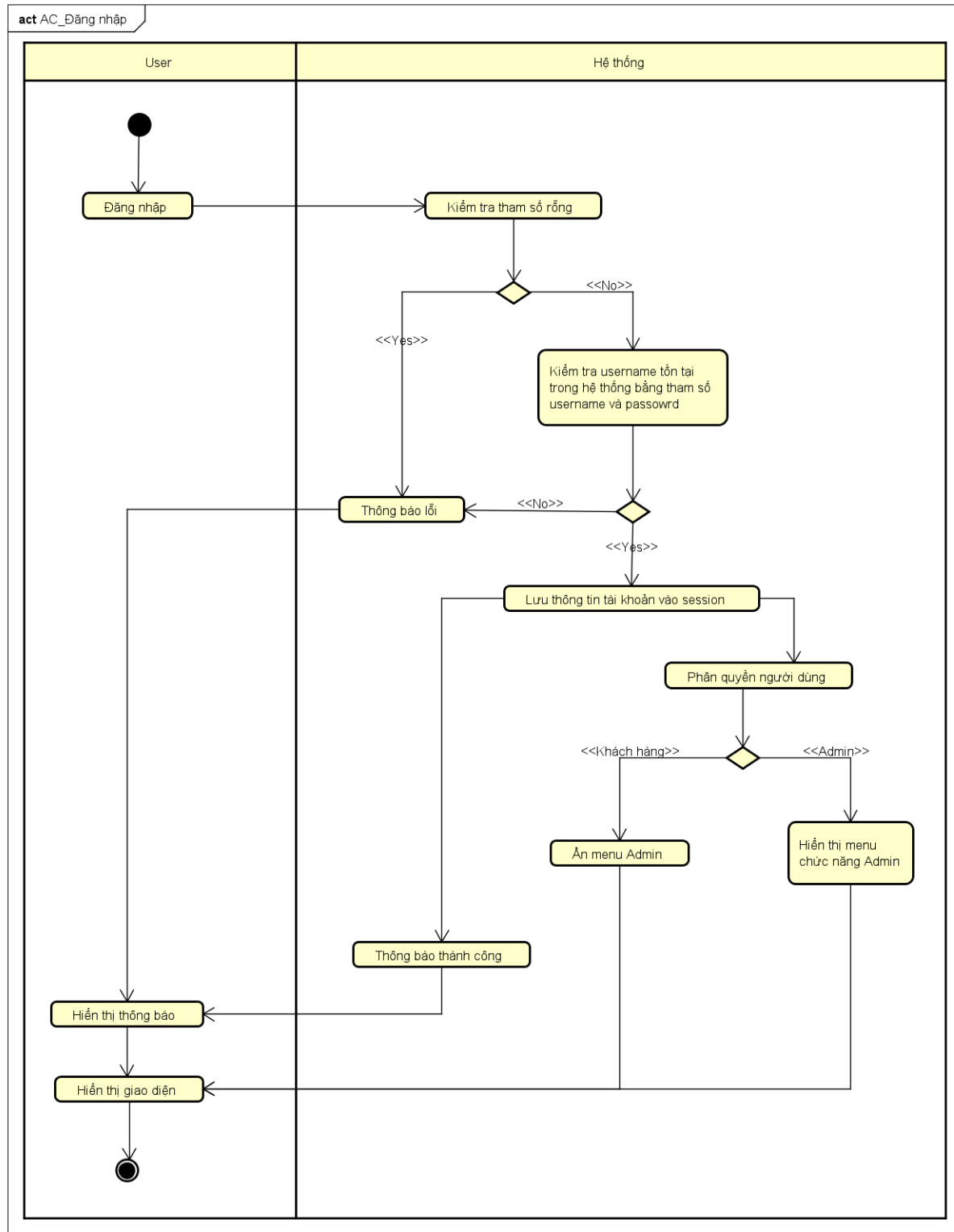
3.2.3. Phân tích – Thiết kế chức năng

Đăng nhập



Mô tả UC:

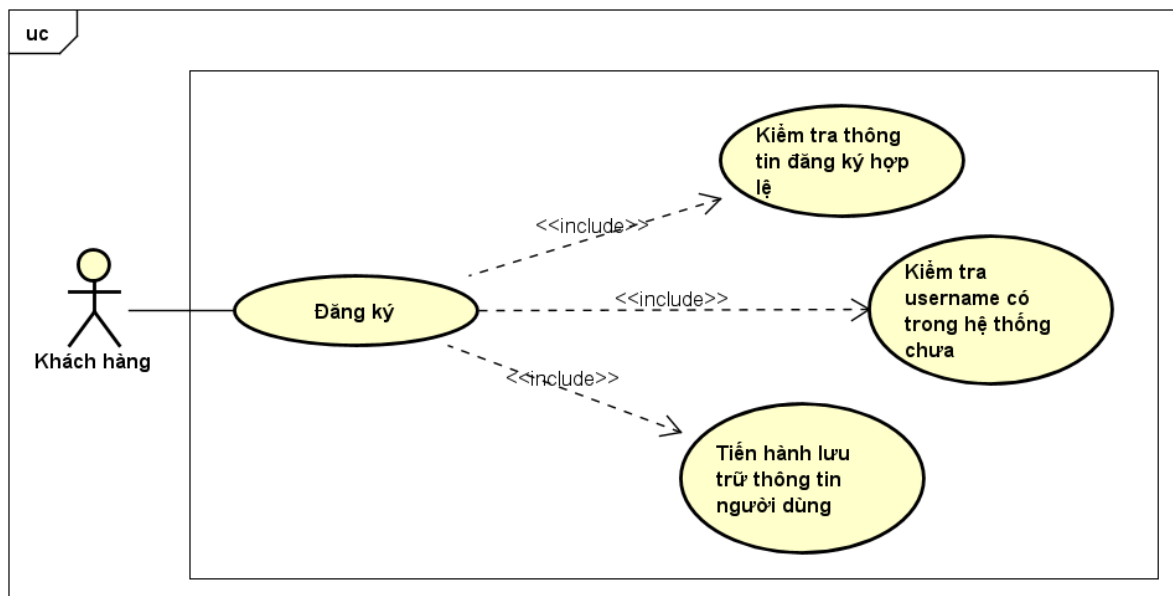
Trước khi sử dụng App/Website, Nhân viên quản lý và Khách hàng đều phải đăng nhập vào hệ thống.



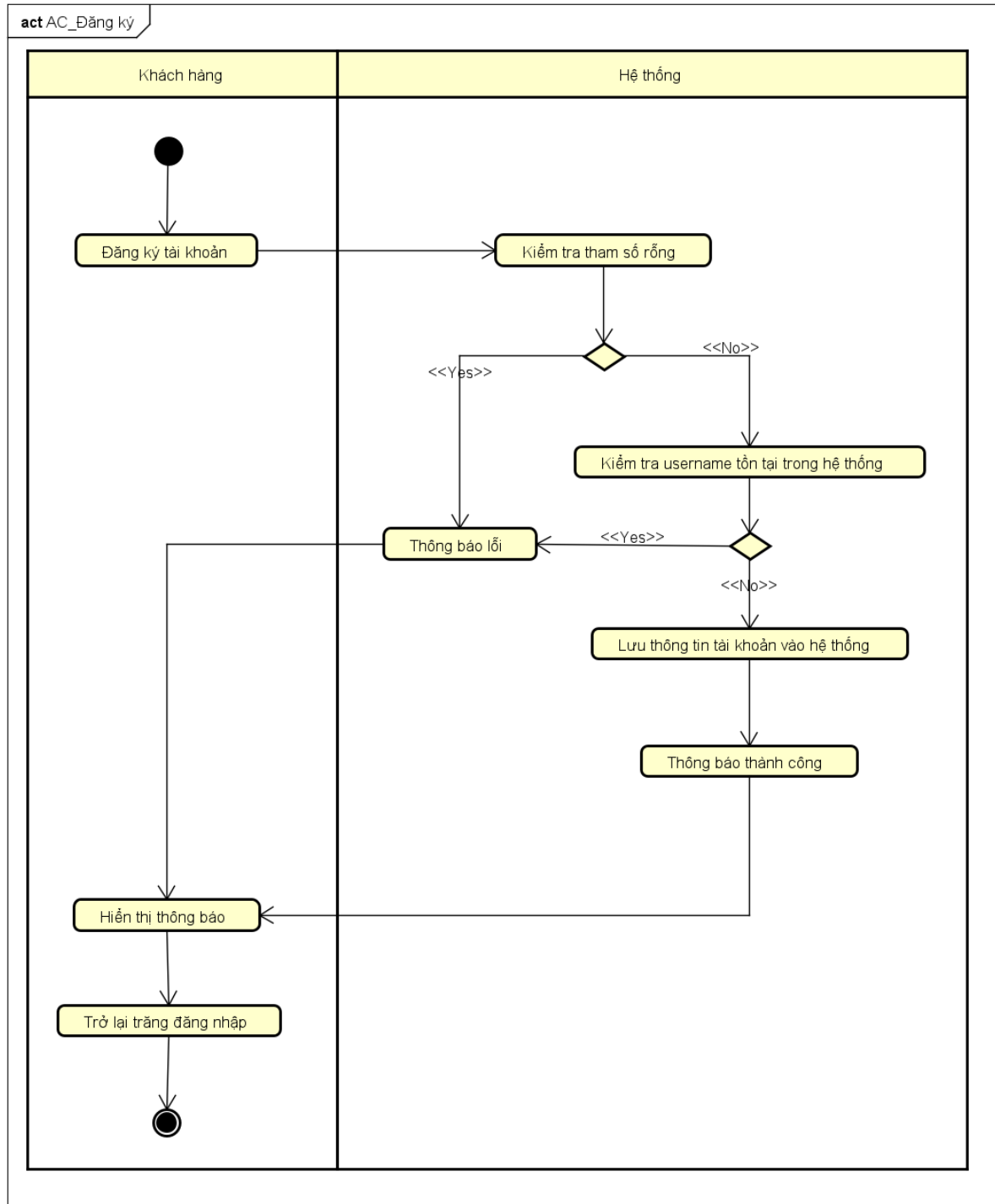
Basic flow	1. Người dùng nhập username/password và nhấn Đăng nhập 2. Hệ thống kiểm tra tham số đầu vào
-------------------	--

	<ul style="list-style-type: none">3. Hệ thống kiểm tra username/password có tồn tại trong database4. Hệ thống lưu thông tin người dùng vào session storage5. Hệ thống kiểm tra phân quyền người dùng Nếu là nhân viên6. Nếu là khách hàng, hệ thống đến page khách hàng/ App6.1. Nếu là Nhân viên quản lý, hệ thống sẽ hiển thị trang chủ có thanh điều hướng vào page admin
--	--

Đăng ký



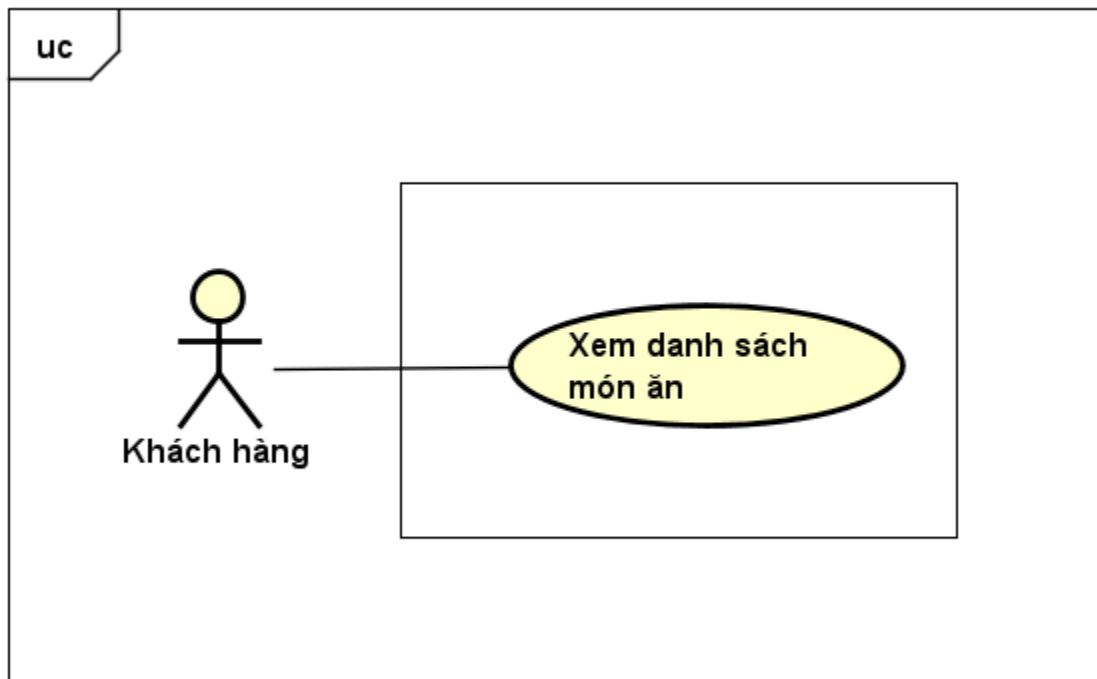
Khi người dùng đăng ký trên website/ app mobile, tài khoản chỉ được phân quyền Khách hàng.



