LỜI CẢM ƠN

Sau 4,5 năm học tập và rèn luyện tại khoa Công nghệ thông tin trường Đại học Thủy Lợi, đến nay em đã hoàn thành chương trình học tập và Đồ án tốt nghiệp.

Xin chân thành cảm ơn Ban giám hiệu nhà trường đã quan tâm tạo điều kiện thuận lợi để chúng em học tập và rèn luyện tốt.

Xin chân thành cảm ơn các Thầy, Cô giáo trong khoa Công nghệ thông tin cũng như các khoa khác đã luôn nhiệt tình giúp đỡ và tạo điều kiện tốt nhất cho em trong suốt quá trình học tập tại trường.

Xin cảm ơn các tác giả, các nguồn tài liệu… đã giúp tôi có thêm kiến thức để hoàn thiện đồ án này.

Em xin chân thành cảm ơn Thạc sĩ ***Nguyễn Văn Nam***, người đã trực tiếp hướng dẫn và tận tình giúp đỡ em trong suốt thời gian qua. Em xin gửi lời cảm ơn chân thành và sâu sắc tới Thầy vì tất cả những sự giúp đỡ quý báu đó.

Xin chân thành cảm ơn các bạn sinh viên lớp 55TH2 - những người bạn đã nhiệt tình giúp đỡ, chia sẻ những kinh nghiệm trong học tập cũng như trong cuộc sống để tôi mãi không quên những năm tháng học tập dưới mái trường này.

Và cuối cùng, con xin tỏ lòng biết ơn sâu sắc đến ông bà, bố mẹ, và cả gia đình đã nuôi dưỡng, giáo dục và luôn động viên để con có ngày hôm nay.

Hà Nội, Tháng 11 năm 2017.

*Người thực hiện*

***Phạm Thăng Long***

MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc497356028)

[MỤC LỤC 2](#_Toc497356029)

[CHƯƠNG 1: GIỚI THIỆU CHUNG 16](#_Toc497356030)

[1.1. Mô tả bài toán 16](#_Toc497356031)

[1.2. Mục tiêu và những yêu cầu khi xây dựng hệ thống 17](#_Toc497356032)

[1.3. Giới hạn đầu bài 19](#_Toc497356037)

[1.4. Mục đích và ý nghĩa của đề tài 19](#_Toc497356038)

[1.5. Các yêu cầu về phần cứng và phần mềm 20](#_Toc497356041)

[CHƯƠNG 2: KIẾN THỨC CHUNG 21](#_Toc497356042)

[2.1. Giới thiệu về đa nền tảng và lập trình đa nền tảng 21](#_Toc497356043)

[2.2. Kiến trúc hệ thống 23](#_Toc497356045)

[2.3. Phía Server 24](#_Toc497356046)

[2.3.1. ASP.NET 24](#_Toc497356047)

[2.3.2. SQL Server 28](#_Toc497356048)

[2.4. Phía Client. 40](#_Toc497356049)

[2.4.1. AngularJS 2.0 40](#_Toc497356050)

[2.4.2. Framework Ionic 2 46](#_Toc497356051)

[2.5. Tổng quan về mã vạch 58](#_Toc497356052)

[2.5.1. Mã vạch là gì? 58](#_Toc497356053)

[2.5.2. Các loại mã vạch 58](#_Toc497356054)

[2.5.3. Mã vạch được ứng dụng để mã hóa những gì? 62](#_Toc497356055)

[2.5.4. Tìm hiểu về mã vạch của hàng hóa 63](#_Toc497356056)

[CHƯƠNG 3: KHẢO SÁT YÊU CẦU NGƯỜI DÙNG 68](#_Toc497356057)

[3.1. Mục đích hệ thống 68](#_Toc497356058)

[3.2. Xác định vai trò người dùng 68](#_Toc497356060)

[3.3. Yêu cầu chức năng cho từng vai trò người dùng 68](#_Toc497356069)

[3.4. Mô tả chi tiết chức năng cho từng nhóm người dùng 70](#_Toc497356104)

[3.4.1. Đối với khách truy cập vào app 70](#_Toc497356105)

[3.4.2. Đối với thành viên (người đã đăng ký tài khoản) 73](#_Toc497356110)

[3.4.3. Đối với quản trị 77](#_Toc497356114)

[CHƯƠNG 4: PHÂN TÍCH HỆ THỐNG 89](#_Toc497356115)

[4.1. Mô tả bài toán 89](#_Toc497356116)

[4.2. Mô hình vai trò người dùng 89](#_Toc497356118)

[4.3. Tổng quan các usecase của hệ thống 89](#_Toc497356120)

[4.4. Sơ đồ quy trình 90](#_Toc497356122)

[4.4.1. Đối với khách 90](#_Toc497356123)

[4.4.2. Đối với thành viên 96](#_Toc497356124)

[4.4.3. Đối với quản trị 104](#_Toc497356127)

[CHƯƠNG 5: PHÂN TÍCH THIẾT KẾ 127](#_Toc497356128)

[5.1. Biểu đồ lớp 127](#_Toc497356129)

[5.1.1. Đối với khách truy cập vào app 127](#_Toc497356130)

[5.1.2. Thành viên( người đã đăng ký tài khoản với hệ thống) 134](#_Toc497356151)

[5.1.3. Quản trị hệ thống 144](#_Toc497356152)

[5.2. Đồng bộ hóa các thực thể 159](#_Toc497356153)

[CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN PHẦN MỀM 175](#_Toc497356155)

[6.1. Kết luận 175](#_Toc497356156)

[6.2. Hướng phát triển đề tài 176](#_Toc497356157)

[TÀI LIỆU THAM KHẢO 177](#_Toc497356158)

[Các tài liệu tham khảo online 177](#_Toc497356159)

[Giáo trình 177](#_Toc497356160)

1. GIỚI THIỆU CHUNG.
   1. Mô tả bài toán.

Trong thời buổi kinh tế mở cửa, chẳng thể tránh khỏi chuyện "vàng thau lẫn lộn", các sản phẩm được làm giả được bày bán nhan nhản, tràn ngập thị trường. Và nếu bạn là người chuộng dùng hàng chính hãng, tất nhiên bạn sẽ chẳng bao giờ muốn phải bỏ ra một khoản tiền không nhỏ chỉ để mua hàng nhái, hàng kém chất lượng cả. Sau một thời gian bị lừa, người tiêu dùng cũng tự trang bị được cho mình những kiến thức phù hợp để phân biệt hàng thật hàng giả. Một trong số đó là cách kiểm tra mã vạch - barcode và QR code.

Ngày nay, điện thoại di động đã trở thành một trong những công cụ liên lạc thiết yếu của con người. Không chỉ thế, ngoài chức năng liên lạc, cùng với sự phát triển của công nghệ, kết nối toàn cầu, điện thoại di động còn được trang bị nhiều ứng dụng giải trí, định vị, mua sắm, thanh toán trực tuyến,… Sự bùng nổ về cả số lượng và chất lượng của điện thoại thông minh một lần nữa khẳng định vị trí quan trọng của nó trong cuộc sống.

Từ thực tế kiểm tra mã vạch để phân biệt hàng thật hàng giả và sự phổ biến của điện thoại thông minh hiện nay thì việc xây dựng ứng dụng quét mã vạch của sản phẩm ngay trên chính chiếc điện thoại để kiểm tra sản phẩm là hàng chính hãng hay hàng nhái, hàng kém chất lượng là cần thiết,đáp ứng các nhu cầu của xã hội, hỗ trợ cho việc kiểm tra thông tin sản phẩm ngay từ phía người tiêu dùng giúp việc mua bán diễn ra một cách nhanh chóng, tiết kiệm.

* 1. Mục tiêu và những yêu cầu khi xây dựng hệ thống
     1. Mục tiêu xây dựng hệ thống.
* Tìm hiểu tổng quan Ionic framework và Apache Cordova. Trên cơ sở đó xây dựng được ứng dụng quét mã vạch hiển thị thông tin sản phẩm, thể hiện đầy đủ quy trình từ việc quét mã vạch đến hiển thị thông tin chi tiết sản phẩm
  + 1. Những yêu cầu đặt ra cho hệ thống.
* Tạo sự thân thiện, dễ sử dụng cho người sử dụng.
* Hệ thống có thể tìm kiếm, quét mã vạch của sản phẩm bất kỳ theo ý người dùng, sau đó hiển thị thông tin về sản phẩm, đánh giá, bình luận gợi ý địa điểm mua sản phẩm.
  + 1. Hoạt động của hệ thống.

Trong hoạt động của hệ thống bao gồm 3 hoạt động chính:

* Hoạt động của khách.
* Hoạt động của thành viên.
* Hoạt động của quản trị.
* **Hoạt động của khách.**
* Xem thông tin chi tiết sản phẩm
* Có thể sử dụng app để quét mã vạch của sản phẩm
* Tìm kiếm sản phẩm theo tên hoặc theo mã vạch của sản phẩm
* Có thể đăng ký tài khoản và đăng nhập vào hệ thống
* **Hoạt động của thành viên.**

Ngoài các chức năng như với khách, thành viên của hệ thống có thêm các chức năng:

* Được đánh giá về sản phẩm đang xem
* Bình luận về sản phẩm đang xem
* Đặt câu hỏi về sản phẩm đang xem
* Trả lời câu hỏi về sản phẩm đang xem
* Đăng xuất tài khoản khỏi hệ thống
* **Hoạt động của quản trị.**

Tài khoản của quản trị có chức năng như của thành viên nhưng có quyền cao hơn khi đăng nhập vào server, tại đây quản trị các chức năng:

+ Quản lý sản phẩm

* Thêm sản phẩm
* Sửa sản phẩm
* Xóa sản phẩm

+ Quản lý danh mục sản phẩm

* Thêm danh mục
* Sửa danh mục
* Xóa danh mục

+ Quản lý tài khoản

* Thêm tài khoản
* Sửa tài khoản
* Xóa tài khoản

+ Quản lý bình luận, câu hỏi, câu trả lời

* Ẩn bình luận
* Ẩn câu hỏi
* Ẩn câu trả lời
  + 1. Phương pháp giải quyết đề tài.

Từ các yêu cầu đặt ra của đề tài, em tiến hành phân tích và đưa ra phương

hướng giải quyết theo các bước như sau:

* Tìm hiểu về các ngôn ngữ lập trình và các dạng cơ sở dữ liệu, từ đó đưa ra giải pháp phù hợp để tiến hành xây dựng.
* Bắt tay vào xây dựng ứng dụng.
* Tiến hành kiểm tra và chạy thử.
* Thay đổi, bổ sung, khắc phục các sự cố, lỗi, từ đó đưa ứng dụng vào sử dụng trong thực tiễn.
  1. Giới hạn đầu bài.

Với thời gian 3 tháng em sẽ xây dựng được ứng dụng chạy trên các hệ điều hành của điện thoại đáp ứng được các yêu cầu trên.Tương lai em sẽ phát triển thêm 1 số tính năng tiện ích mới.

* 1. Mục đích và ý nghĩa của đề tài.
     1. Mục đích.

Mục đích của đề tài là xây dựng ứng dụng trên mobile chạy đa nền tảng cung cấp thông tin,phục vụ nhu cầu kiểm tra mã vạch sản phẩm của người tiêu dùng mọi lúc mọi nơi.

Nhằm đáp ứng được nhu cầu kiểm tra hàng thật hay hàng giả theo mong muốn của mình mà không mất nhiều thời gian và phương tiện đi lại,tiết kiệm nhiều chi phí.

* + 1. Ý nghĩa của đề tài.

Qua việc xây dựng ứng dụng này, em có thể áp dụng những kiến thức được bồi dưỡng trong quá trình học tập tại trường vào công việc thực tế. Từ đó, em đã tích lũy cho bản thân những kinh nghiệm quý báu mà sau này sẽ là nền tảng giúp cho việc lập nghiệp của bản thân.