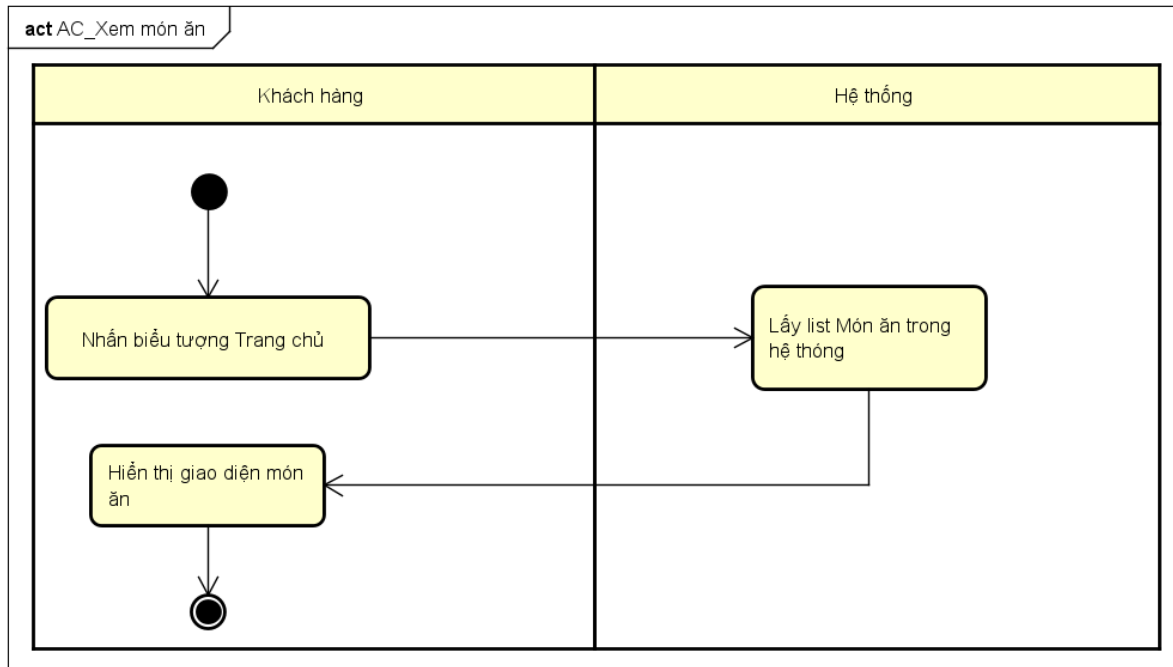
Basic flow	<ol style="list-style-type: none"> 1. Người dùng nhập họ tên, username, password, repassword, sdt, địa chỉ và nhấn Đăng ký 2. Hệ thống kiểm tra tham số đầu vào, password và repassword
-------------------	---

	<p>3. Hệ thống tiến hành kiểm tra username có tồn tại trong hệ thống chưa</p> <p>5. Hệ thống thông báo thành công</p> <p>6. Người dùng nhấn OK</p> <p>7. Hệ thống mở trang đăng nhập</p>
--	--

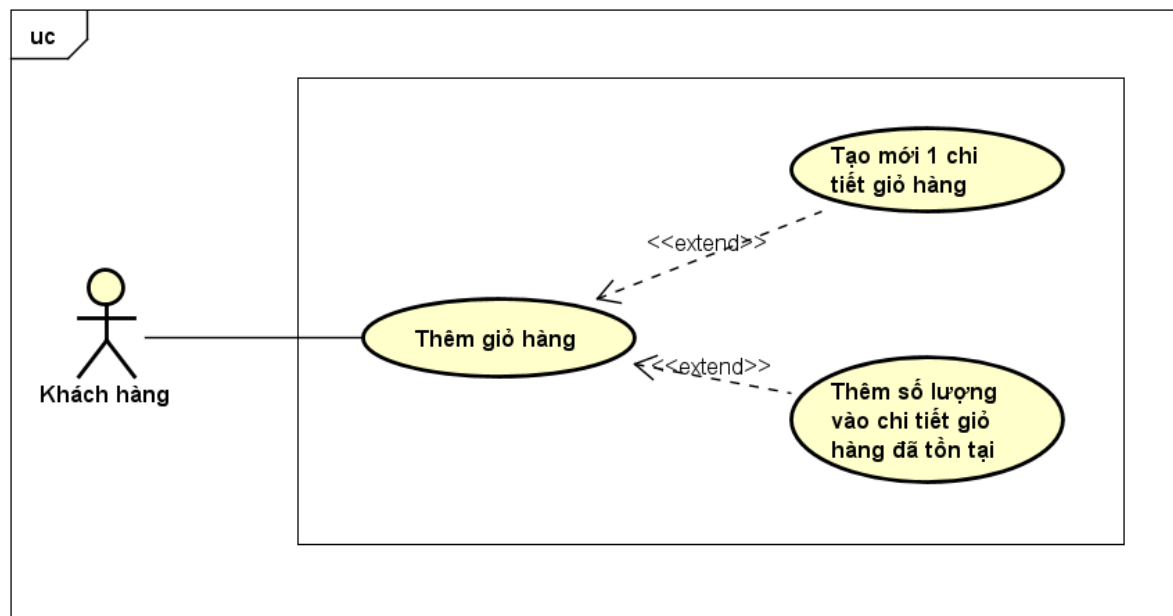
Xem danh sách món ăn



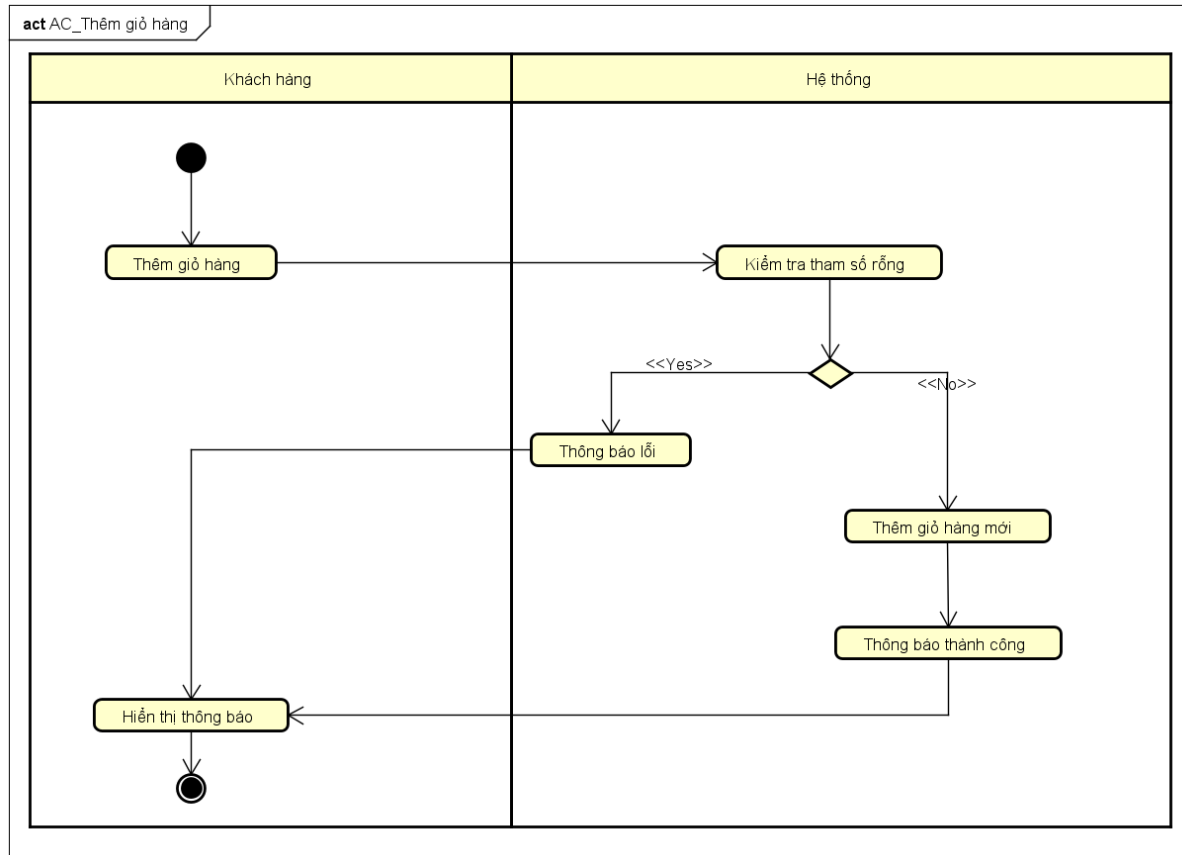
- Khách hàng và Nhân viên quản lý có thể xem danh sách món ăn ngay tại trang chủ.



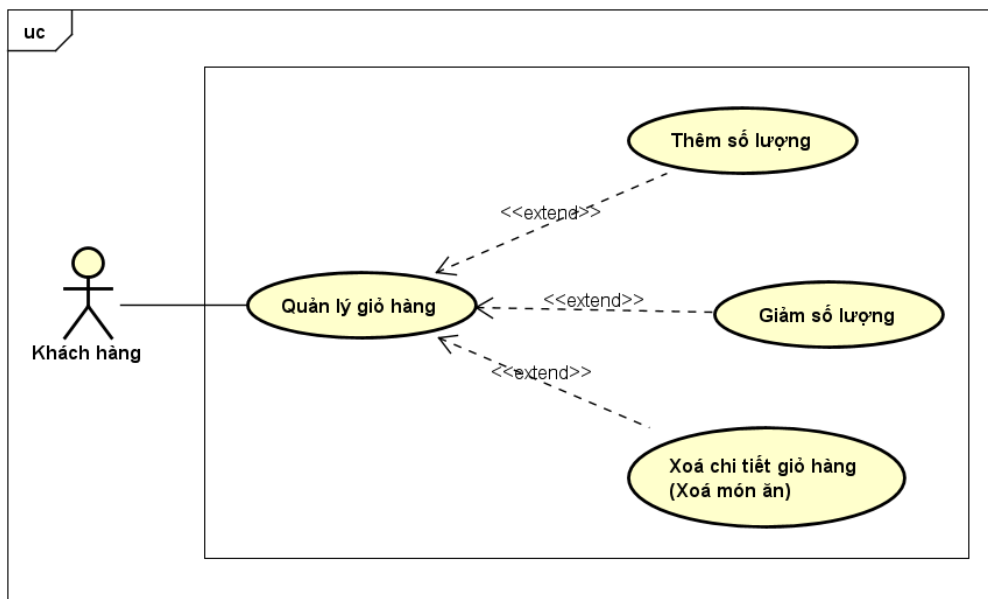
Thêm giỏ hàng



- Khi thực hiện chức năng Thêm giỏ hàng, hệ thống sẽ tạo 1 chi tiết giỏ hàng nếu món ăn chưa có trong giỏ hàng, hoặc sẽ thêm số lượng nếu món ăn đã có trong giỏ hàng

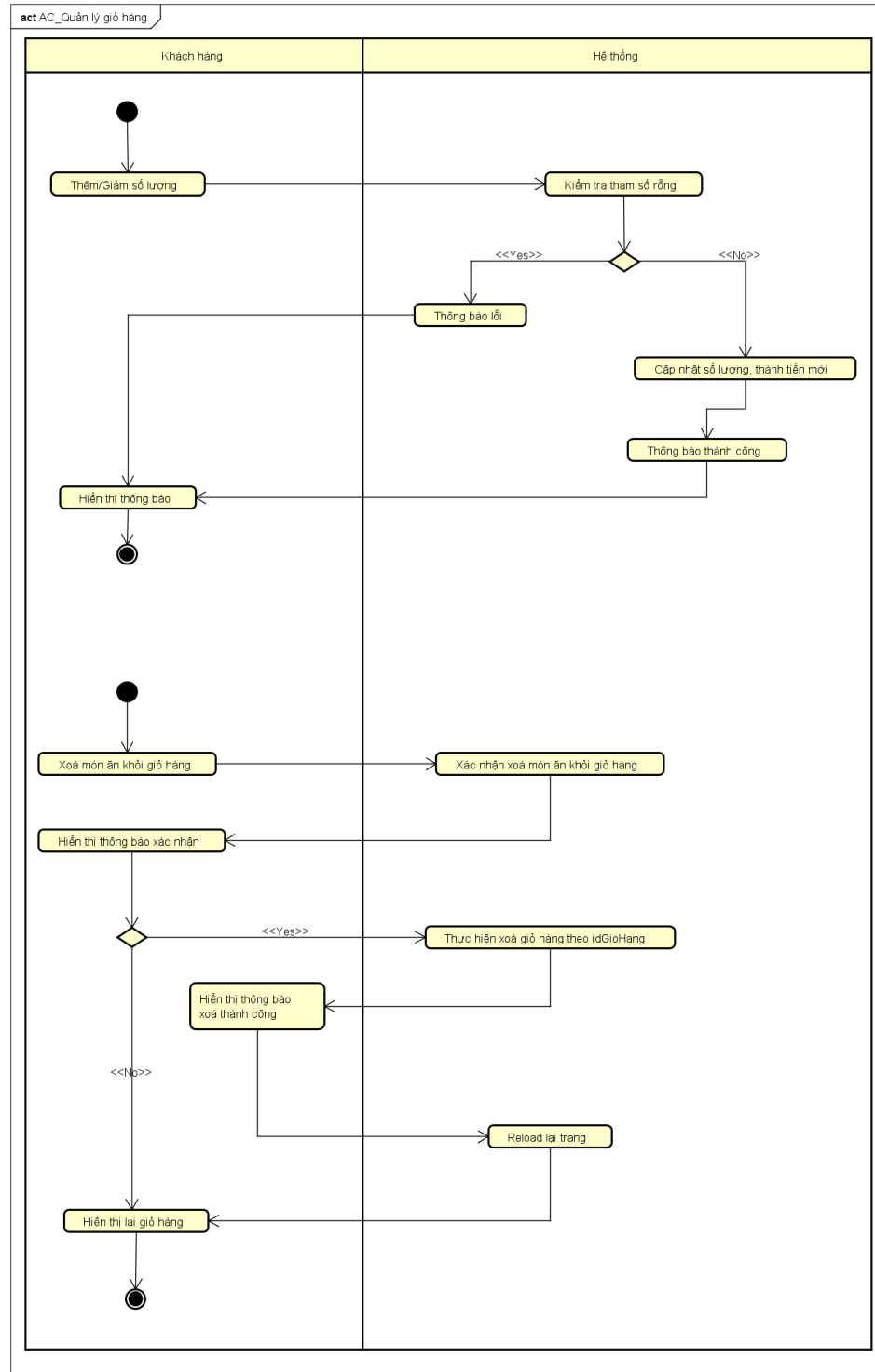


Quản lý giỏ hàng

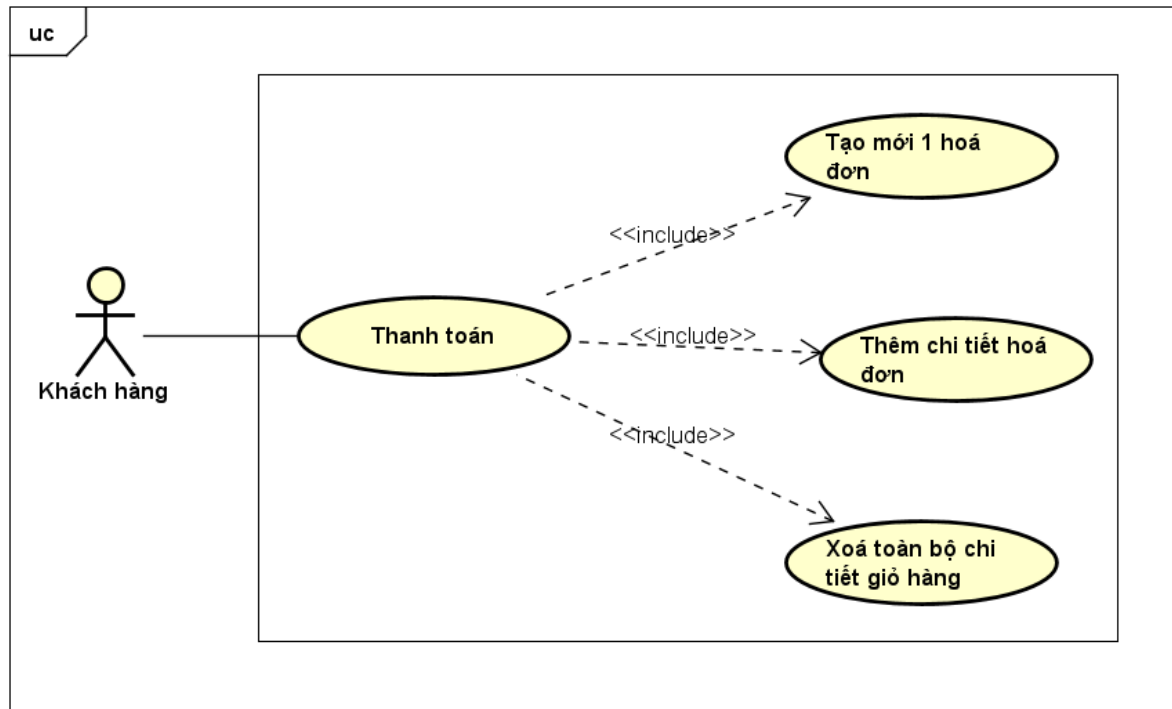


Use Case quản lý giỏ hàng có 3 chức năng chính gồm:

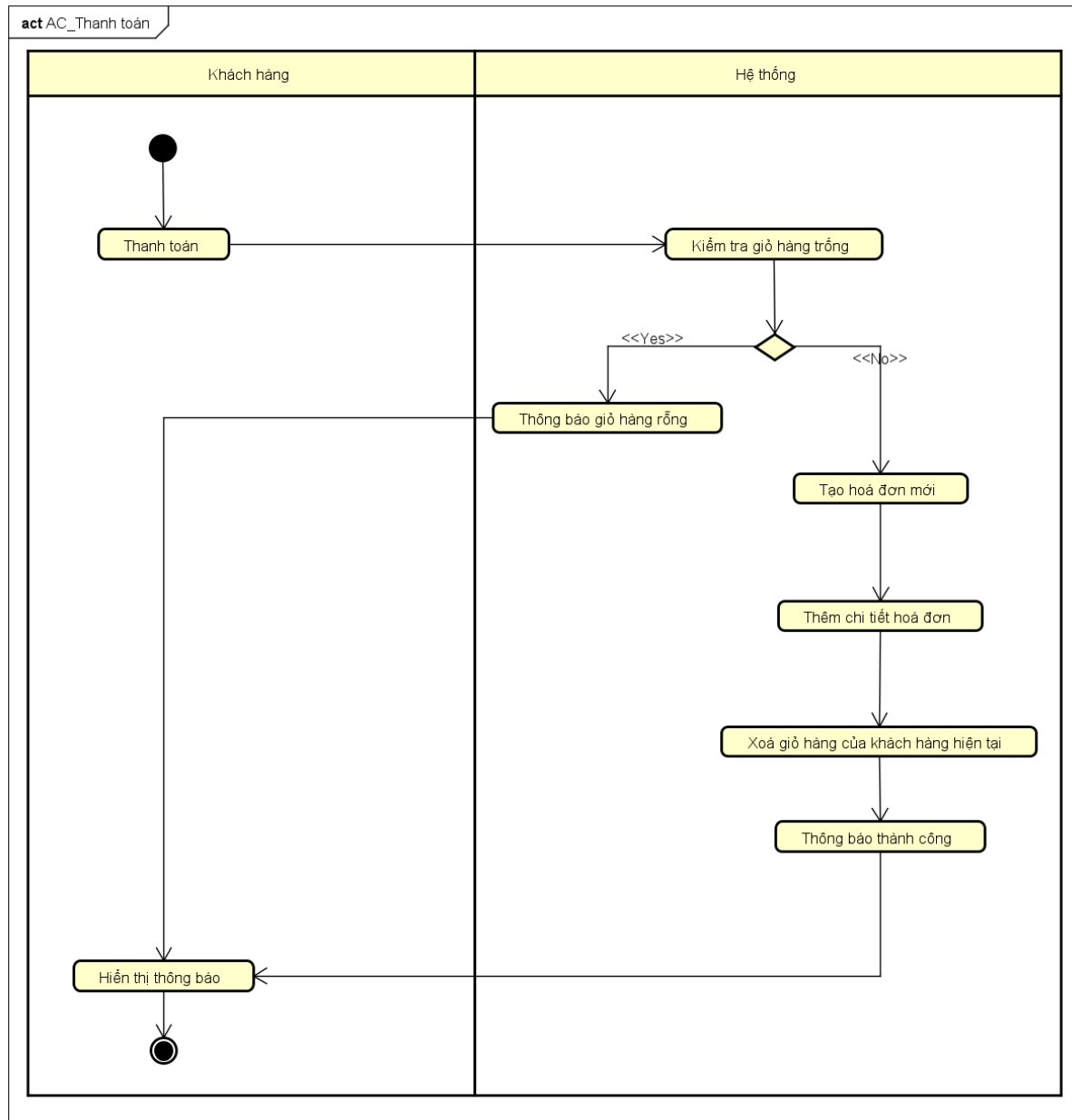
- Thêm số lượng món ăn trong giỏ hàng
- Giảm số lượng món ăn trong giỏ hàng
- Xoá món ăn khỏi giỏ hàng