* 1. Các yêu cầu về phần cứng và phần mềm.

|  |  |  |
| --- | --- | --- |
|  | ***Serve*** | ***Client*** |
| Phần cứng | * Ram 4GB hoặc hơn. * CPU 1.8 GHZ Xeon hoặc hơn. * HDD 100GB hoặc hơn.   + Kết nối Intemet. | * Ram 512Mb hoặc hơn. * Có camera * Kết nối Internet. |
| Phần mềm | * Visual Stuido * Sql server | * Android 4.4 trở lên * Windows 10 Universal App * IOS 8 trở lên. |

1. KIẾN THỨC CHUNG.
   1. Giới thiệu về đa nền tảng và lập trình đa nền tảng
      1. Đa nền tảng

Đa nền tảng (tiếng Anh: cross-platform hay multi-platform) là một thuật ngữchỉ các phần mềm máy tính, các ứng dụng điện thoại hay các phương thức điện toán được thực thi đầy đủ và vận hành cùng nhau trên nhiều nền tảng máy tính và điện thoại và được gắn liền với công nghệ điện toán.Phần mềm đa nền tảng có thể được chia thành hai loại:

* Loại yêu cầu phải thiết kế hoặc biên dịch từng phiên bản cho mỗi nền tảng nó hỗ trợ.
* Loại có thể chạy trực tiếp trên bất cứ nền tảng nào mà không cần sự chuẩn bị thêm đặc biệt nào - chẳng hạn như phần mềm được viết bằng một ngôn ngữ thông dịch hay được biên dịch trước sang mã bytecode có khả năng di động bằng một trình thông dịch hay các gói run-time thường dùng hoặc là thành phần tiêu chuẩn trên mọi nền tảng.  
  + 1. **Lập trình đa nền tảng**
* **Giới thiệu sơ lược về lập trình đa nền tảng**

Là khái niệm dùng để chỉ việc xây dựng và phát triển các ứng dụng có thể chạy đa nền tảng. Nhằm mục đích để người dùng ở những nền tảng khác nhau có thể sử dụng ứng dụng như nhau và nhà phát triển sẽ tiết kiệm thời gian cũng như tiền bạc để đầu tư vào việc phát triển chúng. Hiện nay, có rất nhiều nền tảng có thể kể đến như:

* Máy tính: Microsoft Window, Linux, Mac OS X, Ubuntu, MS DOS…
* Điện thoại (tiếng Anh: Mobile-platform): iOS, Android, Window Phone, Symbian OS…

Với việc thiết bị cầm tay đang ngày càng phổ biến đặc biệt là điện thoại di động, vì nó có thể sử dụng ở bất kỳ đâu, bất kỳ lúc nào và dễ dàng thuận tiện cho việc mang theo. Do đó, sẽ có rất rất nhiều người sử dụng và liên tục phát triển. Các nhà phát triển nền tảng cũng như ứng dụng đang tập trung khai thác vào mảng này.

* + 1. **Đa nền tảng trên thiết bị di động**

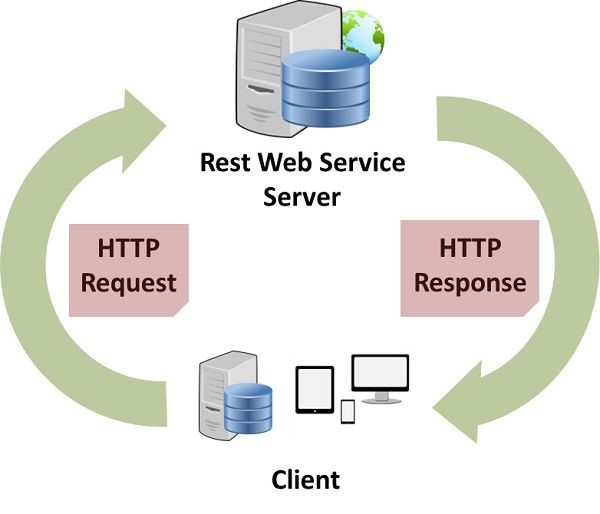
Từ trước tới nay mỗi nền tảng cho thiết bị di động đều có các ngôn ngữ lập trình cũng như trình biên dịch riêng hay được gọi với cái tên Native App như:

* Android-Platform: sử dụng công cụ Android Studio hoặc Eclipse, ngôn ngữ Java.iOS-Platform: sử dụng công cụ Xcode, ngôn ngữ Objective-C hoặc Swift.
* Window Phone: sử dụng công cụ Visual Studio, ngôn ngữ C# và mã XAML.

Ngoài ra, các công nghệ phát triển ứng dụng đan nền tảng được chia làm hai loại chính là Mobile Web Application và Hybrid Application

* + 1. **Phân biệt Mobile Web Application và Hybrid Application**
* **Mobile Web Application:** hay Responsive Web là ứng dụng được lập trình ra sẽ chạy trên nền tảng web, được nhúng file HTML, CSS, JS vào trong đó hoặc nó truy cập trực tiếp vào website nhà cung cấp. Điểm mạnh của nó là có thể tạo app rất nhanh, phù hợp với những ứng dụng có tính marketing cao, không phụ thuộc hệ điều hành, rất tiết kiệm chi phí (không phải thuê nhân lực cho lập trình từng nền tảng). Tuy nó nó chỉ chạy trên nền tảng web nhưng nó vẫn giải quyết được vẫn đa nền tảng vì mọi nền tảng di động đều sử dụng ứng dụng web được. Nhưng bù lại nó hầu như không có khả năng tương tác với hệ điều hành, với phần cứng của máy, hiệu năng thấp và cho trải nghiệm người dùng không cao và đặc biệt chỉ hoạt động khi có internet. Do vậy nó không phù hợp với ứng dụng có tính phức tạp cao, yêu cầu nghiệp vụ và trải nghiệm người dùng lớn.
* **Hybrid Application:** hay nó là một đứa con lai của Native app với MobileWeb Application. Bản chất nó sử dụng giao diện được viết bằng mã HTML, CSS, JS nhưng lại cho phép gọi được các API của hệ thống để thao tác với hệ điều hành (đọc ghi file, blutooth, camera, sensor…). Vì vậy, Hybrid Application vừa tận dụng điểm mạnh của Mobile Web Application vừa khai thác hết khả năng thao tác trên hệ thống của Native App. Trong tương lai Hybrid hứa hẹn sẽ là sự thay thế hoàn hảo cho các ứng dụng mobile hiện nay. Đại diện cho công nghệ Hybrid có thể kể đến PhoneGap, Ionic, Trigger.IO,

* 1. Kiến trúc hệ thống.



*Hình 1. Mô hình kiến trúc hệ thống.*

Hệ thống có 2 thành phần chủ yếu, đó là:

* **Client**: là ứng dụng phía người dùng.Ứng dụng hiển thị giao diện để người dùng có thể dễ dàng tương tác với hệ thống, người dùng có thể tra, xem chi tiết sản phẩm…Hầu hết các chức năng đều phải sử dụng kết nối mạng vì client kết nối với server thông qua giao thức HTTP để lấy dữ liệu về và theo chiều ngược lại client cũng gửi dữ liệu lên server, server kết nối với database và thực hiện truy vấn.
* **Server**: là thành phần quan trọng của hệ thống.Nó cung cấp các web-service phục vụ cho client.Server dùng database để lưu trữ dữ liệu.Khi có yêu cầu, server sẽ thực hiện kết nối với database và thực hiện câu truy vấn.
  1. Phía Server.

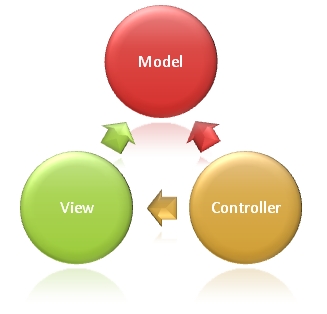
Server là nơi xử lý tất cả các yêu cầu từ phía client như: thêm mới, cập nhật, xóa…Với số lượng client lớn và cùng một thời điểm phải xử lý rất nhiều yêu cầu, thì Server phải đảm bảo được tính ổn định, xử lý nhanh, mạnh mẽ…Dưới đây là những công nghệ sẽ được sử dụng để đáp ứng những nhu cầu trên.

* + 1. ASP.NET.
       1. Tổng quan về ASP.NET MVC

Mẫu kiến trúc Model – View – Controller được sử dụng nhằm chi ứng dụng thành ba thành phần chính: model, view và controller. Nền tảng ASP.NET MVC giúp cho chúng ta có thể tạo được các ứng dụng web áp dụng mô hình MVC thay vì tạo ứng dụng theo mẫu ASP.NET Web Forsm. Nền tảng ASP.NET MVC có đặc điểm nổi bật là nhẹ (lighweigt), dễ kiểm thử phần giao diện (so với ứng dụng Web Forms), tích hợp các tính năng có sẵn của ASP.NET. Nền tảng ASP.NET MVC được định nghĩa trong namespace System.Web.Mvc và là một phần của name space System.Web.

MVC là một mẫu thiết kế (design pattern) chuẩn mà nhiều lập trình viên đã quen thuộc. Một số loại ứng dụng web sẽ thích hợp với kiến trúc MVC. Một số khác vẫn thích hợp với ASP.NET Web Forms và cơ chế postbacks. Đôi khi có những ứng dụng kết hợp cả hai kiến trúc trên.

Nền tảng MVC bao gồm các thành phần dưới đây:



**Models**: Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL. Ví dụ như, một đối tượng Product (sản phẩm) sẽ lấy dữ liệu từ CSDL, thao tác trên dữ liệu và sẽ cập nhật dữ liệu trở lại vào bảng Products ở SQL Server.

Trong các ứng dụng nhỏ, model thường là chỉ là một khái niệm nhằm phân biệt hơn là được cài đặt thực thụ, ví dụ, nếu ứng dụng chỉ đọc dữ liệu từ CSDL và gởi chúng đến view, ứng dụng không cần phải có tầng model và các lớp lien quan. Trong trường hợp này, dữ liệu được lấy như là một đối tượng model (hơn là tầng model).

**Views**: Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model. Ví dụ như, view dùng để cập nhật bảng Products sẽ hiển thị các hộp văn bản, drop-down list, và các check box dựa trên trạng thái hiện tại của một đối tượng Product.

**Controllers**: Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng. Ví dụ, controller sẽ quản lý các dữ liệu người dùng gởi lên (query-string values) và gởi các giá trị đó đến model, model sẽ lấy dữ liệu từ CSDL nhờ vào các giá trị này.

Mẫu MVC giúp tạo được các ứng dụng mà chúng phân tách rạch ròi các khía cạnh của ứng dụng (logic về nhập liệu, logic xử lý tác vụ và logic về giao diện). Mẫu MVC chỉ ra mỗi loại logic kể trên nên được thiết lập ở đâu trên ứng dụng. Logic giao diện (UI logic) thuộc về views. Logic nhập liệu (input logic) thuộc về controller. Và logic tác vụ (Business logic – là logic xử lý thông tin, mục đích chính của ứng dụng) thuộc về model. Sự phân chia này giúp giảm bớt được sự phức tạp của ứng dụng và chỉ tập trung vào mỗi khía cạnh cần được cài đặt ở mỗi thời điểm. Ví dụ như chỉ cần tập trung vào giao diện (views) mà không phải quan tâm đến logic xử lý thông tin của ứng dụng.