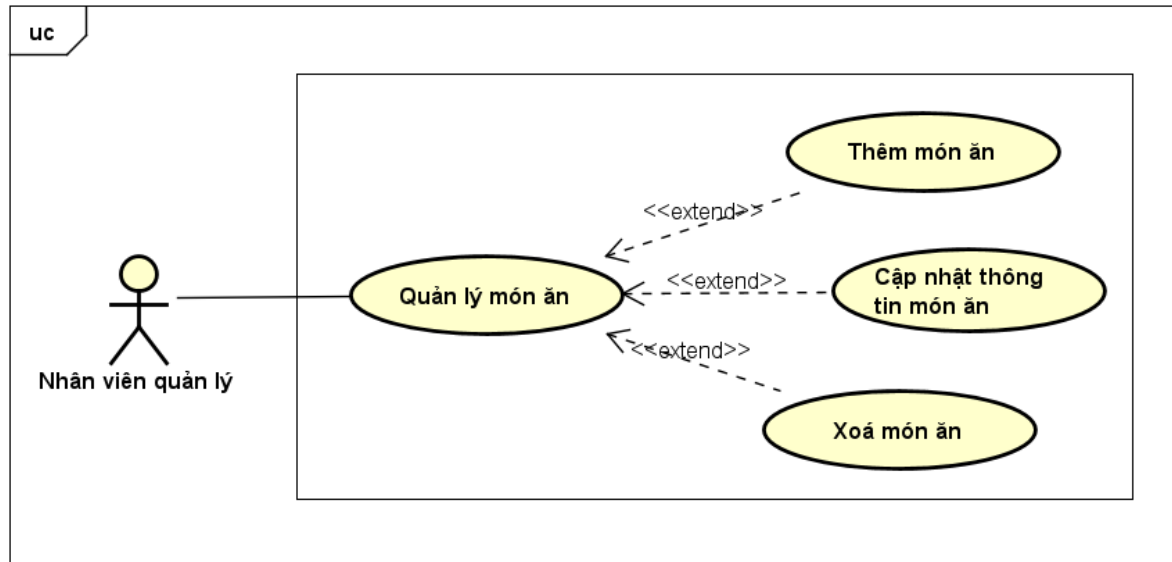
Thanh Toán



- Khi bấm Thanh toán, nội dung giỏ hàng sẽ được chuyển thành hoá đơn và toàn bộ chi tiết giỏ hàng sẽ được làm trống.

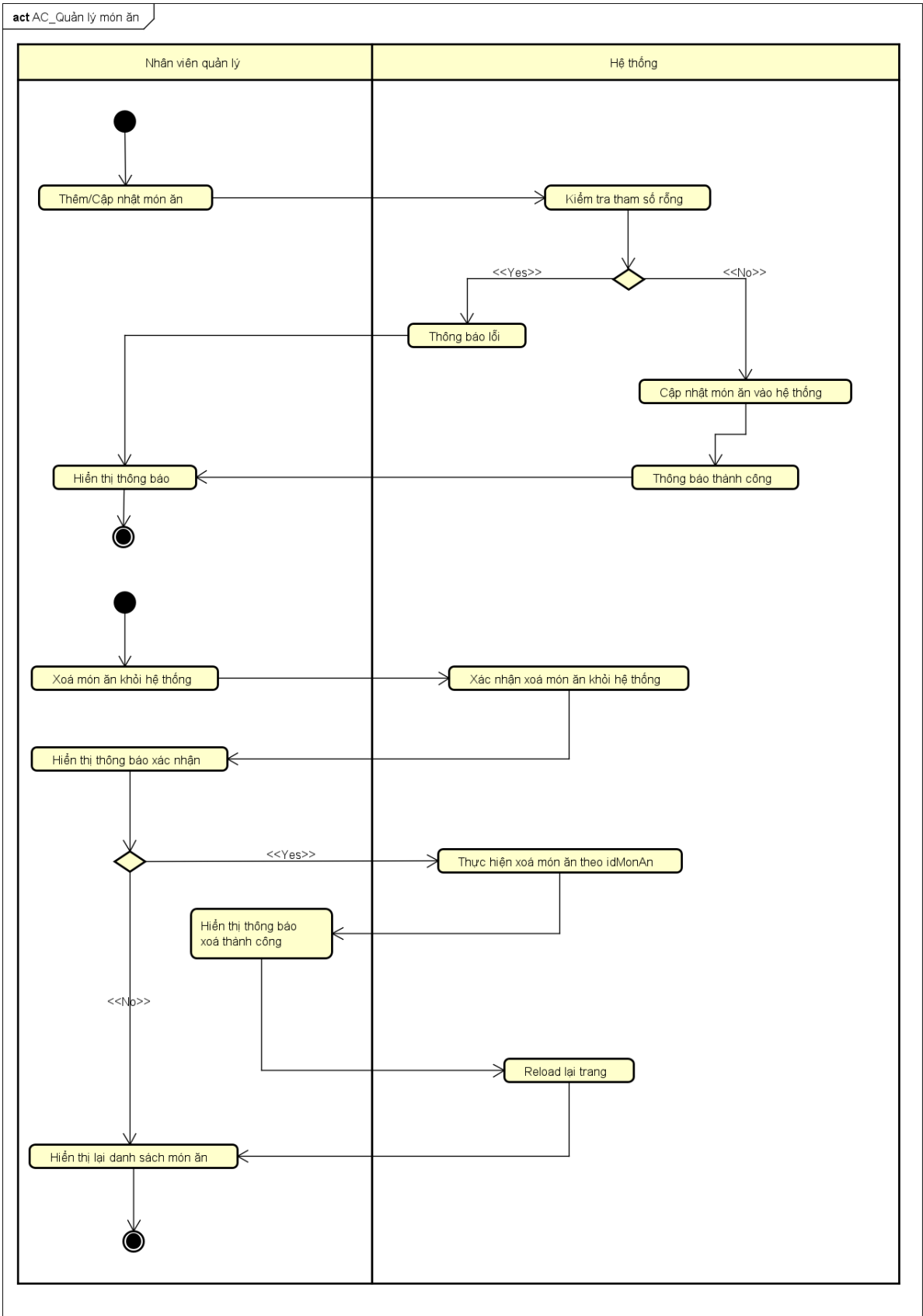


Quản lý món ăn

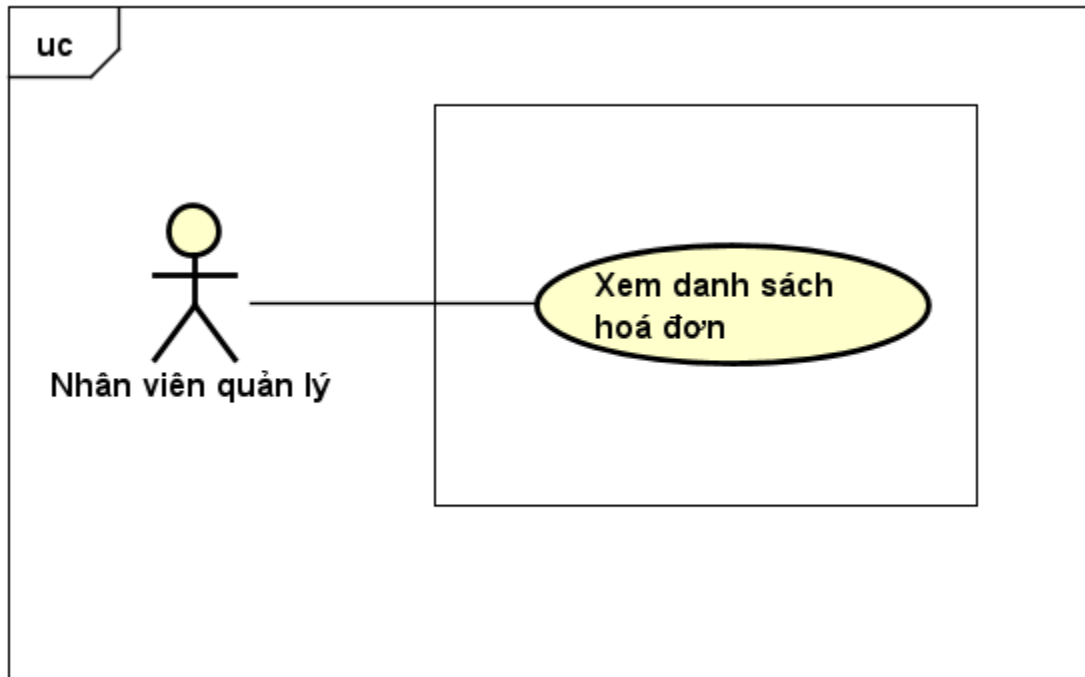


Tương tự use case quản lý giỏ hàng, Use Case quản lý món ăn có 3 chức năng chính gồm:

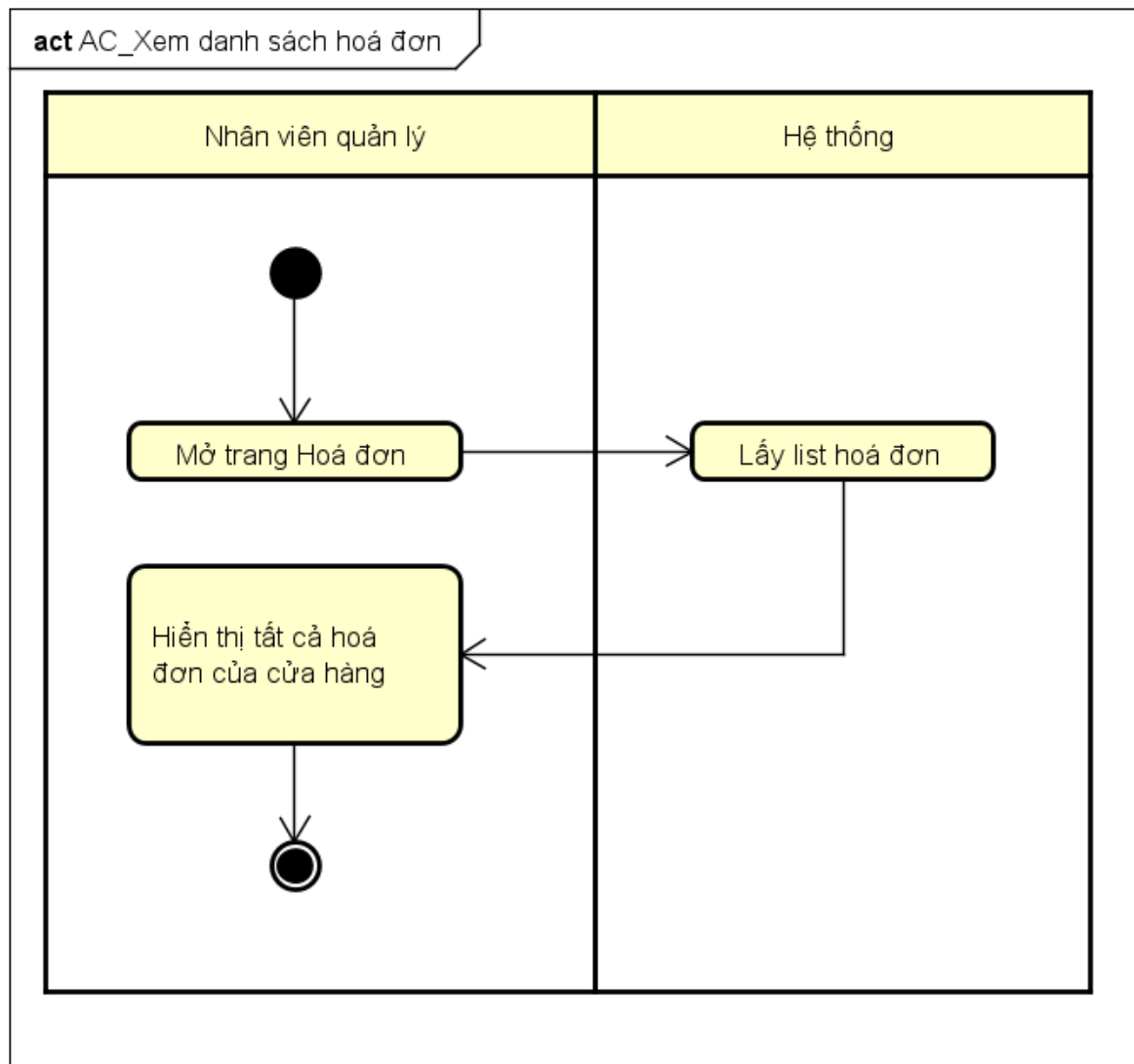
- Thêm món ăn mới vào hệ thống
- Cập nhật thông tin/mô tả/hình ảnh/sô lượng/giá bán của món ăn
- Xoá món ăn khỏi hệ thống



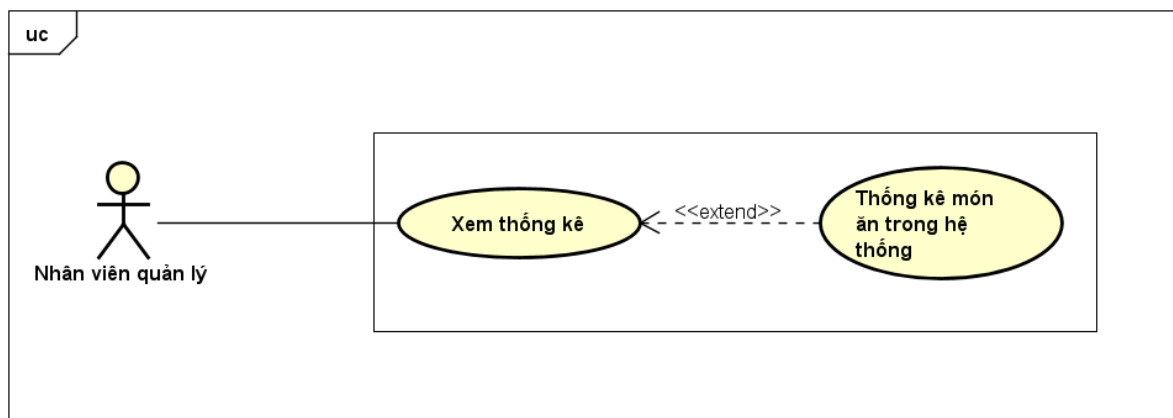
Xem danh sách hoá đơn



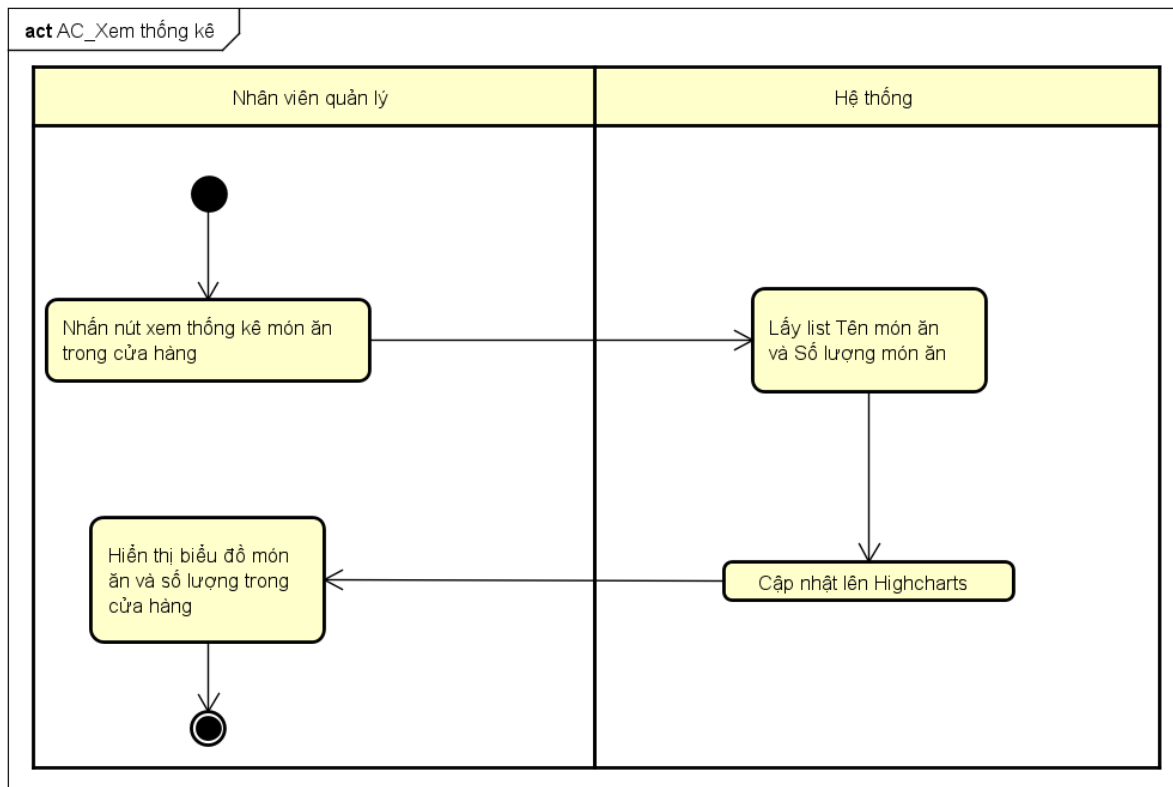
- Nhân viên quản lý có thể xem toàn bộ hoá đơn trong hệ thống.



Xem Thống Kê



- Nhân viên quản lý có thể xem thống kê món ăn tồn kho trong hệ thống.



3.2.4. Cài đặt demo và giải thích một số chức năng chính

a. Cài đặt

➤ Đối với web

- + Website đã được deploy lên https (online) sử dụng database online

➤ Đối với mobile app

- + Điện thoại di động android 28,29,39
- + Ram tối thiểu 1GB
- + Bộ nhớ trống khoảng 100mb
- + Khi cài đặt chỉ cần cấp quyền audio để sử dụng chat bot

b. Demo giải thích các chức năng chính

➤ API

GET	/api/Ratings/details	🔒
SaleCampaigns ▾		
GET	/api/SaleCampaigns/{id}	🔒
DELETE	/api/SaleCampaigns/{id}	🔒
POST	/api/SaleCampaigns	🔒
PUT	/api/SaleCampaigns	🔒
GET	/api/SaleCampaigns	🔒
GET	/api/SaleCampaigns/valid	🔒
Users ▾		
POST	/api/Users/Login	🔒
POST	/api/Users/Register	🔒
GET	/api/Users	🔒
POST	/api/Users	🔒
GET	/api/Users/role/{id}	🔒

- + "Quản lý các đường dẫn API":
- + "Users": "/api/Users",
- + "Carts": "/api/Carts",
- + "Categories": "/api/Category",
- + "Foods": "/api/Food",
- + "OrderDetails": "/api/OrderDetails",
- + "Orders": "/api/Orders",
- + "OrderStatuses": "/api/OrderStatus",
- + "Promotions": "/api/Promotions",
- + "Ratings": "/api/Ratings",
- + "AppRoles": "/api/AppRoles",
- + "Images": "/api/Images",
- + "Addresses": "/api/Addresses",
- + "Notifications": "/api/Notifications",
- + "SaleCampaigns": "/api/SaleCampaigns",

+ "Home": `"/api/Home"`

Sử dụng các đường dẫn để test và fetch data

➤ Web Admin

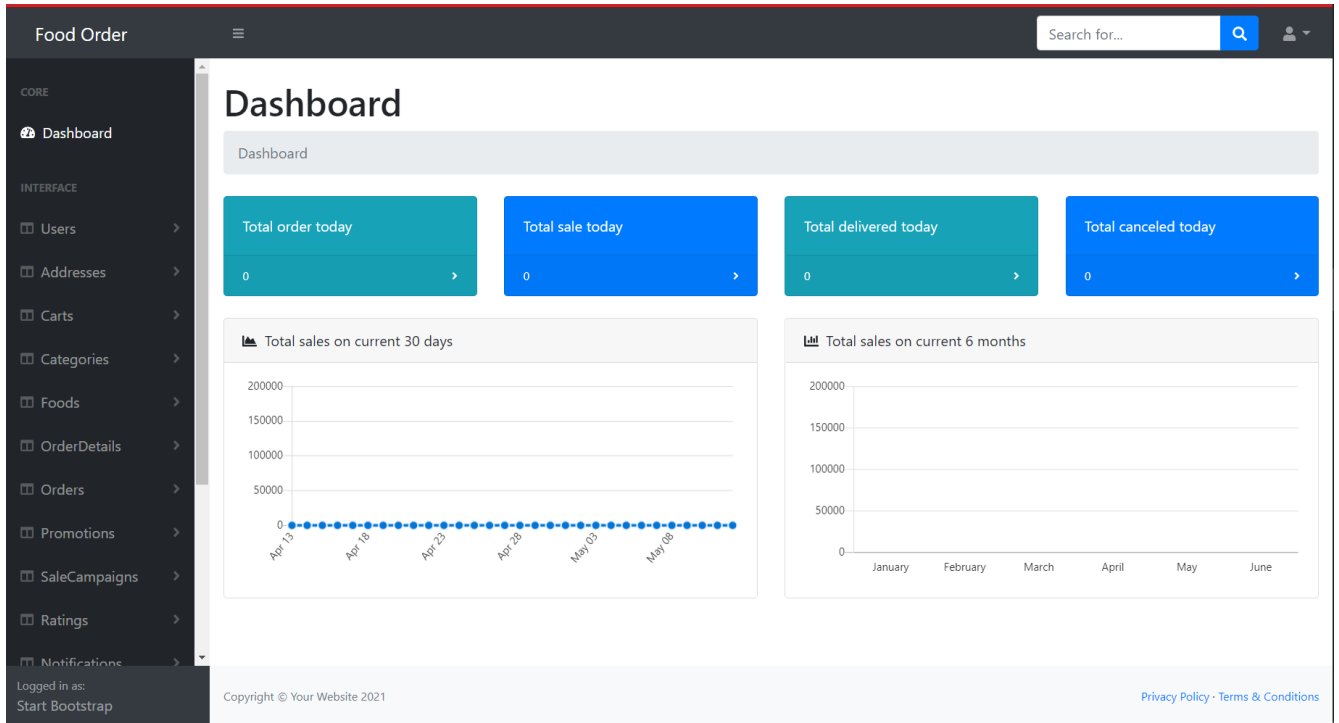
+ Đăng nhập

Copyright © Your Website 2021

Privacy Policy · Terms & Conditions

+ Trang chủ

Đề tài: Ứng dụng đặt đồ ăn với Flutter và Dart



+ Các trang quản lý thức ăn

Food Order

Search for...