Sự phân tách rạch ròi ba thành phần của ứng dụng MVC còn giúp cho việc lập trình diễn ra song song. Ví dụ như một lập trình viên làm việc với view, lập trình viên thứ hai lo cài đặt logic của controller và lập trình viên thứ ba có thể tập trung vào logic tác vụ của model tại cùng một thời điểm…

* + - 1. Các tính năng của nền tảng ASP.NET MVC.
* Tách bạch các tác vụ của ứng dụng (logic nhập liệu, business logic, và logic giao diện), dễ dàng kiểm thử và mặc định áp dụng hướng phát triển TDD. Tất cả các tính năng chính của mô hình MVC được cài đặt dựa trên interface và được kiểm thử bằng cách sử dụng các đối tượng mocks, mock object là các đối tượng mô phỏng các tính năng của những đối tượng thực sự trong ứng dụng. Có thể kiểm thử unit-test cho ứng dụng mà không cần chạy controller trong tiến trình ASP.NET, và điều đó giúp unit test được áp dụng nhanh chóng và tiện dụng. Có thể sử dụng bất kỳ nền tảng unit-testing nào tương thích với nền tảng .NET.
* MVC là một nền tảng khả mở rộng (extensible) & khả nhúng (pluggable). Các thành phần của ASP.NET MVC được thiết kể để chúng có thể được thay thế một cách dễ dàng hoặc dễ dàng tùy chỉnh. Có thể nhúng thêm view engine, cơ chế định tuyến cho URL, cách kết xuất tham số của action-method và các thành phần khác. ASP.NET MVC cũng hỗ trợ việc sử dụng Dependency Injection (DI) và Inversion of Control (IoC). DI cho phép gắn các đối tượng vào một lớp cho lớp đó sử dụng thay vì buộc lớp đó phải tự mình khởi tạo các đối tượng. IoC quy định rằng, nếu một đối tượng yêu cầu một đối tượng khác, đối tượng đầu sẽ lấy đối tượng thứ hai từ một nguồn bên ngoài, ví dụ như từ tập tin cấu hình. Và nhờ vậy, việc sử dụng DI và IoC sẽ giúp kiểm thử dễ dàng hơn.
* ASP.NET MVC có thành phần ánh xạ URL mạnh mẽ cho phép xây dựng những ứng dụng có các địa chỉ URL xúc tích và dễ tìm kiếm. Các địa chỉ URL không cần phải có phần mở rộng của tên tập tin và được thiết kế để hỗ trợ các mẫu định dạng tên phù hợp với việc tối ưu hóa tìm kiếm (URL) và phù hợp với lập địa chỉ theo kiểu REST.
* Hỗ trợ sử dụng đặc tả (các thẻ) của các trang ASP.NET(.aspx), điều khiển người dùng (.ascx) và trang master page (.marter). Có thể sử dụng các tính năng có sẵn của ASP.NET như là sử dụng lồng các trang master page, sử dụng in-line expression (<%= %>), sử dụng server controls, mẫu, data-binding, địa phương hóa (localization) và hơn thế nữa.
* Hỗ trợ các tính năng có sẵn của ASP.NET như cơ chế xác thực người dùng, quản lý thành viên, quyền, output caching và data caching, seession và profile, quản lý tình trạng ứng dụng, hệ thống cấu hình…
* ASP.NET MVC 3 còn bổ sung một view engine mới là Razor View Engine cho phép thiết lập các view nhanh chóng, dễ dàng và tốn ít công sức hơn so với việc sử dụng Web Forms view engine.
  + 1. SQL Server.
       1. Giới thiệu về SQL Server.

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS)) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server….

* + - 1. Lịch sử ra đời của SQL Server và các phiên bản

Phiên bản đầu tiên của Microsoft SQL Server ra đời đầu tiên vào năm 1989 cho các hệ điều hành chạy 16 bít với SQL Server phiên bản 1.0 và tiếp tục phát triển cho tới ngày nay.

SQL Server của Microsoft được thị trường chấp nhận rộng rãi kể từ version 6.5. Sau đó Microsoft đã cải tiến và hầu như viết lại một engine mới cho SQL Server 7.0. Cho nên có thể nói từ version 6.5 lên version 7.0 là một bước nhảy vọt. Có một số đặc tính của SQL Server 7.0 không tương thích với version 6.5. Trong khi đó từ Version 7.0 lên version 8.0 (SQL Server 2000) thì những cải tiến chủ yếu là mở rộng các tính năng về web và làm cho SQL Server 2000 đáng tin cậy hơn.

Một điểm đặc biệt đáng lưu ý ở phiên bản 2000 là Multiple-Instance. Tức là có thể cài đặt phiên bản 2000 chung với các phiên bản trước mà không cần phải gỡ chúng. Nghĩa là có thể chạy song song version 6.5 hoặc 7.0 với phiên bản 2000 trên cùng một máy (điều này không thể xảy ra với các phiên bản trước đây). Khi đó phiên bản cũ trên máy là Default Instance còn phiên bản 2000 mới vừa cài sẽ là Named Instance.

Từ tháng 10 năm 2016, các phiên bản sau được Microsoft hỗ trợ:

SQL Server 2008 R2

SQL Server 2012

SQL Server 2014

SQL Server 2016

Phiên bản hiện tại là Microsoft SQL Server 2016, xuất bản vào ngày 1/6/2016.

SQL Server 2016 chỉ hỗ trợ cho các bộ vi xử lý 64 bít.

* + - 1. Các thành phần cơ bản trong SQL Server

SQL Server được cấu tạo bởi nhiều thành phần như Database Engine, Reporting Services, Notification Services, Integration Services, Full Text Search Service…. Các thành phần này khi phối hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc lưu trữ và phân tích dữ liệu một cách dễ dàng.

Các thành phần chính SQL Server:

Database Engine– Cái lõi của SQL Server:

Ðây là một engine có khả năng chứa data ở các quy mô khác nhau dưới dạng table và support tất cả các kiểu kết nối (data connection) thông dụng của Microsoft như

ActiveX Data Objects (ADO), OLE DB, and Open Database Connectivity (ODBC).

Ngoài ra nó còn có khả năng tự điều chỉnh (tune up) ví dụ như sử dụng thêm các tài nguyên (resource) của máy khi cần và trả lại tài nguyên cho hệ điều hành khi một user log off.

Replication– Cơ chế tạo bản sao (Replica):

Giả sử có một database dùng để chứa dữ liệu được các ứng dụng thường xuyên cập nhật. Một ngày đẹp trời muốn có một cái database giống y hệt như thế trên một server khác để chạy báo cáo (report database) (cách làm này thường dùng để tránh ảnh hưởng đến performance của server chính). Vấn đề là report server của cũng cần phải được cập nhật thường xuyên để đảm bảo tính chính xác của các báo cáo. Không thể dùng cơ chế back up and restore trong trường hợp này. Thế thì phải làm sao? Lúc đó cơ chế replication của SQL Server sẽ được sử dụng để bảo đảm cho dữ liệu ở 2 database được đồng bộ (synchronized)

Integration Services (DTS) – Integration Services

Là một tập hợp các công cụ đồ họa và các đối tượng lập trình cho việc di chuyển, sao chép và chuyển đổi dữ liệu.

Nếu làm việc trong một công ty lớn trong đó data được chứa trong nhiều nơi khác nhau và ở các dạng khác nhau cụ thể như chứa trong Oracle, DB2 (của IBM), SQL Server, Microsoft Access…. chắc chắn sẽ có nhu cầu di chuyển data giữa các server này (migrate hay transfer) và không chỉ di chuyển còn muốn định dạng (format) nó trước khi lưu vào database khác, khi đó sẽ thấy DTS giúp giải quyết công việc trên dễ dàng.

Analysis Services– Một dịch vụ phân tích dữ liệu rất hay của Microsoft

Dữ liệu (Data) chứa trong database sẽ chẳng có ý nghĩa gì nhiều nếu như không thể lấy được những thông tin (Information) bổ ích từ đó. Do đó Microsoft cung cấp cho một công cụ rất mạnh giúp cho việc phân tích dữ liệu trở nên dễ dàng và hiệu quả bằng cách dùng khái niệm hình khối nhiều chiều (multi-dimension cubes) và kỹ thuật “khai phá dữ liệu” (data mining)

Notification Services

Dịch vụ thông báo Notification Services là nền tảng cho sự phát triển và triển khai các ứng dụng tạo và gửi thông báo. Notification Services có thể gửi thông báo đến hàng ngàn người đăng ký sử dụng nhiều loại thiết bị khác nhau.

Reporting Services

Reporting Services bao gồm các thành phần server và client cho việc tạo, quản lý và triển khai các báo cáo. Reporting Services cũng là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

Full Text Search Service

Dịch vụ SQL Server Full Text Search là một dịch vụ đặc biệt cho đánh chỉ mục và truy vấn cho dữ liệu văn bản không cấu trúc được lưu trữ trong các CSDL SQL Server. Đánh chỉ mục với Full Text Search có thể được tạo trên bất kỳ cột dựa trên dữ liệu văn bản. Nó sẽ rất hiệu quả cho việc tìm các sử dụng toán tử LIKE trong SQL với trường hợp tìm văn bản.

Service Broker

Được sử dụng bên trong mỗi Instance, là môi trường lập trình cho việc các ứng dụng nhảy qua các Instance. Service Broker giao tiếp qua giao thức TCP/IP và cho phép các component khác nhau có thể được đồng bộ cùng nhau theo hướng trao đổi các message. Service Broker chạy như một phần của bộ máy cơ sở dữ liệu, cung cấp một nền tảng truyền message tin cậy và theo hàng đợi cho các ứng dụng SQL Server.

* + - 1. Định nghĩa cơ bản

1. **Định nghĩa cơ sở dữ liệu, bảng, cột.**

CSDL là tên của CSDL chúng ta muốn dùng, có thể bao gồm 1 hay nhiều bảng.

Bảng là 1 bảng giá trị nằm trong CSLD, có thể bao gồm nhiều cột.

Cột là 1 giá trị nằm trong bảng, dùng để lưu trữ các trường dữ liệu, có thể có hoặc không có thuộc tính.

1. **Định nghĩa một số thuật ngữ.**

NULL: giá trị cho phép rỗng.

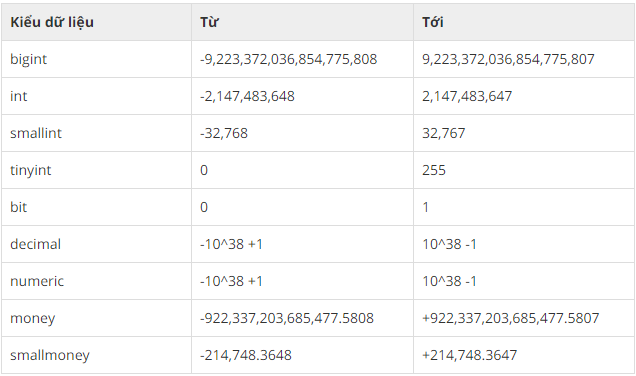
IDENTITY: Cho phép giá trị tự động tăng.

UNSIGNED: Phải là số nguyên dương.

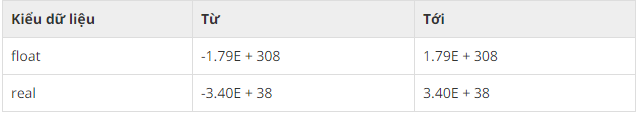
PRIMARY KEY: Khóa chính trong bảng.

1. **Các kiểu dữ liệu trong MySQL.**

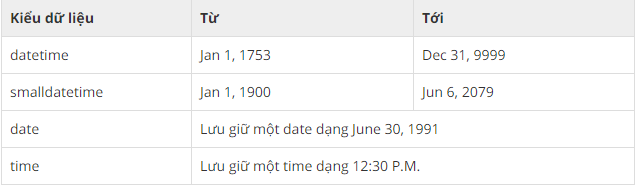
* Kiểu dữ liệu Exact Numeric( số chính xác, không sai số):



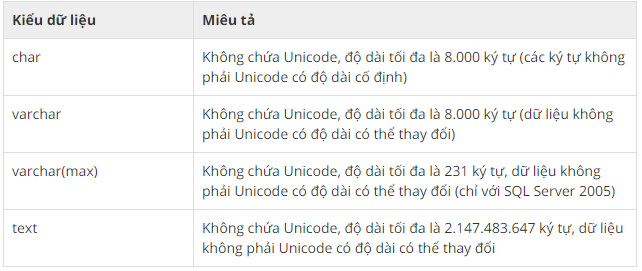
* Kiểu dữ liệu Approximate Numeric:



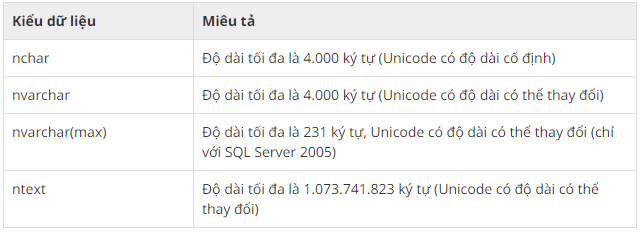
* Kiểu dữ liệu date time:



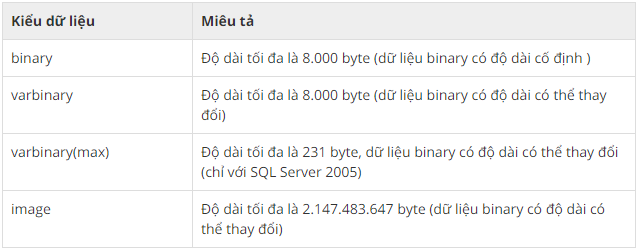
* Kiểu dữ liệu Character String (Kiểu chuỗi)



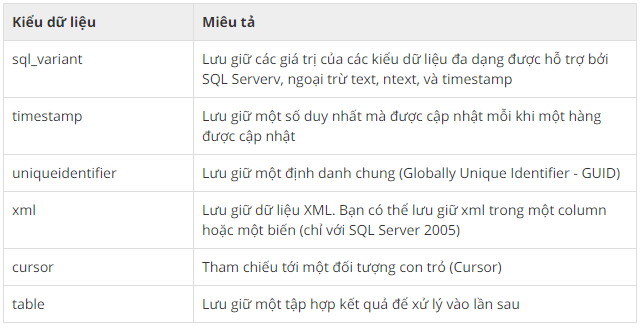
* Kiểu dữ liệu Unicode Character String (kiểu chuỗi có chứa Unicode)



* Kiểu dữ liệu Binary



* Các kiểu dữ liệu khác



* + - 1. Những cú pháp cơ bản.