List Food

Dashboard / List Food

Create New

Show 10 entries

Search:

ID	Image	Name	Description	Price	Count	Action
1		Steak thân ngoại bò úc	Steak thân ngoại bò úc	15000	100	Action
2		Súp kem bí đỏ	Súp kem bí đỏ	10000	57	Action
3		Steak thân lưng bò úc	Steak thân lưng bò úc	7000	1000	Action

Logged in as: Start Bootstrap

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

Back to list

Name

Description

Price

0

Count

0

ImageData

Chọn tệp Không có tệp nào được chọn

Categories

cơm trưa
Drink
Cakes
Bò

Create

Back to List

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

Delete Food

Dashboard / Delete Food

Back to list

ID

1

Name

Steak thân ngoại bò úc

Description

Steak thân ngoại bò úc

Price

15000

Count

100

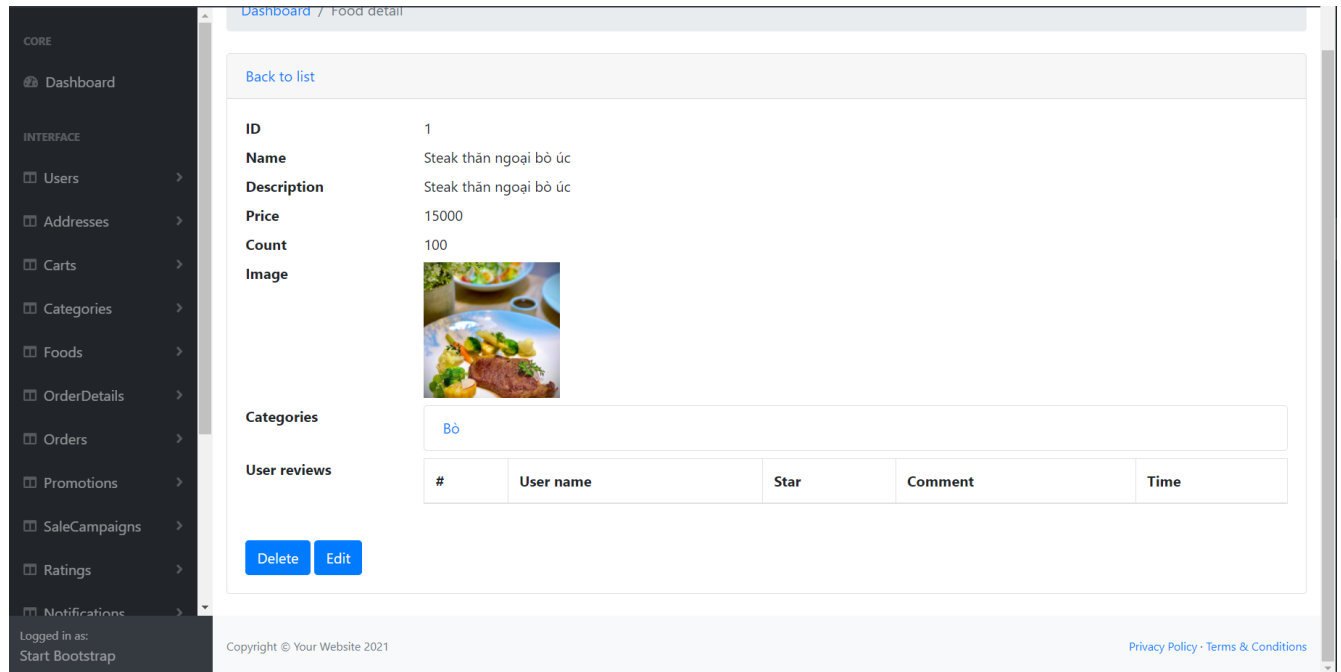
Delete

Back to List

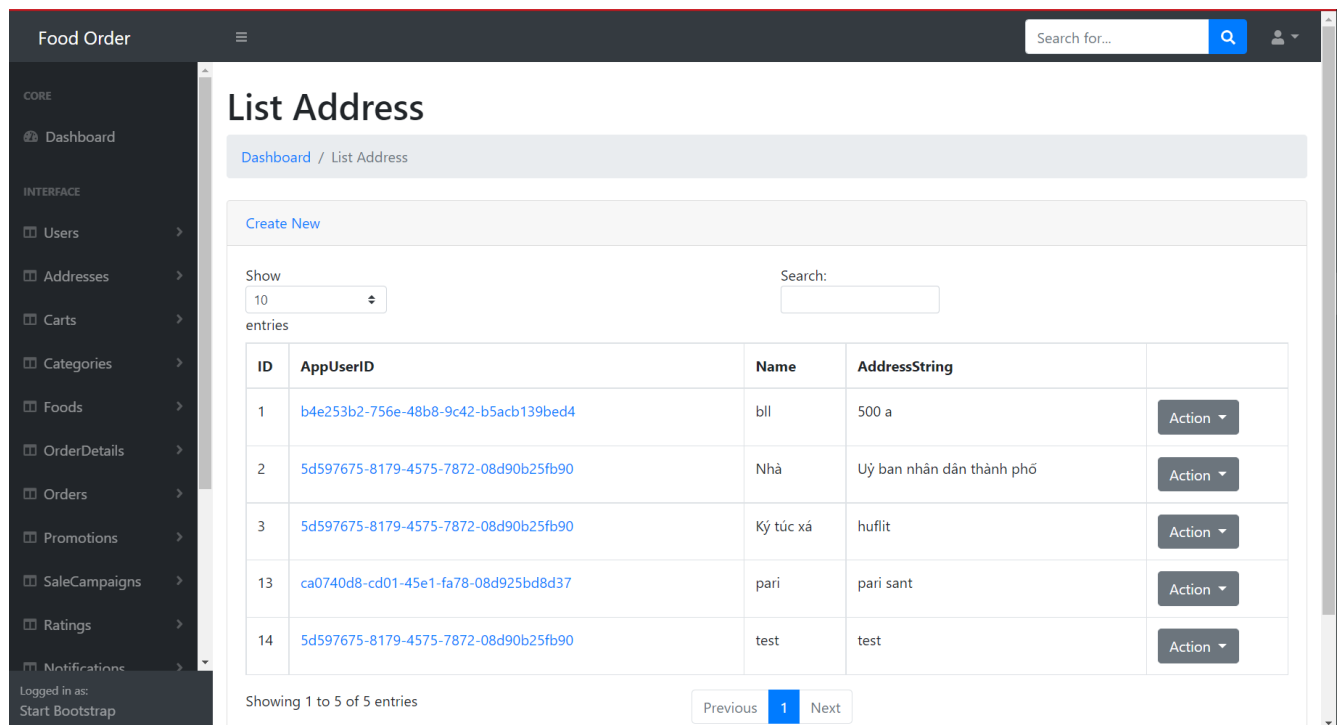
Copyright © Your Website 2021

Privacy Policy · Terms & Conditions

Đề tài: Ứng dụng đặt đồ ăn với Flutter và Dart



+ Các trang quản lý khác



Đề tài: Ứng dụng đặt đồ ăn với Flutter và Dart

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

List user

Dashboard / List user

Create New

Show 10 entries

Search:

Username	FirstName	LastName	DateOfBirth	Email	
admin	Khang	Lien Minh	4/4/2021 12:00:00 AM	khoid1@gmail.com	Action
admin	Mary	Jame	5/21/1999 12:00:00 AM	jame@gmail.com	Action
admin2	admin	admin	4/29/2021 12:00:00 AM	khoid@gmail.com	Action
dung	dung	dung	5/5/2021 12:00:00 AM	dung@gmail.com	Action
duong	duong	duong	5/4/2021 12:00:00 AM	duong@gmail.com	Action
empape	Em	Pape	6/15/2021 12:00:00 AM	empape@gmail.com	Action

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap





List Category

Dashboard / List Category

Create New

Show 10 entries

Search:

ID	Image	Name	Description	
1		cơm trưa	com7	Action
2		Drink	Nước giải khát, trà sữa, chè, ...	Action
3		Cakes	Bánh	Action
9		Bò	Các món ăn chế biến từ thịt bò	Action

Đề tài: Ứng dụng đặt đồ ăn với Flutter và Dart

Food Order

Dashboard

Cart

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

AppRoles

Pages

ADDONS

Charts

Tables

Logged in as:
Start Bootstrap

Search for...

Q

List Order

Dashboard / List Order

Create New

Đơn hàng đang xử lý

Show 10 entries

Search:

ID	Receiver	FinalTotalPrice	CreatedDate	Order Status	Promotion
Search	Search	Search	Search	Search	Search
45	admin admin	0	5/31/2021 1:16:18 PM	Đang chuẩn bị	3
47	admin admin	0	5/31/2021 1:20:38 PM	Đang chuẩn bị	1
58	admin admin	0	6/1/2021 2:14:58 PM	Đang chuẩn bị	3
59	admin admin	0	6/1/2021 2:16:39 PM	Đang chuẩn bị	3
60	admin admin	0	6/1/2021 2:48:31 PM	Đang chuẩn bị	3

Food Order

Dashboard

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

Search for...

Q

List SaleCampaigns

Dashboard / List SaleCampaigns

Create New

Show 10 entries

Search:

ID	Name	StartDate	EndDate	Percent	Enabled	
1	Ăn no không lo giá Giảm 15%	5/11/2021 12:00:00 AM	6/30/2021 12:00:00 AM	15	<input checked="" type="checkbox"/>	Action
2	test	5/27/2021 12:00:00 AM	6/1/2021 12:00:00 AM	30	<input type="checkbox"/>	Action

Showing 1 to 2 of 2 entries

Previous 1 Next

Copyright © Your Website 2021

Privacy Policy · Terms & Conditions

Đề tài: Ứng dụng đặt đồ ăn với Flutter và Dart

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

List Promotion

Dashboard / List Promotion

Create New

Show 10 entries

Search:

ID	Name	Code	StartDate	EndDate	UseTimes	Percent	Enabled	
1	Chào thành viên mới Giảm 50%	THANHVIENMOI	6/3/2021 12:00:00 AM	1/1/2022 12:00:00 AM	3	50	<input checked="" type="checkbox"/>	Edit Details Delete
3	Ăn ở nhà cũng ngon Khao 60%.	KHAO60	5/5/2021 12:00:00 AM	6/5/2021 12:00:00 AM	3	60	<input checked="" type="checkbox"/>	Edit Details Delete

Showing 1 to 2 of 2 entries

Previous 1 Next

Copyright © Your Website 2021

Privacy Policy · Terms & Conditions

Food Order

Search for...

CORE

Dashboard

INTERFACE

Users

Addresses

Carts

Categories

Foods

OrderDetails

Orders

Promotions

SaleCampaigns

Ratings

Notifications

Logged in as:
Start Bootstrap

List AppRole

Dashboard / List Food

Create New

Show 10 entries

Search:

Id	Description	Name	
1267bf30-8eea-4370-bbcd-12f74e29c656	This is Manager role.	manager	Edit Details Delete
74d17d8b-3f91-4d3a-893f-09b4e51e3834	This is User role.	user	Edit Details Delete
b9e08f48-883c-42dc-a700-da5cf1d81aa3	This is Admintrator role.	admin	Edit Details Delete

Showing 1 to 3 of 3 entries

Previous 1 Next

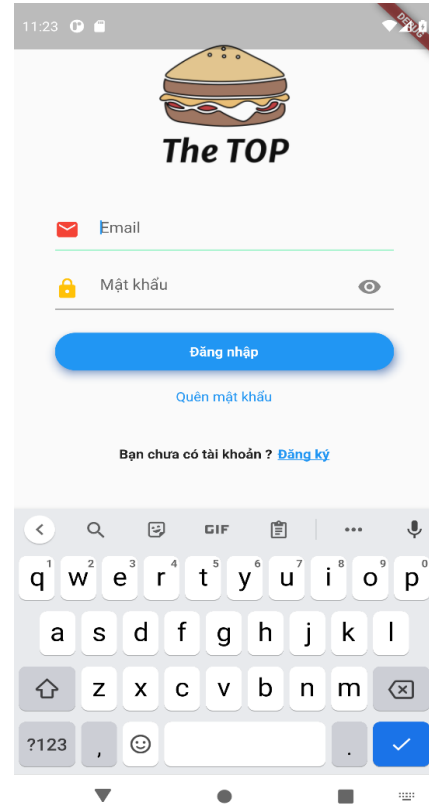
Copyright © Your Website 2021

Privacy Policy · Terms & Conditions

➤ Mobile (Flutter)

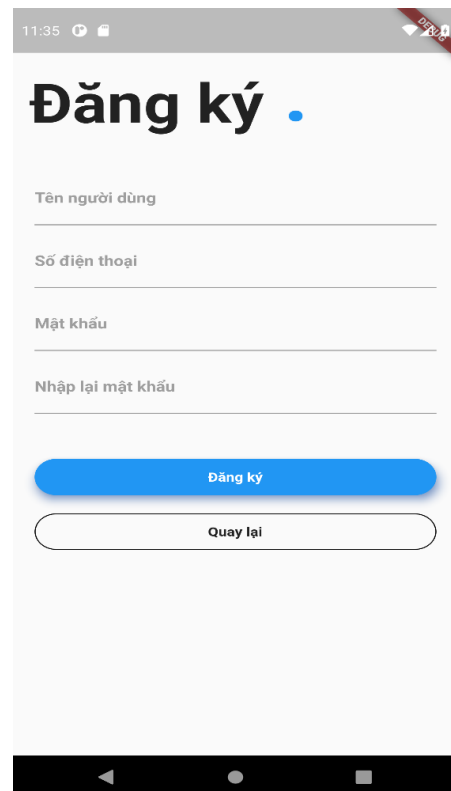
a) Đăng nhập khách hàng.

- Khi người dùng nhập thông tin và submit thì Bloc sẽ gửi event đến và thực thi service từ API sẽ thực hiện xác nhận tài khoản đã đăng kí hay chưa, trả về kết quả đăng nhập về State rồi cập nhật về UI
- API sử dụng
+ `/api/Users/Login`



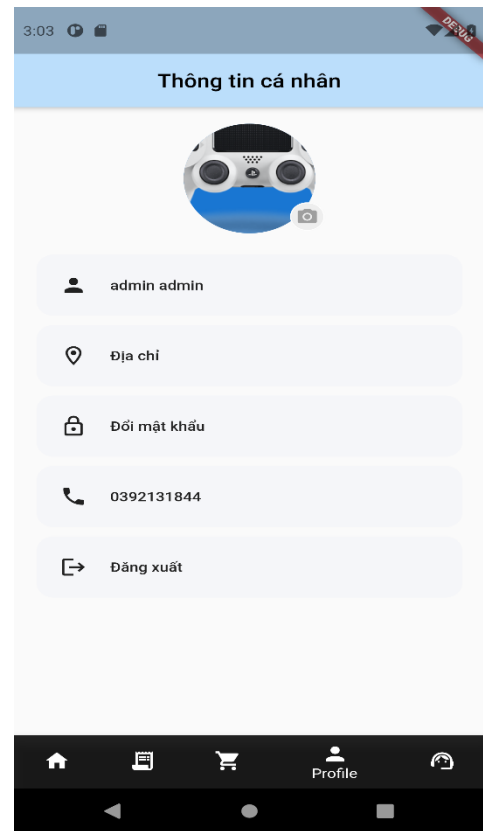
b) Đăng ký khách hàng

- Yêu cầu người dùng đăng kí tài khoản. Khi người dùng submit, Bloc sẽ gửi event tới service đăng kí, thông tin sẽ được lưu lên API rồi trả về State rồi cập nhật về UI
- API sử dụng: `/api/Users/Register`



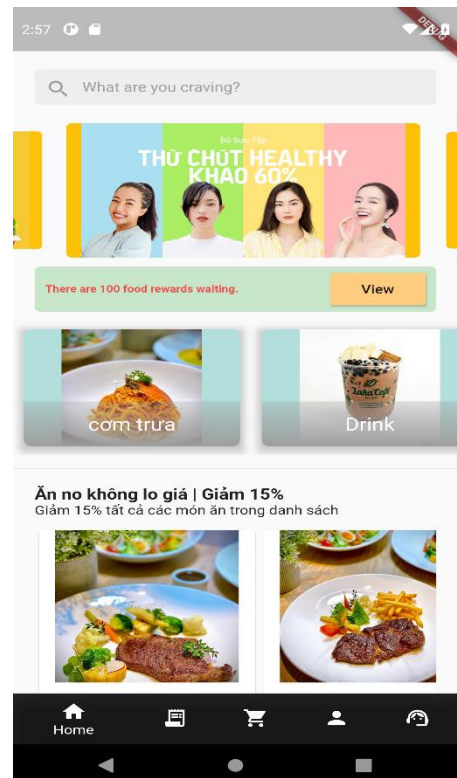
c) *Thông tin của khách hàng*

- Sau khi đăng nhập thành công, Ứng dụng sẽ chuyển hướng đến trang chủ của ứng dụng.
- Trang thông tin của khách hàng sẽ được hiển thị.
- API sử dụng: `/api/Users/{id}`



d) *Trang chủ*

- Xây dựng giao diện màn hình chính là bước hết sức quan trọng vì đây là màn hình đầu tiên khách hàng nhìn thấy sau khi mở ứng dụng, gây được nhiều ấn tượng từ lần đầu sử dụng ứng dụng, Chính vì vậy màn hình chính đòi hỏi được thiết kế đẹp mắt ấn tượng, cùng việc thiết kế phải đơn giản, dễ sử dụng.



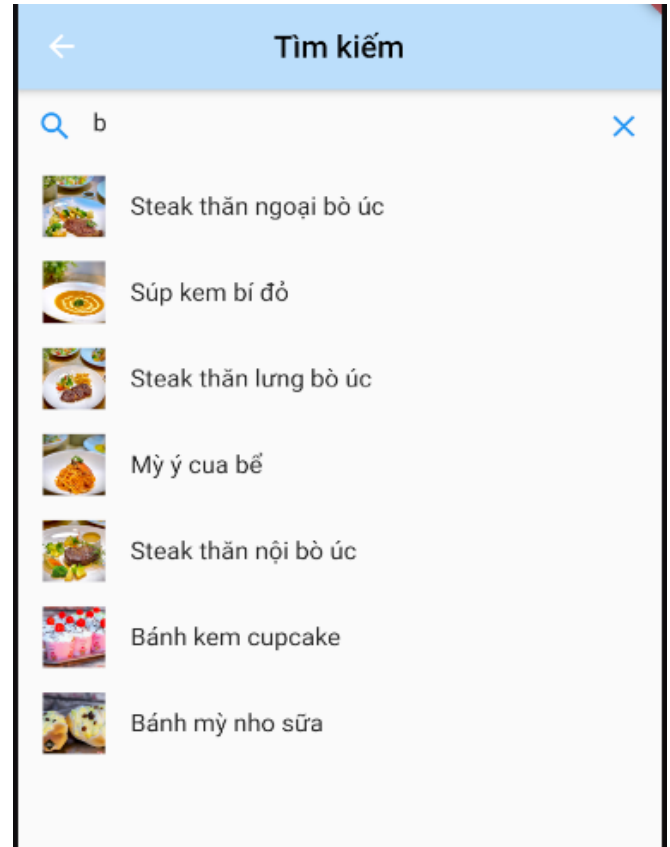
- API sử dụng
- /api/SaleCampaigns
- /api/Category
- /api/Food/best_selling

e) *Tìm kiếm.*

- Một tính năng cực kì cần thiết cho một ứng dụng bán hàng là tìm kiếm. Việc tìm kiếm một sản phẩm theo ý thích giữa hàng trăm, hàng nghìn sản phẩm là điều rất khó khăn thì tính năng tìm kiếm sản phẩm theo tên là một trợ thủ đắc lực trong việc tìm kiếm sản phẩm. Người dùng chỉ cần nhập tên sản phẩm bất kì mình muốn tìm kiếm trên thanh công cụ tìm kiếm, sau đó nhấn nút tìm, ứng dụng sẽ tự trả về kết quả dựa theo từ khóa người dùng cung cấp.
- API sử dụng: /api/Food

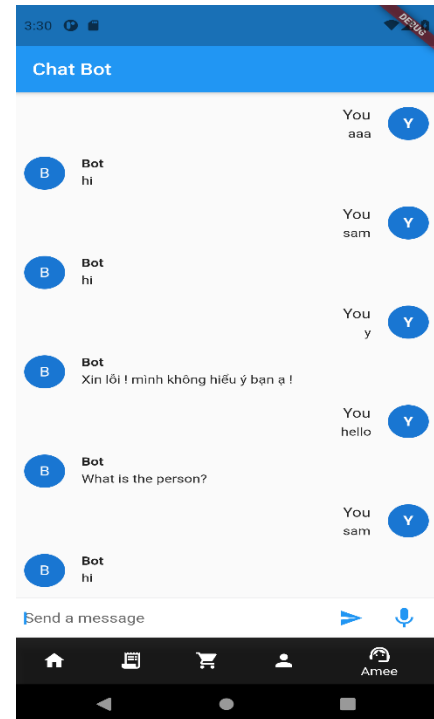
f) *Thông tin sản phẩm.*

- Trang thông tin sản phẩm được thiết kế với đầu trang là hình ảnh bao gồm rating và promotion.
- Trong phần thông tin sản phẩm thì mô tả chi tiết được hiển thị tab 1 và đánh giá ở tab 2
- Phần cuối có giá chọn số lượng đặt hàng
- API sử dụng /api/Food/{id}



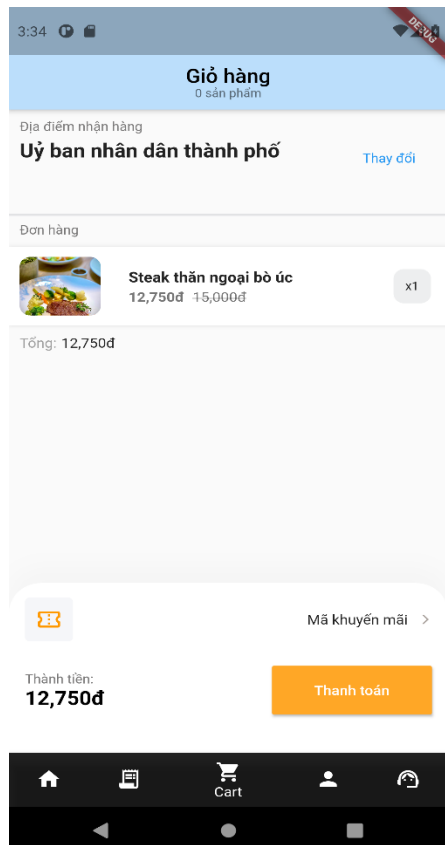
g) Nhắn tin với nhà bán hàng (Chat bot) – đang phát triển

- Một trong những tính năng đặc biệt là người khách hàng có thể nhắn tin trao đổi thắc mắc với nhà bán hàng.
- Tag thông tin sản phẩm cần hỏi được gắn trên đầu trang.
- Người dùng có thể gửi hình ảnh cho người bán



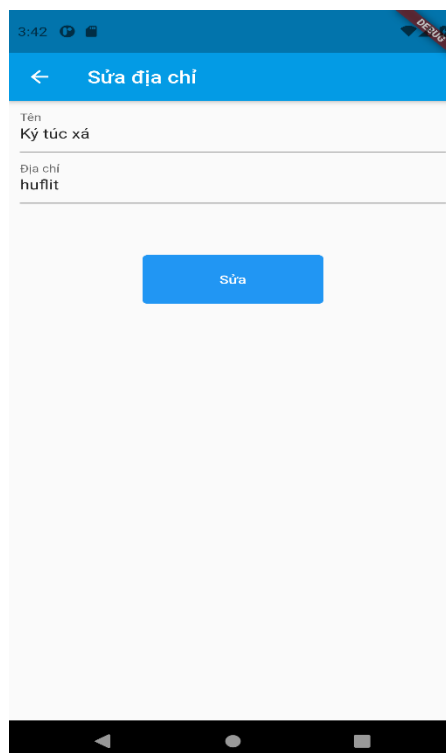
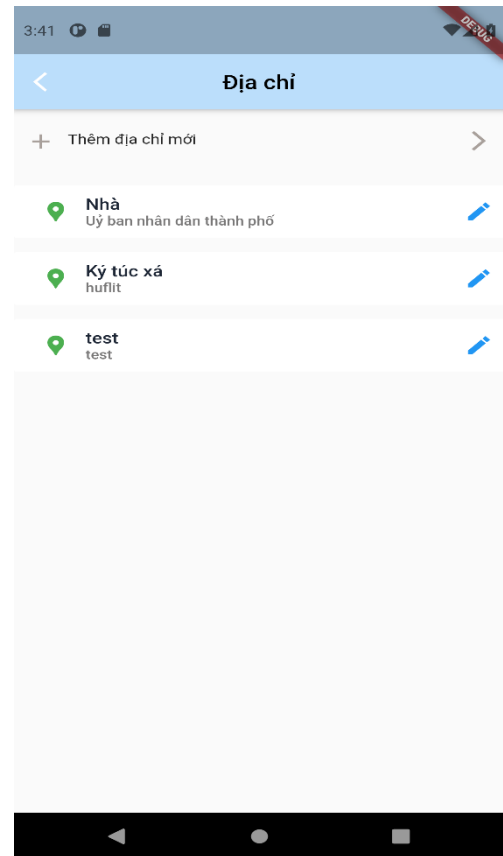
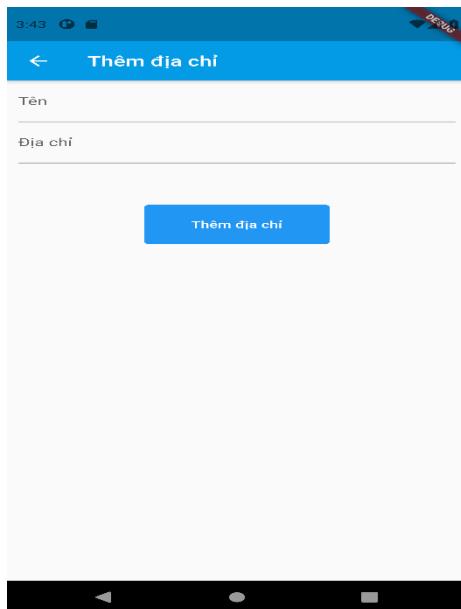
h) Thông tin giỏ hàng.