Các lệnh SQL chuẩn để tương tác với Relational Database là CREATE, SELECT, INSERT, UPDATE, DELETE và DROP. Các lệnh này có thể được phân loại thành các nhóm dựa trên bản chất của chúng.

* DDL (Data Definition Language) – Ngôn ngữ định nghĩa dữ liệu
* Lệnh CREATE: Tạo một bảng, một View của bảng, hoặc đối tượng khác trong Database.Cú pháp cơ bản của lệnh CREATE TABLE trong SQL như sau:



CREATE TABLE là một từ khóa nói cho Database System những gì muốn làm. Trong trường hợp này, muốn tạo một bảng mới. Tên hoặc định danh duy nhất cho bảng đó theo sau lệnh CREATE TABLE.

Sau đó, trong cặp dấu ngoặc đơn, liệt kê phần định nghĩa cho mỗi cột trong bảng và từng kiểu dữ liệu cho nó. theo dõi ví dụ sau để hiểu hơn về khái niệm này.

Ví dụ:Ví dụ sau sẽ tạo một bảng SINHVIEN với ID là khóa chính (primary key) và ràng buộc là NOT NULL chỉ rằng các trường này không thể là NULL trong khi tạo các bản ghi trong bảng này.



* Lệnh ALTER: Sửa đổi một đối tượng Database đang tồn tại, ví dụ như một bảng.

Cú pháp cơ bản của lệnh ALTER TABLE để thêm một cột mới trong một bảng đang tồn tại là như sau:

ALTER TABLE ten\_bang ADD ten\_cot kieu\_du\_lieu;

Cú pháp cơ bản của lệnh ALTER TABLE để XÓA CỘT trong một bảng đang tồn tại là như sau:

ALTER TABLE ten\_bang DROP COLUMN ten\_cot;

Cú pháp cơ bản của lệnh ALTER TABLE để thay đổi KIỂU DỮ LIỆU của một cột trong một bảng là như sau:

ALTER TABLE ten\_bang MODIFY COLUMN ten\_cot kieu\_du\_lieu;

Cú pháp cơ bản của lệnh ALTER TABLE để THÊM một ràng buộc NOT NULL cho một cột trong một bảng là như sau:

ALTER TABLE ten\_bang MODIFY ten\_cot kieu\_du\_lieu NOT NULL;

Cú pháp cơ bản của lệnh ALTER TABLE để THÊM ràng buộc UNIQUE cho một bảng là như sau:

ALTER TABLE ten\_bang

ADD CONSTRAINT MyUniqueConstraint UNIQUE(cot1, cot2...);

Cú pháp cơ bản của lệnh ALTER TABLE để THÊM ràng buộc CHECK cho một bảng là như sau:

ALTER TABLE ten\_bang

ADD CONSTRAINT MyUniqueConstraint CHECK (CONDITION);

Cú pháp cơ bản của lệnh ALTER TABLE để THÊM ràng buộc PRIMARY KEY cho một bảng là như sau:

ALTER TABLE ten\_bang

ADD CONSTRAINT MyPrimaryKey PRIMARY KEY (cot1, cot2...);

Cú pháp cơ bản của lệnh ALTER TABLE để XÓA RÀNG BUỘC cho một bảng là như sau:

ALTER TABLE ten\_bang

DROP CONSTRAINT MyUniqueConstraint;

* + Lệnh: Xóa toàn bộ một bảng, một View của bảng hoặc đối tượng khác trong một Database.

DELETE FROM ten\_bang

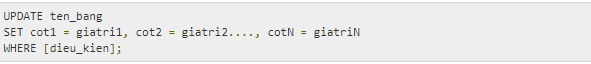
* **DML (Data Manipulation Language) – Ngôn ngữ thao tác dữ liệu**
  + Lệnh SELECT: Lấy các bản ghi cụ thể từ một hoặc nhiều bảng. Cú pháp cơ bản của lệnh SELECT trong SQL như sau:

SELECT cot1, cot2, cot3, cot4 FROM ten\_bang

* + Lệnh INSERT: Tạo một bản ghi. Cú pháp cơ bản của lệnh INSERT trong SQL như sau:



* + Lệnh UPDATE: Sửa đổi các bản ghi.Cú pháp cơ bản của lệnh UPDATE trong SQL như sau:



* + Lệnh DELETE: Xóa các bản ghi.Cú pháp cơ bản của lệnh DELETE trong SQL như sau:



* + Lệnh DROP DATABASE: Xóa database, cú pháp cơ bản của lệnh DROP DATABASE trong SQL là như sau:

DROP DATABASE ten\_database

* DCL (Data Control Language) – Ngôn ngữ điều khiển dữ liệu
  + Lệnh GRANT: Trao một quyền tới người dùng.
  + Lệnh REVOKE: Thu hồi quyền đã trao cho người dùng.
  1. Phía Client.
     1. AngularJS 2.0.
        1. AngularJS là gì?

Đầu tiên, AngularJS là một framework có cấu trúc cho các web động, mã nguồn mở phát triển dựa trên Javascript. Được phát triển lần đầu năm 2009 bởi Misko Hevery và Adam Abrons, hiện tại được duy trì và phát triển bởi Google. Tính đến tháng 5/2016, AngularJS đã có phiên bản thứ 2. Căn bản AngularJS cho phép chúng ta sử dụng HTML như là ngôn ngữ mẫu cho phép mở rộng cú pháp của HTML để diễn đạt các thành phần ứng dụng một cách rõ ràng và súc tích, thường được sử dụng để xây dựng project Single Page Application (SPA – là một trang web hay một ứng dụng web, mà tất cả những thao tác xử lý của trang web đều diễn ra trên một trang duy nhất).

Angular 2 là 1 framework phát triển trên nền JavaScript, kế thừa các đặc điểm của AngularJS và phát triển một phương thức tiếp cận việc xây dựng ứng dụng hoàn toàn mới, phương pháp hướng Component.

Tại sao phải sử dụng Angular 2?

Angular 2 là đơn giản hơn so với Angular 1 và khái niệm của nó làm cho nó dễ dàng hơn để hiểu. Có thể cập nhật các bộ dữ liệu lớn với bộ nhớ tối thiểu. Nó sẽ tăng tốc độ tải ban đầu thông qua cơ chế rendering trên server

* + - 1. Đặc tính

Mobile First

Đây là một hướng phát triển khá là táo bạo của Angular Team khi họ hướng đến bài toán thiết kế cho ứng dụng di dộng, giải quyết các vấn đề về thời gian tải, hiệu năng và các thách thức lớn trên nền tảng di dộng trước khi quay lại với máy tính và biến nó trở thành một bài toán đơn giản.

Hướng đến mobile , Angular Team thiết kế các thư viện và các routing đặc biệt cho mobile giúp lập trình viên có thể xây dựng các ứng dụng di dộng một cách “lean”-nghĩa là nhỏ gọn , tối ưu nên người sử dụng có lưu lượng băng thông thấp có thể sử dụng mà không cần chờ đợi. Các tính năng Mobile first gồm các first-class hỗ trợ cho sự kiện cử chỉ cảm ứng, điều chỉnh hiệu suất và sử dụng bộ nhớ thấp trên nền tảng Mobile mà vẫn cung cấp đầy đủ các thành phần cho việc thiết kế giao diện người dùng với hỗ trợ , đáp ứng tối đa thiết bị.

Future ready

Angular 2 được viết trong TypeScript - một superset của ECMAScript 6 (ES6) với việc bổ sung các tính năng bao gồm các type và annotation. TypeScript cho phép Angular được hưởng lợi từ các sản phẩm tốt nhất của JavaScript, trong khi duy trì clean (sạch sẽ) và easy-to-read (dễ đọc) mã nguồn.

ECMAScript 6 là gì?

Javascript, - ngôn ngữ lập trình web mà chúng ta vẫn sử dụng còn có một tên gọi khác là ECMAScript? ECMAScript hiện nay không phải là phiên bản đầu tiên. Dưới đây là các mốc thời gian của ECMAScript

ECMAScript 1: (ES1) ra đời năm 1997

ECMAScript 2: (ES2) được giới thiệu năm 1998

ECMAScript 3: (ES3) được giới thiệu năm 1999

ECMAScript 4: (ES4) được giới thiệu năm 2000

jQuery ra đời năm 2006

NodeJS ra đời năm 2009

ECMAScript 5: (ES5) ra đời năm 2011

ECMAScript 6: (ES6) tháng 6 năm 2015

ES6 cung cấp một loạt các API rất hữu dụng với cấu trúc Array. Các xử lý cũng linh hoạt lên rất nhiều, trong đó nổi bật nhất phải kể đến Array Comprehension.

Đơn giản là ES6 là những cải tiến của ngôn ngữ JavaScript giúp chúng khoa học và ngắn gọn hơn, đưa chúng lên thành một ngôn ngữ lập trình chính thống chứ không còn là một ngôn ngữ hỗ trợ và tiêu biểu là sự phát triển của Node.js trong thời gian gần đây.

Phát triển linh hoạt

Ngoài việc cung cấp hỗ trợ đầy đủ cho ES6 và TypeScript, Angular 2 hoạt động tốt như nhau với ES5, Dart, CoffeeScript, và các ngôn ngữ khác để biên dịch JavaScript, Angular 2 có thể hoạt động với nhiều ngôn ngữ

Speed & Performance

Thay đổi phát hiện nay giúp angular 2 nhanh hơn 5 lần, nhờ tunings cho công cụ JavaScript hiện đại. Nếu sử dụng các cấu trúc dữ liệu mà hàm ý đảm bảo sự thay đổi (như immutables và observables), AngularJS 2 có thể sử dụng những bảo đảm để tăng tốc độ ứng dụng.

Angular JS ứng dụng được xây dựng xung quanh liên kết dữ liệu giữa DOM và đối tượng của JS. Tốc độ của Angular JS ứng được định hướng chủ yếu theo cơ chế phát hiện sự thay đổi cơ bản mà Angular Team sử dụng. Chúng tôi đã nói chuyện rất lâu về cách Angular Team hy vọng sẽ thực hiện điều này nhanh hơn bằng cách sử dụng Object.observe () khi nó trở nên có sẵn trong Chrome M35. Object.observe () là một API cấp thấp cho phép thêm một người biết lắng nghe để được thông báo khi một đối tượng JavaScript thay đổi trạng thái.

Đơn giản và ý nghĩa

Hãy rõ ràng ý định của bằng cách sử dụng tự nhiên với cú pháp easy-to-write. Giảm độ phức tạp cho nhóm người sử dụng mới, cấu trúc phong phú các mẫu có thể được đọc và dễ hiểu trong nháy mắt.

Comprehensive Rounting

Thiết kế tinh vi views: đường dẫn URL đồ các thành phần ứng dụng, và sử dụng các tính năng tiên tiến như các tuyến đường lồng nhau và anh chị em. Angular 2 hỗ trợ chuyển hướng card chồng, những chuyển động, và lười biếng tải cho người dùng di động. Nếu đã sử dụng định tuyến từ một phiên bản trước của Angular, có thể dễ dàng di chuyển đến góc 2 định tuyến.

Chúng tôi đang tham gia một cái nhìn thận trọng ở những trường hợp sử dụng biết và triển khai thực hiện các router khác trên nhiều mô hình ứng dụng vì vậy chúng tôi có thể cung cấp một router chưa mở rộng đơn giản mà nên phù hợp với tập rộng nhất của các ứng dụng.

Một vài trường hợp mà Angular Team đặc biệt quan tâm hỗ trợ bao gồm:

Định tuyến dựa trên trạng thái

Tích hợp với xác thực và ủy quyền

Tùy chọn bảo quản nhà nước của một số quan điểm. Đặc biệt cần thiết cho điện thoại di động!