- Hiện thị toàn bộ giỏ hàng mà khách hàng đã thêm, hình ảnh, tên và số lượng.
- Người dùng có thể xóa giỏ hàng khi thay đổi ý kiến, trở về và tiếp tục mua sắm.
- API sử dụng: /api/Carts



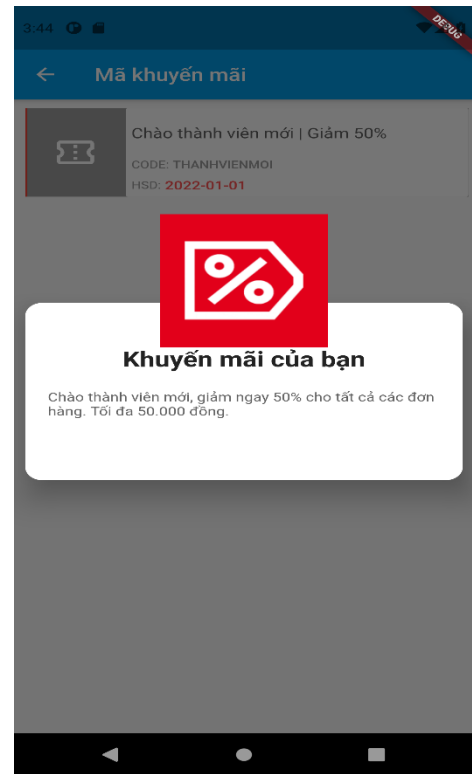
i) Quản lý địa chỉ

- Lựa chọn địa chỉ giao hàng. Nếu chưa có thì có thể thêm địa chỉ hoặc chỉnh sửa và xóa địa chỉ cũ
- API sử dụng: /api/Addresses



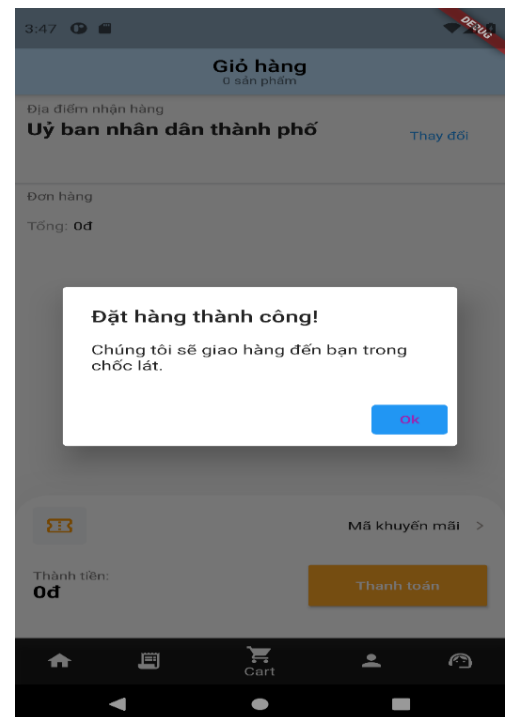
j) Hiển thị khuyến mãi

- Khi người dùng nhấp vào khuyến mãi thì ứng dụng hiển thị một Dialog giới thiệu
- API sử dụng: `/api/Promotions/{id}`



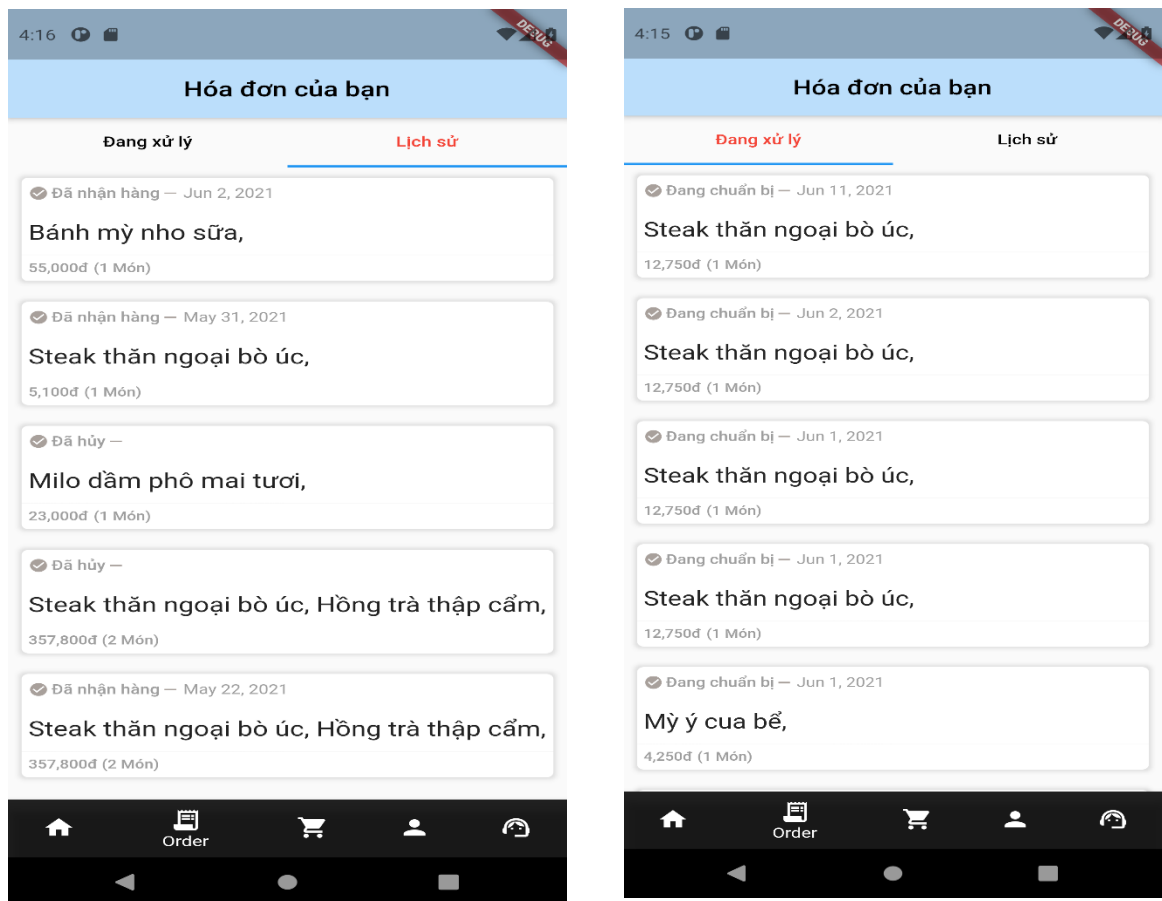
k) Đặt hàng thành công

- Khi người dùng nhấp vào đặt hàng thì ứng dụng hiển thị một Dialog giao hàng thành công
- Xóa dữ liệu giỏ hàng
- API sử dụng: `/api/Orders`



l) Xem thông tin đơn hàng.

- Kiểm tra, theo dõi đơn hàng hiện tại là một trong những tính năng hay và cần thiết của một ứng dụng mua bán, Ở đó, người dùng có thể xem được trạng thái đơn hàng hiện tại của mình như thế nào, xem đơn hàng đã được người bán nhận được chưa? Hàng đã được đóng gói hay đang trong quá trình vận chuyển? Nhờ vào việc chủ động trong việc kiểm tra đơn hàng khách hàng có thể biết được tình trạng đơn hàng, quá trình giao nhận góp phần tạo thêm sự yên tâm và niềm tin đối với cá nhân, doanh nghiệp thực hiện việc mua bán. Đồng thời khắc phục những sự cố giao nhận(nếu có) một cách sớm nhất. Về mặt giao diện, màn hình bao gồm một danh sách các đơn hàng, ở từng đơn hàng sẽ được hiển thị những thông tin như: ngày đặt hàng, mã số đơn hàng và trạng thái hiện tại của đơn hàng.
- API sử dụng: /api/Orders/user - /api/OrderStatus



CHƯƠNG 4: KẾT LUẬN

4.1. CÁC KẾT QUẢ ĐẠT ĐƯỢC

Ứng dụng bán hàng trên di động với Flutter và chọn cơ sở dữ liệu là API online đã đáp ứng được nhu cầu cơ bản của một ứng dụng bán hàng online. Đem lại sự tiện lợi, an toàn, nhanh chóng đáp ứng mọi yêu cầu thiết yếu của một ứng dụng bán hàng online đem lại sự trải nghiệm tuyệt vời. Bên cạnh đó, nó còn mang lại sự nhất quán và an toàn trong công tác quản lý của các cá nhân doanh nghiệp sử dụng dịch vụ. Thông qua việc xây dựng chương trình, em cũng đã nắm rõ hơn về kiến thức về phân tích thiết kế hệ thống đã được học. Hơn thế nữa, em còn học được cách lên kế hoạch cho một dự án.

4.2. HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN

Hạn chế

- Tính năng đăng kí tài khoản bằng email, chưa hoàn thành việc xác thực email có chính chủ hay không, chưa hoàn thành tính năng xác nhận số điện thoại qua Firebase Phone Authentication qua việc xác nhận OTP.
- Chưa thật sự tiện lợi trong việc giao tiếp giữa cá nhân, doanh nghiệp, người sử hữu ứng dụng với người tiêu dùng.
- Chưa tiện lợi trong việc thanh toán, hệ thống chưa đa dạng các hình thức thanh toán.

Hướng phát triển

Những hạn chế của chương trình cũng chính là những mục tiêu định hướng phát triển sau này của chương trình, nhằm phát triển chương trình hoàn thiện hơn, xây dựng giao diện người dùng thân thiện, dễ sử dụng đem lại sự tiện lợi, thoải mái nhất cho người sử dụng

TÀI LIỆU THAM KHẢO

- [1] Eric Windmill, *Flutter in Action*.
- [2] Dart document, <https://dart.dev/guides>.
- [3] Flutter document, <https://flutter.dev/docs>.
- [4] Provider Pub package, Published Jun 5- 2020, <https://pub.dev/packages/provider>.
- [5] Cloud Firestore Pub package(Jun 20-2020), https://pub.dev/packages/cloud_firestore
- [6] Firebase Auth Pub package, https://pub.dev/packages/firebase_auth
- [7] <https://flutter.dev/docs/cookbook/networking/fetch-data>
- [8] <https://www.c-sharpcorner.com/article/flutter-rest-api/>