Animations

Chạm trực tiếp vào cấp thấp hình ảnh động hỗ trợ trên môi trường di động và máy tính để bàn với easy-to-use sự kiện của Angular. có thể sử dụng CSS, JavaScript, và các Web Animations API để xử lý một cách thông minh cho thay đổi hình ảnh động để đáp ứng với sự kiện người dùng. Có kế hoạch lưu chuyển hoạt hình phức tạp bằng cách sắp xếp các hành vi của toàn bộ một trang web trên một timeline.

Người dùng đã quen với cảm ứng nhất định mô hình sử dụng biết. Ví dụ như di chuyển qua một danh sách bằng cách sử dụng ngón tay của mình, khoanh tròn qua các hình ảnh trong một băng chuyền, loại bỏ danh mục bằng cách swiping chúng đi. Tuy nhiên:

Triển khai hiện tại của băng chuyền, di chuyển vô hạn,... không chung một phổ biến và bằng cách này có rất nhiều dự phòng và phương pháp tiếp cận khác nhau.

Triển khai hiện nay chủ yếu là không cung cấp một tùy chọn để sử dụng các sự kiện cuộn bản địa, như các trình duyệt cũ và cũng có một số các trình duyệt hiện tại không hỗ trợ họ tốt. Tuy nhiên, bằng cách này họ ngăn chặn hiệu suất tối ưu trên các thiết bị mới.

Hierarchical Dependency Injection

Angular 2 mạnh mẽ, nhưng simple-to-use tiêm phụ thuộc, cho phép duy trì các ứng dụng module mà không cần viết mã lặp lại và tẻ nhạt. Dependency injection giúp viết bài kiểm tra bằng cách làm cho nó dễ dàng để tiêm kiểm tra 2 lần.

Hỗ trợ cho các thành phần web

Và với sự ra đời của ECMAScript 6:(ES6) Điều này có nghĩa là trình duyệt hỗ trợ cho những thứ như các mô-đun, các class, lambdas, generators, vv Các tính năng cơ bản chuyển đổi các kinh nghiệm lập trình JavaScript . Và thuật ngữ Web Components thường đề cập đến một bộ sưu tập của bốn thông số kỹ thuật W3C có liên quan:

* Custom Elements - Cho phép mở rộng của HTML thông qua thẻ tùy chỉnh.
* HTML Imports- Cho phép đóng gói các nguồn tài nguyên khác nhau (HTML, CSS, JS, vv).
* Template Element - Cho phép sự include của inert HTML trong một tài liệu.
* Shadow DOM - Cho phép đóng gói các DOM và CSS.

Bằng cách kết hợp bốn khả năng phát triển web có thể tạo ra các thành phần khai báo (Custom Elements) được đóng gói đầy đủ (Shadow DOM). Các thành phần này có thể mô tả quan điểm riêng của họ (Template Element) và có thể được dễ dàng đóng gói để phân phối cho các nhà phát triển khác (Imports HTML).

webcomponent Những thứ như scope,controller,modules hay dirctives đã không còn tồn tại trong Angular 2 mà đơn giản chỉ còn Components và Template syntax. Điều này làm trang web trở nên đơn giản và rõ ràng hơn nhiều. Một sự thay đổi lớn từ Angular 1.x sang Angular 2+

Internationalization (I18N) & Accessibility

Sử dụng định dạng tin nhắn ICU quen thuộc trong cú pháp nội suy góc ({{}}), bao gồm cả pluralization và giới luật. Tự động hóa khai thác tin nhắn, giả nội địa hóa, và cập nhật bản dịch. Tạo các ứng dụng tĩnh cho mỗi miền địa phương. Thúc đẩy khả năng tiếp cận dễ dàng thông qua trình đọc màn hình và các thiết bị trợ giúp bằng cách tự động tạo ra các thuộc tính ARIA thích hợp

TỔNG KẾT

Angular 2 là nhanh hơn và dễ dàng hơn so với Angular 1.

Nó hỗ trợ tất các phiên bản của trình duyệt và cũng hỗ trợ các trình duyệt cũ bao gồm IE9 + và Android 4.1 trở lên.

Nó là một framework đa nền tảng.

Angular 2 chủ yếu tập trung vào các ứng dụng di động.

Cấu trúc mã là rất đơn giản hơn so với phiên bản trước.

Typing sử dụng TypeScript, Dart, JavaScript

* Ưu điểm

Nếu một ứng dụng được tải nặng, sau đó Angular 2 giữ nó hoàn toàn giao diện người dùng.

Nó sử dụng máy chủ rendering cho view nhanh trên điện thoại di động.

Nó hoạt động tốt với ECMAScript và các ngôn ngữ khác để biên dịch JavaScript.

Nó sử dụng Dependency Injection chạy các ứng dụng mà không cần viết mã quá dài.

Tất cả mọi thứ sẽ là cách tiếp cận dựa trên Component

AngularJS cung cấp khả năng tạo ra các Single Page Application một cách rất rõ ràng và dễ dàng để duy trì.

AngularJS cung cấp khả năng Data binding tới HTML do đó giúp người dùng cảm giác linh hoạt, thân thiện.

AngularJS code dễ dàng khi unit test.

AngularJS cung cấp khả năng tái sử dụng các component (thành phần).

Với AngularJS, lập trình viên sẽ viết ít code hơn, với nhiều chức năng hơn.

Với AngularJS, view là thành phần trong trang HTML thuần, trong khi controller được viết bởi JavaScript với quá trình xử lý nghiệp vụ.

Và trên tất cả, ứng dụng AngularJS có thể chạy trên hết các trình duyệt web, trên các nền tảng Android và iOS.

* Nhược điểm

Mặc dù AngularJS có thể kể đến rất nhiều các ưu điểm, nhưng đến thời điểm này, nó vẫn có một số điểm yếu sau:

Không an toàn: Là một Javascript framework, ứng dụng được viết bởi

AngularJS không an toàn. Phải có các tính năng bảo mật và xác thực phía server sẽ giúp ứng dụng trở nên an toàn hơn.

Nếu người sử dụng ứng dụng của vô hiệu hóa JavaScript thì họ chỉ nhìn được trang cơ bản, không thấy gì thêm.

* + 1. Framework Ionic 2.
       1. Giới thiêu.

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 hiển thị các đoạn mã HTML.

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.

Có thể coi Ionic như là một bộ khung front-end giúp kiểm soát hình ảnh và trải nghiệm trên ứng dụng. 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.

Ionic được xây dựng trên Cordova stack. Không thể tạo các ứng dụng di động với một mình Ionic bởi vì nó chỉ xử lý một phần giao diện người dùng. Nó cần phải làm việc với Angular, cái mà xử lý các logic của ứng dụng, và Cordova, framework ứng dụng đa nền tảng cho phép biên dịch ứng dụng của thành một tập tin có thể cài đặt và chạy nó bên trong web view của thiết bị di động.

Ứng dụng được xây dựng với Cordova và Ionic có thể chạy trên cả thiết bị Android và iOS, cũng có thể cài đặt các plugin của Cordova để cung cấp các chức năng gốc như truy cập vào camera và làm việc với các thiết bị Bluetooth Low Energy

* + - 1. Những điểm mới trong ionic 2.
* **Trình duyệt hỗ trợ.**

Ionic 1 được xây dựng chỉ cho các ứng dụng lai. Nhưng Ionic 2 còn được xây dựng để hỗ trợ các ứng dụng web hiện đại và các ứng dụng Electron. Điều này có nghĩa rằng, bây giờ có thể xây dựng không chỉ các ứng dụng Ionic chạy bên trong môi trường Cordova mà còn các ứng dụng web hiện đại, sử dụng tính năng web hiện đại để cung cấp một trải nghiệm giống như ứng dụng cho người dùng.

Cũng có thể chọn Electron, một nền tảng cho việc xây dựng các ứng dụng Desktop đa nền tảng bằng cách sử dụng HTML, CSS, và JavaScript. Electron khá giống với Cordova, nhưng cho các hệ điều hành desktop như Windows, Ubuntu hay macOS.

* **Open-source.**

Mục đích của Ionic framework không chỉ là xây dựng một bộ khung phát triển ứng dụng di động đa nền tảng mà còn muốn xây dựng một nền tảng để chia sẻ các kiến thức cho các nhà phát triển, tạo ra một cách để đưa ra những design pattern tốt nhất để xây dựng các ứng dụng di động. Vì vậy, Ionic là một framework mở hoàn toàn cho phép tất cả mọi người có thể chia sẻ những kiến thức, kinh nghiệm để xây dựng những ứng dụng di động tuyệt vời trên nền tảng HTML5.

Ionic sẽ đặt nền tảng cho ứng dụng của bạn dựa trên HTML5, CSS và Javascript. Thay vì mỗi lập trình viên phải tự tìm tòi những khuyết điểm của việc phát triển ứng dụng di động trên HTML5 thì Ionic sẽ cho bạn CSS cơ bản và tạo cho bạn 1 kiến trúc tốt để phát triển phía trên nó.

* **Angular 2 và TypeScript**

Ionic 2 giờ đây sử dụng Angular 2 cho template và logic của ứng dụng. Điều này có nghĩa rằng các nhà phát triển sẽ phải tìm hiểu cú pháp mới của Angular 2 trước khi họ có thể tạo ra các ứng dụng Ionic 2 hiệu quả. Tuy nhiên, đừng lo lắng, bởi vì các khái niệm vẫn giống như Angular 1. Ngoài ra còn có các nguồn tài nguyên như ngMigrate sẽ giúp chuyển đổi Angular 1 sang Angular 2.

Ngoài Angular 2, Ionic 2 cũng sử dụng TypeScript. Đối với những người không quen với nó, TypeScript là một superset của JavaScript. Điều này có nghĩa rằng vẫn có thể sử dụng cú pháp thuần JavaScript để viết các ứng dụng. Nếu muốn sử dụng các tính năng đi kèm với TypeScript, chẳng hạn như cú pháp ES6 và ES7, static-typing, và tính năng hoàn tác code thông minh, thì có thể sử dụng cú pháp cụ thể của TypeScript. Có các plugin mà có thể cài đặt trên trình soạn thảo văn bản hoặc IDE ưa thích để tận dụng những lợi ích của của các tính năng hoàn tác code cao cấp của TypeScript.

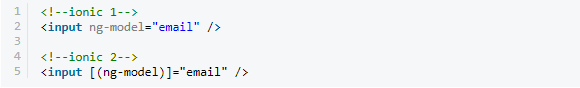
**Cú pháp**

Như tôi đã đề cập các cú pháp template trong Ionic 2 đã thay đổi đáng kể, phần lớn là do quá trình chuyển đổi để sử dụng Angular 2. thậm chí có thể thấy rằng các cú pháp mới là đơn giản và súc tích hơn. Dưới đây là một vài ví dụ cú pháp của Ionic 1 và Ionic 2 nằm cạnh nhau:

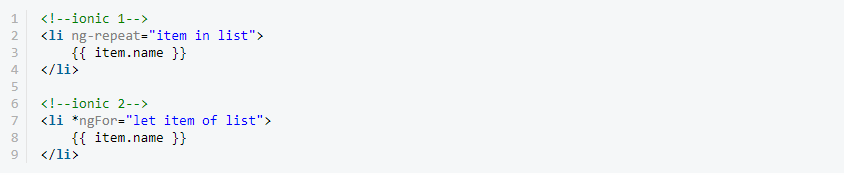
Lắng nghe các Sự kiện:



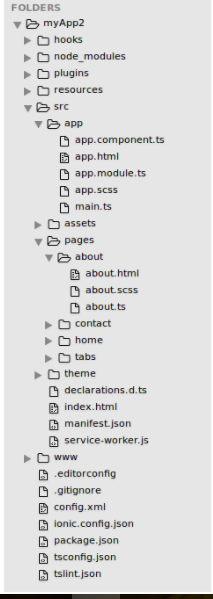
Sử dụng một Model:

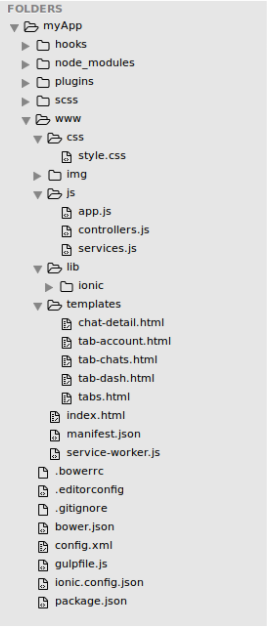


Lặp qua một mảng và hiển thị mỗi phần tử:



Nếu so sánh cấu trúc thư mục của một dự án Ionic 1 và một dự án Ionic 2 sẽ thấy rằng hầu hết các thư mục mà thường thấy trong một dự án Ionic 1 vẫn còn Ionic 2. Điều này là bởi vì nền tảng cơ bản vẫn chưa thực sự thay đổi—Ionic 2 vẫn sử dụng Cordova. Những thứ duy nhất đã thay đổi là các bộ phận để làm việc với các tập tin mã nguồn. Dưới đây là ảnh chụp màn hình cấu trúc thư mục của ứng dụng ionic 1 và 2

Ionic 2

Ionic 1

Nếu nhìn kỹ hơn sẽ nhận thấy rằng bây giờ thư mục src. Đó là nơi chứa tất cả các tập tin mã nguồn của dự án, và mỗi khi thay đổi một tập tin trong thư mục đó, các tập tin đã thay đổi được biên dịch và sao chép lên vào thư mục www/build.Để trở nên thoải mái khi làm việc với một dự án Ionic 2, cần phải tự mình làm quen với cấu trúc thư mục. Đối với những người mới bắt đầu, cần biết mỗi thư mục được sử dụng cho mục đích gì để từ đó biết nơi đặt các tập tin nguồn và nơi để tìm các tập tin cần thiết.

**node\_modules:** đây là nơi các phụ thuộc của Ionic 2 được lưu trữ. Phần lớn, không cần phải chạm vào thư mục này trừ khi có vấn đề với một trong các phụ thuộc và phải cài đặt lại nó.

**Platforms**: đây là nơi đặt code cho nền tảng cụ thể và nơi trình cài đặt ứng dụng được đặt khi build ứng dụng để chạy trên thiết bị hoặc trên emulator. Điều đó có nghĩa là tất cả các tập tin trong thư mục www và plugins của sẽ được sao chép vào đây mỗi khi build ứng dụng của mình.

**Plugins:** đây là nơi mà các plugin được lưu trữ, tất nhiên rồi—bao gồm cả các plugin Ionic mặc định và nhiều plugin khác mà cài đặt.

**Resources:** đây là nơi sẽ đặt các tài nguyên của ứng dụng chẳng hạn như biểu tượng và màn hình khởi động (hay còn gọi splash screen).

**Src**: đây là nơi sẽ viết phần lớn code. Tất cả các tập tin template, stylesheet và TypeScript tạo nên ứng dụng của được lưu trữ ở đây.

**www**: đây là nơi các tập tin đã được biên dịch của được lưu. Các tập tin ở đây phục vụ cho bản xem trước ứng dụng.

**hooks**: đây là nơi mà các script hook của Cordova được lưu trữ. Chúng được sử dụng nếu có các script tuỳ biến mà muốn thực thi tại một thời điểm trong vòng đời phát triển (ví dụ, khi một nền tảng hoặc plugin được thêm vào).

Cấu trúc thư mục cũng được tổ chức tốt hơn. Nếu kiểm tra thư mục src/pages, có thể thấy rằng mỗi trang có thư mục riêng của mình, và bên trong mỗi trang là các tập tin HTML, CSS và JavaScript cho trang cụ thể đó.

Trước đó, trong Ionic 1 đã chỉ định một thư mục rỗng và tự do cấu trúc dự án theo ý muốn cá nhân. Nhưng điều này đi kèm với nhược điểm không buộc người dùng làm các thứ theo cách tốt nhất. Người dùng có thể lười biếng và gắn bó với một cấu trúc gộp tất cả các tập tin với nhau, nó có thể làm cho các thứ trở nên khó khăn cho các nhóm lớn hơn làm việc trên các ứng dụng phức tạp.

**Theme**

Không giống như phiên bản trước đó của Ionic, chỉ có một giao diện duy nhất cho tất cả các nền tảng, Ionic 2 bây giờ hỗ trợ ba chế độ: Material Design, iOS và Windows. Bây giờ, Ionic tệp với giao diện của nền tảng mà nó được triển khai. Vì vậy nếu ứng dụng của được cài đặt trên android, ví dụ là vậy, nó sẽ sử dụng một phong cách và hành vi tương tự như ứng dụng Android gốc.

Có một sự hỗ trợ cho thiết kế theme trong Ionic, mặc dù tại thời điểm viết bài này, nó chỉ đi kèm với theme Light mặc định. Nếu muốn tinh chỉnh theme, có thể chỉnh sửa tập tin src/theme/variables.scss.

* **Công cụ**

Ionic 2 cũng đi kèm với các công cụ mới mà sẽ tạo ra một hứng thú để tạo ra các ứng dụng di động. Tôi sẽ cho thấy một số công cụ trong phần này.

**Generator:** Ionic 2 bây giờ cung cấp một generator cho phép tạo các trang và dịch vụ cho ứng dụng của:



Điều này sẽ tạo ra các tập tin trong thư mục app/pages:



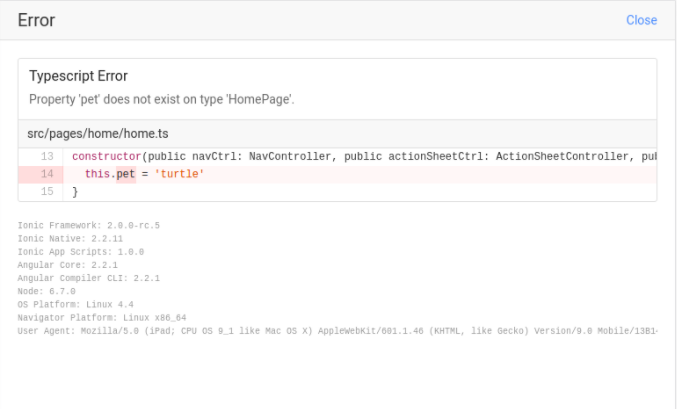
Mỗi tập tin cũng có một số code mẫu trong đó:



Chúng cũng phục vụ như là một hướng dẫn cho các nhà phát triển mới, do đó họ biết được cách làm tốt nhất cho cấu trúc code của họ. Đây là code TypeScript được tạo ra để xử lý logic cho trang ở trên:



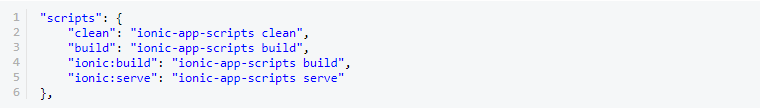
**Báo Lỗi**: Ionic 2 bây giờ đi kèm với một công cụ báo lỗi cho front-end. Điều này có nghĩa rằng bất kỳ thời điểm nào có một lỗi với code hiện tại thì Ionic sẽ mở một cửa sổ ngay bên phải trong preview của chính app. Điều này thực sự giúp dễ dàng cho các nhà phát triển tìm hiểu về các lỗi khi chúng xảy ra trong ứng dụng.



**Ionic App Script**: Các Ionic App Script là một bộ sưu tập các build script cho các dự án Ionic. Trước đó, Ionic sử dụng Gulp để xử lý quá trình build.

Ionic 2 đi kèm với một vài script này giúp nó dễ dàng hơn để hoàn thành các tác vụ phát triển thông thường. Điều này bao gồm những thứ như chuyển đổi code TypeScript thành ES5, phục vụ cho ứng dụng để thử nghiệm trong trình duyệt, hoặc chạy nó trên một thiết bị cụ thể.

Có thể tìm thấy các script mặc định trong tập tin package.json của dự án:



* **Các Thành phần mới**

Các thành phần là các khối xây dựng giao diện người dùng trong Ionic. Ví dụ như button, card, list, và các input. Rất nhiều thành phần mới đã được thêm vào Ionic 2

* **Slides**Nếu muốn ứng dụng có một hướng dẫn cho người dùng lần đầu, thì thành phần Slides giúp dễ dàng tạo ra một hướng dẫn như vậy. Thành phần này cho phép tạo ra các layout dựa trên trang mà người dùng có thể vuốt qua để đọc tất cả về ứng dụng.
* **Action Sheet**

Các Action Sheet là các menu trượt lên từ phía dưới màn hình. Một Action Sheet được thể hiện trên lớp trên cùng của màn hình, do đó phải bỏ qua nó bằng cách bấm vào khoảng trống hoặc chọn một tùy chọn từ menu. Điều này thường được sử dụng cho các xác nhận chẳng hạn như khi xoá bỏ một tập tin trên thiết bị iOS...

* **Các Tính năng và Cải tiến mới**

Ionic 2 cũng được đóng gói cùng với các tính năng và cải tiến mới. Những điều này chủ yếu là do nó chuyển sang Angular 2 và TypeScript.

* **Web Animations API**

Một lợi ích khi chuyển sang Angular 2 là hệ thống hiệu ứng động mới của Angular, được xây dựng dựa trên Web Animations API. Lưu ý rằng Web Animations API không được hỗ trợ trong tất cả các trình duyệt. Nhược điểm duy nhất của cách này là nó sẽ làm cho kích thước cài đặt lớn hơn. Một tùy chọn khác là sử dụng một polyfill.

* **Hiệu năng**

Các ứng dụng được tạo ra với Ionic 2 thì linh hoạt hơn so với những cái được tạo ra với Ionic 1. Đây là lý do tại sao:

* **Angular 2:** Hiệu suất thao tác DOM và JavaScript đã được cải thiện rất nhiều trong Angular 2. Có thể kiểm tra bảng này nếu muốn tìm hiểu về các chi tiết cụ thể. Một lợi ích khác đi kèm với Angular 2 đó là tiền biên dịch—các template được biên dịch trước bằng cách sử dụng một công cụ build thay vì được biên dịch như ứng dụng chạy trong trình duyệt. Điều này làm cho ứng dụng khởi tạo nhanh hơn vì không cần phải biên dịch các template đồng thời.
* **Cuộn trang Gốc:** Ionic không còn sử dụng JavaScript để cuộn trang nữa. Thay vào đó, nó bây giờ sử dụng cuộn trang gốc được hỗ trợ trong WebView. Nó bây giờ cũng được kích hoạt trên tất cả các nền tảng (trái ngược với nó chỉ được hỗ trợ trên Android trong Ionic 1). Ngoài cuộn trang gốc, còn có Cuộn trang Ảo cho phép di chuyển trên một danh sách rất lớn của các phần tử rất ít ảnh hưởng đến hiệu suất. Hai thay đổi này làm cho việc cuộn trang trở nên mượt mà hơn.
* **Web Worker:** Web Worker cho phép chạy các script trong nền, cô lập khỏi các tiến trình chạy các trang web. Ionic 2 cài đặt các web worker thông qua thành phần ion-img của chúng. Sử dụng thành phần này thay vì thành phần img tiêu chuẩn cho phép phân bổ các yêu cầu HTTP để lấy hình ảnh đến một Web Worker. Điều này làm cho việc tải hình ảnh trở nên mạnh mẽ hơn, đặc biệt là bên trong các danh sách lớn. Thành phần ion-img cũng xử lý kỹ thuật lazy loading, nó sẽ chỉ yêu cầu và kết xuất hình ảnh khi nó hiện hữu trong viewport của người dùng.
* **Ionic Native**

Ionic Native tương đương với ngCordova cho Ionic 2. Cả hai đều hoạt động như các wrapper cho các plugin Cordova để cài đặt các tính năng gốc (ví dụ như Camera, vị trí). Thậm chí có thể sử dụng Inonic Native trong ứng dụng Ionic 1 của nếu muốn. Sự khác biệt chính đó là Ionic Native cho phép viết code sử dụng các tính năng của ES6 và cú pháp của TypeScript. Điều này giúp dễ dàng hơn để làm việc với Ionic 2 vì nó đã mặc định sử dụng TypeScript.

* 1. Tổng quan về mã vạch
     1. Mã vạch là gì?

Mã vạch (Barcode) theo định nghĩa là phương pháp lưu trữ và truyền tải thông tin bằng một lọai ký hiệu gọi là ký mã vạch (Barcode symbology). Ký mã vạch hay gọi tắt cũng là mã vạch, là 1 ký hiệu tổ hợp các khoảng trắng và vạch thẳng để biểu diễn các mẫu tự, ký hiệu và các con số. Sự thay đổi trong độ rộng của vạch và khoảng trắng biểu diễn thông tin số hay chữ số dưới dạng mà máy có thể đọc được.

Mã số mã vạch được thu nhận bằng một máy quét mã vạch, là một máy thu nhận hình ảnh của mã vạch in trên các bề mặt và chuyển thông tin chứa trong mã vạch đến máy tính hay các thiết bị cần thông tin này. Nó thường có một nguồn sáng kèm theo thấu kính, để hội tụ ánh sáng lên mã vạch, rồi thu ánh sáng phản xạ về một cảm quang chuyển hóa tín hiệu ánh sáng thành tín hiệu điện. Ngoài ra, nhiều máy quét mã vạch còn có thêm mạch điện tử xử lý tín hiệu thu được từ cảm quang để chuyển thành tín hiệu phù hợp cho kết nối với máy tính.

* + 1. Các loại mã vạch

Có thể nói mã vạch cũng giống như một đạo quân các ký hiệu quen thuộc, chúng xuất hiện ở khắp mọi nơi, mọi chỗ, trên hầu hết các sản phẩm lưu hành hợp pháp trên thị trường. Ai cũng đều thấy chúng nhưng ít ai hiểu được nhiều về chúng. Nhưng vì nghĩ mã vạch là “vô thưởng vô phạt” nên cũng chẳng ai quan tâm đến chúng cả. Khi được hỏi về mã vạch, đa số người ta chỉ biết mã vạch là … mã vạch. Nó mã hóa một con số gì đó mà người ta không hiểu. Nói như vậy nghiễm nhiên mã vạch chỉ có một lọai duy nhất là … mã vạch và nó được sử dụng để lưu trữ 1 con số gì đó như giá tiền chẳng hạn.

Thực ra mã vạch gồm nhiều chủng lọai khác nhau. Tùy theo dung lượng thông tin, dạng thức thông tin được mã hóa cũng như mục đích sử dụng mà người ta chia ra làm rất nhiều lọai, trong đó các dạng thông dụng trên thị trường mà ta thấy gồm UPC, EAN, Code 39, Interleaved 2of 5, Codabar và Code 128. Ngoài ra, trong 1 số loại mã vạch người ta còn phát triển làm nhiều Version khác nhau, có mục đích sử dụng khác nhau, thí dụ UPC có các version là UPC-A, UPC-B, UPC-C, UPC-D và UPC-E; EAN có các version EAN-8, EAN-13, EAN-14, Code 128 gồm Code 128 Auto, Code 128-A, Code 128-B, Code 128-C.

1. UPC (Universal Product Code)

UPC là 1 lọai ký hiệu mã hóa số được ngành công nghiệp thực phẩm ứng dụng vào năm 1973. Ngành công nghiệp thực phẩm đã phát triển hệ thống này nhằm gán mã số không trùng lặp cho từng sản phẩm. Người ta sử dụng UPC như “giấy phép bằng số” cho các sản phẩm riêng lẻ.

UPC gồm có 2 phần: phần mã vạch mà máy có thể đọc được và phần số mà con người có thể đọc được.

Số của UPC gồm 12 ký số, không bao gồm ký tự. Đó là các mã số dùng để nhận diện mỗi một sản phẩm tiêu dùng riêng biệt.

Ký số thứ 1: Ở đây là số 0, gọi là ký số hệ thống số (number system digit) hoặc còn gọi là “Family code”. Nó nằm trong phạm vi của 7 con số định rõ ý nghĩa và chủng lọai của sản phẩm như sau:

\* 5 - Coupons: Phiếu lĩnh hàng hóa

\* 4 - Dành cho người bán lẽ sử dụng

\* 3 - Thuốc và các mặt hàng có liên quan đến y tế.

\* 2 - Các món hàng nặng tự nhiên như thịt và nông sản.

\* 0, 6, 7 - Gán cho tất cả các mặt hàng khác như là một phần nhận diện của nhà sản xuất.

Năm ký số thứ 2: ám chỉ mã người bán (Vendor Code), mã doanh nghiệp hay mã của nhà sản xuất (Manufacturer code). Ở Hoa kỳ, mã này được cấp bởi hiệp hội UCC (The Uniform Code Council) và mã được cấp cho người bán hoặc nhà sản xuất là độc nhất. Như vậy khi hàng hóa lưu thông trên thị trường bằng mã UPC thì chỉ cần biết được 5 ký số này là có thể biết được xuất xứ của hàng hóa.

Năm ký số kế tiếp: Dành cho người bán gán cho sản phẩm của họ. Người bán tự tạo ra 5 ký số này theo ý riêng của mình để mã hóa cho sản phẩm.

Ký số cuối cùng: Là ký số kiểm tra, xác nhận tính chính xác của toàn bộ số UPC

UPC được phát triển thành nhiều phiên bản (version) như UPC-A, UPC-B, UPC-C, UPC-D và UPC-E trong đó UPC-A được coi như phiên bản chuẩn của UPC, các phiên bản còn lại được phát triển theo những yêu cầu đặc biệt của ngành công nghiệp.

1. EAN (European Article Number)

EAN là bước phát triển kế tiếp của UPC. Về cách mã hóa nó cũng giống hệt như UPC nhưng về dung lượng nó gồm 13 ký số trong đó 2 hoặc 3 ký số đầu tiên là ký số “mốc”, dùng để biểu thị cho nước xuất xứ. Các ký số này chính là “mã quốc gia” của sản phẩm được cấp bởi Tổ chức EAN quốc tế (EAN International Organization)

EAN này được gọi là EAN-13 để phân biệt với phiên bản EAN-8 sau này gồm 8 ký số.



Theo ký hiệu EAN-13 như hình vẽ phía trên, có thể phân chia như sau:

\* 893 - Mã quốc gia Việt Nam

\* 604381125 - 9 ký số này được phân chia làm 2 cụm: cụm mã nhà sản xuất có thể 4, 5 hoặc 6 ký số tùy theo được cấp, cụm còn lại là mã mặt hàng.

\* 4 - Mã kiểm tra tính chính xác của toàn bộ số EAN.

EAN có một biến thể khác của nó là JAN (Japaneses Artical Numbering), thực chất là EAN của người Nhật với mã quốc gia là 49.

Vì EAN phát triển với mã quốc gia nên nó được sử dụng trên những sản phẩm lưu thông trên toàn cầu. Các tiêu chuẩn của EAN do Tổ chức EAN quốc tế quản lý. Ở Việt Nam, các doanh nghiệp muốn sử dụng được mã EAN trên sản phẩm của mình, phải là thành viên của Tổ chức Mã Số Mã Vạch Việt Nam, gọi tắt là EAN Việt Nam, để được cấp mã số doanh nghiệp.

1. Code 39

UPC và EAN dù là 2 lọai mã vạch có tính chất chuyên nghiệp và quốc tế nhưng khuyết điểm của nó là dung lượng có giới hạn và chỉ mã hóa được số, không mã hóa được chữ.

Code 39 được phát triển sau UPC và EAN là ký hiệu chữ và số thông dụng nhất. Nó không có chiều dài cố định như UPC và EAN do đó có thể lưu trữ nhiều lượng thông tin hơn bên trong nó. Do tính linh họat như vậy, Code 39 được ưa chuộng rộng rãi trong bán lẻ và sản xuất. Bộ ký tự này bao gồm tất cả các chữ hoa, các ký số từ 0 đến 9 và 7 ký tự đặc biệt khác.

Nhiều tổ chức đã chọn một dạng thức Code 39 để làm chuẩn công nghiệp của mình trong đó đáng chú ý là Bộ Quốc Phòng Mỹ đã lấy Code 39 làm bộ mã gọi là LOGMARS.

* + 1. Mã vạch được ứng dụng để mã hóa những gì?

Có thể mã hoá đủ loại thông tin thành mã vạch. Ví dụ:

1. Số hiệu linh kiện (Part Numbers)

2. Số nhận diện người bán, nhận diện nhà sản xuất, doanh nghiệp (Vendor ID Numbers, ManufactureID Numbers)

3. Số hiệu Pallet (Pallet Numbers)

4. Nơi trữ hàng hoá

5. Ngày nhận

6. Tên hay số hiệu khách hàng

7. Giá cả món hàng

8. Số hiệu lô hàng và số xê ri

9. Số hiệu đơn đặt gia công

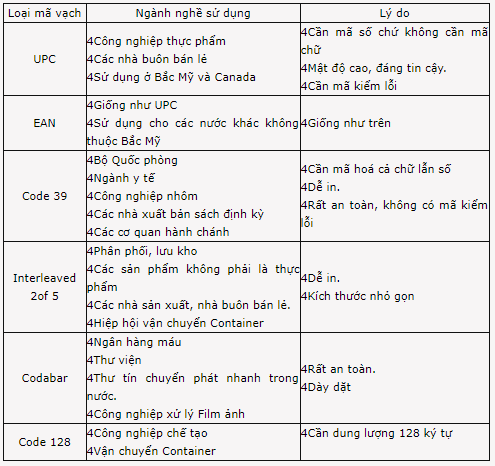
10. Mã nhận diện tài sản

11. Số hiệu đơn đặt mua hàng

v.v….

Một khi công ty đã xác định xong thông tin cần mã hoá, bước tiếp theo là xác định loại mã vạch thích hợp, kích thước của mã vạch, công nghệ mã hoá thông tin và công nghệ in thích hợp nhất.

Bảng dưới đây mô tả công dụng mã hoá của các loại mã vạch thông dụng:



* + 1. Tìm hiểu về mã vạch của hàng hóa

Để tạo thuận lợi và nâng cao năng suất, hiệu quả trong bán hàng và quản lý kho người ta thường in trên hàng hoá một loại mã hiệu đặc biệt gọi là mã số mã vạch của hàng hoá. Mã số mã vạch của hàng hoá bao gồm hai phần: mã số của hàng hoá và mã vạch là phần thể hiện mã số bằng vạch để cho máy đọc.

Mã số của hàng hoá là một dãy con số dùng để phân định hàng hoá, áp dụng trong quá trình luân chuyển hàng hoá từ người sản xuất, qua bán buôn, lưu kho, phân phối, bán lẻ tới người tiêu dùng. Nếu thẻ căn cước giúp ta phân biệt người này với người khác thì mã số hàng hoá là “thẻ căn cước” của hàng hoá, giúp ta phân biệt được nhanh chóng và chính xác các loại hàng hoá khác nhau.

2.5.4.a. Mã số của hàng hoá có các tính chất sau:

- Nó là con số duy nhất đặc trưng cho hàng hoá. Mỗi loại hàng hoá được nhận diện bởi một dãy số và mỗi dãy số chỉ tương ứng với một loại hàng hoá.

- Bản thân mã số chỉ là một dãy số đại diện cho hàng hoá, không liên quan đến đặc điểm của hàng hoá. Nó không phải là số phân loại hay chất lượng của hàng hoá, trên mã số cũng không có giá cả của hàng hoá.

Hiện nay, trong thương mại trên toàn thế giới chủ yếu áp dụng hai hệ thống mã số hàng hoá sau:

- Hệ thống UPC (Universal Product Code) là hệ thống thuộc quyền quản lý của Hội đồng mã thống nhất Mỹ UCC (Uniform Code Council, Inc.), được sử dụng từ năm 1970 và hiện vẫn đang sử dụng ở Mỹ và Canada.

- Hệ thống EAN (European Article Number) được thiết lập bởi các sáng lập viên là 12 nước châu Âu với tên gọi ban đầu là Hội EAN (European Article Numbering Association), được sử dụng từ năm 1974 ở châu Âu và sau đó phát triển nhanh chóng, được áp dụng ở hầu hết các nước trên thế giới. Chính vì lý do này nên từ năm 1977, EAN trở thành một tổ chức quốc tế với tên gọi EAN quốc tế (EAN International)

Trong hệ thống mã số EAN cho sản phẩm bán lẻ có hai loại, một loại sử dụng 13 con số (EAN-13) và loại kia sử dụng 8 con số (EAN-8)

Mã số EAN-13 gồm 13 con số có cấu tạo như sau: từ trái sang phải

+ Mã quốc gia: hai hoặc ba con số đầu

+ Mã doanh nghiệp: có thể gồm từ bốn, năm hoặc sáu con số

+ Mã mặt hàng: có thể là năm, bốn, hoặc ba con số tùy thuộc vào mã doanh nghiệp

+ Số cuối cùng là số kiểm tra

Để đảm bảo tính thống nhất và tính đơn nhất của mã số, mã quốc gia phải do tổ chức mã số vật phẩm quốc tế cấp cho các quốc gia là thành viên của tổ chức này. Mã số quốc gia của Việt Nam là 893. Danh mục mã số quốc gia của các nước trong phụ lục kèm theo.

Mã doanh nghiệp do tổ chức mã số vật phẩm quốc gia cấp cho các nhà sản xuất là thành viên của họ. ở Việt Nam, mã doanh nghiệp do EAN-VN cấp cho các doanh nghiệp thành viên của mình.

Mã mặt hàng do nhà sản xuất quy định cho hàng hoá của mình. Nhà sản xuất phải đảm bảo mỗi mặt hàng chỉ có một mã số, không được có bất kỳ sự nhầm lẫn nào.

2.5.4.b. Mã số hàng hóa (MSHH)nó được cấu tạo như thế nào?

Đến nay, trong giao dịch thương mại tồn tại 2 hệ thống cơ bản về MSHH:

Một là, hệ thống MSHH được sử dụng tại thị trường Hoa Kỳ và Canada. Đó là hệ thống UPC (Universal Product Code), được lưu hành từ thập kỷ 70 của thế kỷ XX cho đến nay.

Hai là, hệ thống MSHH được sử dụng rộng rãi ở các thị trường còn lại của thế giới, đặc biệt là châu Âu, châu Á… trong đó phổ biến là hệ thống EAN (European Article Number). Trong hệ thống MSHH EAN có 2 loại ký hiệu con số: Loại EAN-13 và EAN-8.

+ Cấu trúc của EAN-13:

Mã số EAN-13 là 1 dãy số gồm 13 chữ số nguyên (từ số 0 đến số 9), trong dãy số chia làm 4 nhóm, mỗi nhóm có ý nghĩa như sau (xem hình 1):

Nhóm 1: Từ trái sang phải, ba chữ số đầu là mã số về quốc gia (vùng lãnh thổ)

Nhóm 2: Tiếp theo gồm bốn chữ số là mã số về doanh nghiệp.

Nhóm 3: Tiếp theo gồm năm chữ số là mã số về hàng hóa.

Nhóm 4: Số cuối cùng (bên phải) là số về kiểm tra

Ví dụ theo quy ước trên, số kiểm tra (C) có ý nghĩa về quản lý đối với việc đăng nhập, đăng xuất của các loại sản phẩm hàng hóa của từng loại doanh nghiệp.

Vậy xác định như thế nào?

Ví dụ: Mã số 8 9 3 3 4 8 1 0 0 1 0 6 - C:

Bước 1 - Xác định nguồn gốc hàng hóa: 893 là MSHH của quốc gia Việt Nam; 3481 là MS doanh nghiệp thuộc quốc gia Việt Nam; 00106 là MSHH của doanh nghiệp.

Bước 2 - Xác định C.

Cộng tổng giá trị của các số ở thứ tự lẻ bắt đầu được tính từ phải sang trái của dãy MS (trừ số C), ta có: 6 + 1 + 0 + 8 + 3 + 9 = 27 (1)

Nhân tổng của (1) với 3, ta có: 27 x 3 = 81 (2)

P -893 - nhóm 1.

M -4602 - nhóm 2.

I -00107 - nhóm 3.

C -8 - nhóm 4.

Cộng tổng giá trị của các số ở thứ tự chẵn còn lại, ta có:

0 + 0 + 1 + 4 + 3 + 8 = 16 (3)

Cộng giá trị (2) với (3), ta có: 81 + 16 = 97 (4)

Lấy giá trị của (4) làm tròn theo bội số của 10 (tức là 100) sát nhất với giá trị của (4) trừ đi giá trị của (4) ta có: 100 - 97 = 3. Như vậy C = 3.

Trong trường hợp này mã số EAN - VN 13 có MSHH đầy đủ là:

893 3481 00106 3

+ Cấu trúc của EAN - 8:

Về bản chất tương tự như EAN-13 chỉ khác là EAN-8 gồm 8 chữ số nguyên, tuỳ theo sắp xếp và lựa chọn các chữ số từ số 0 đến số 9 được chia làm 3 nhóm:

Mã số quốc gia: Gồm 3 chữ số đầu tiên (bên trái)

Mã số hàng hóa: Gồm 4 chữ số tiếp theo.

Mã số kiểm tra: Gồm 1 chữ số đứng cuối cùng. Nhận dạng số C cũng được tính từ 7 số đứng trước nó và cách tính cũng tương tự như EAN-13.

Cần lưu ý rằng, việc sử dụng EAN-13 hay EAN-8 là do Tổ chức EAN thế giới phân định. Sau khi EAN Việt Nam được cấp MS, các doanh nghiệp của Việt Nam muốn sử dụng mã số EAN-VN thì phải có đơn đệ trình là thành viên EAN-VN, sau đó đăng ký xin EAN-VN cấp cho MS cho đồng loại hàng hóa. Việc cấp đăng ký MS cho sản phẩm hàng hóa của doanh nghiệp Việt Nam do các tổ chức EAN-VN có thẩm quyền cấp và được EAN thế giới công nhận, được lưu trữ trong ngân hàng dữ liệu EAN thế giới.

1. KHẢO SÁT YÊU CẦU NGƯỜI DÙNG.
   1. Mục đích hệ thống

Ứng dụng di động quét mã vạch giúp người sử dụng truy xuất thông tin, nguồn gốc xuất xứ của sản phẩm, qua đó tránh được tình trạng mua phải hàng giả, hàng nhái, hàng không rõ nguồn gốc xuất xứ.

* 1. Xác định vai trò người dùng

Hệ thống chia làm ba nhóm người dùng chính

* Khách truy cập vào app
* Thành viên (người đã đăng ký tài khoản)
* Quản trị hệ thống

Định nghĩa từng nhóm người dùng

* Khách truy cập vào app: nhóm người này sử dụng app mà không cần đăng ký tài khoản, họ có thể sử dụng phần mềm để quét mã vạch của sản phẩm, tìm kiếm sản phẩm theo tên hoặc theo mã vạch của sản phẩm
* Thành viên (người đã đăng ký tài khoản): trong quá trình sử dụng app, nhóm người này đã đăng ký tài khoản và đăng nhập vào hệ thống của phần mềm nên nhóm người này có đầy đủ chức năng thao tác với app như của nhóm khách truy cập vào app đồng thời nhóm này còn có thêm khả năng bình luận và đánh giá về sản phẩm.
* Quản trị hệ thống: là người có khả năng quản lý, thao tác với dữ liệu về sản phẩm và thành viên của hệ thống, nhóm này có khả năng quản lý tài khoản của thành viên, quản lý sản phẩm, tìm kiếm và xem thống kê về số lượt truy cập app, số lượt bình luận và đánh giá cho từng sản phẩm dựa vào điều kiện tìm kiếm (tên sản phẩm, danh mục thời gian)
  1. Yêu cầu chức năng cho từng vai trò người dùng
* Khách truy cập vào app:

+ Đăng ký tài khoản

+ Đăng nhập vào hệ thống

+ Quét mã vạch của sản phẩm

+ Xem thông tin chi tiết sản phẩm

+ Tìm kiếm sản phẩm

* Thành viên( người đã đăng ký tài khoản với hệ thống)

+ Đánh giá về sản phẩm đang xem

+ Bình luận về sản phẩm đang xem

+ Đặt câu hỏi về sản phẩm đang xem

+ Trả lời câu hỏi về sản phẩm đang xem

+ Đăng xuất tài khoản khỏi hệ thống

+ Đăng nhập vào hệ thống

+ Quét mã vạch của sản phẩm

+ Xem thông tin chi tiết sản phẩm

+ Tìm kiếm sản phẩm

* Quản trị hệ thống

+ Quản lý sản phẩm

* Thêm sản phẩm
* Sửa sản phẩm
* Xóa sản phẩm

+ Quản lý danh mục sản phẩm

* Thêm danh mục
* Sửa danh mục
* Xóa danh mục

+ Quản lý tài khoản

* Thêm tài khoản
* Sửa tài khoản
* Xóa tài khoản

+ Quản lý bình luận, câu hỏi, câu trả lời

* Ẩn bình luận
* Ẩn câu hỏi
* Ẩn câu trả lời

+ Xem thống kê về số lượt truy cập app, số lượt bình luận và đánh giá cho từng sản phẩm dựa vào điều kiện tìm kiếm (tên sản phẩm, danh mục thời gian)

* 1. Mô tả chi tiết chức năng cho từng nhóm người dùng
     1. Đối với khách truy cập vào app

1. Đăng ký tài khoản

|  |  |
| --- | --- |
| Mô tả | Để làm thành viên của hệ thống người dùng phải đăng ký với hệ thống một tài khoản, việc đăng ký tài khoản của hệ thống được thực hiện khi người dùng chọn chức năng đăng ký tài khoản hoặc khi người dùng muốn trả lời hoặc bình luận về sản phẩm mà chưa có tài khoản thì sẽ được chuyển đến màn hình đăng ký đề người dùng có thể đăng ký tài khoản |
| Đầu vào | Tên đại diện, mật khẩu, email, số điện thoại. |
| Xử lý | Hệ thống kiểm tra thông tin người dùng nhập vào:  + Tên đại diện trong khoảng từ 8 -16 ký tự  + Mật khẩu có độ dài từ 8- 24 ký tự  + Email phải đúng định dạng chuẩn của email  + Số điện thoại phải đúng định dạng  Nếu các điều kiện chính xác, hệ thống sẽ xử lý tạo tài khoản trên hệ thống với thông tin tài khoản do người dùng vừa nhập vào |
| Đầu ra | Đưa ra thông báo thành công và yêu cầu người dùng vào mail để kích hoạt tài khoản. |

1. Quét mã vạch của sản phẩm

|  |  |
| --- | --- |
| Mô tả | Người dùng dùng camera đưa lại gần mã vạch của sản phẩm để camera có thể nhận diện được mã vạch, hệ thống sẽ xử lý hình ảnh của mã vạch để trả về một mã số, hệ thống sẽ kiểm tra mã số này xem có phải là mã vạch hay không (chi tiết kiểm tra mã vạch tại mục 2.4.5.b) dùng mã số này để tìm kiếm sản phẩm trong cơ sở dữ liệu rồi trả về thông tin chi tiết của sản phẩm sau đó hiển thị thông tin chi tiết của sản phẩm ấy lên màn hình ứng dụng cho người dùng |
| Đầu vào | Camera của điện thoại. |
| Xử lý | Khi camera nhận diện được mã vạch, hệ thống sẽ xử lý hình ảnh của mã vạch để trả về một mã số, hệ thống sẽ dùng mã số này để tìm kiếm sản phẩm trong cơ sở dữ liệu rồi trả về thông tin chi tiết của sản phẩm |
| Đầu ra | Hiển thị thông tin chi tiết của sản phẩm (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |

1. Xem thông tin chi tiết sản phẩm

|  |  |
| --- | --- |
| Mô tả | Ở màn hình trang chủ sẽ có một danh sách các sản phẩm và mô tả ngắn cho từng sản phẩm, bằng việc nhấn chọn vào sản phẩm hoặc mô tả của sản phẩm ấy, người dùng có thể xem thông tin chi tiết của sản phẩm ấy (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |
| Đầu vào | Nhấn chọn vào sản phẩm cần xem |
| Xử lý | Hệ thống sẽ dựa vào mã của sản phẩm mà người dùng chọn để lấy thông tin chi tiết của sản phẩm ấy trong cơ sở dữ liệu sau đó hiển thị lên màn hình ứng dụng cho người dùng |
| Đầu ra | Hiển thị thông tin chi tiết của sản phẩm (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |

1. Tìm kiếm sản phẩm

|  |  |
| --- | --- |
| Mô tả | Tại màn hình trang chủ sẽ có khả năng tìm kiếm, khi người dùng nhấn vào sẽ cho nhập văn bản, tại đây người dùng có thể nhập tên sản phẩm hoặc mã vạch của sản phẩm, hệ thống sẽ dựa vào nội dung người dùng nhập vào để tìm kiếm trong cơ sở dữ liệu những sản phẩm có tên hoặc mã vạch đúng hoặc gần đúng với từ khóa người dùng nhập vào, sau đó hiển thị danh sách kết quả sản phẩm tìm kiếm được lên màn hình cho người dùng  Người dùng tìm kiếm sản phẩm bằng tên hoặc bằng mã vạch của sản phẩm |
| Đầu vào | Tên hoặc mã vạch của sản phẩm |
| Xử lý | Hệ thống sẽ dựa vào nội dung người dùng nhập vào để tìm kiếm trong cơ sở dữ liệu những sản phẩm có tên hoặc mã vạch đúng hoặc gần đúng với từ khóa người dùng nhập vào, sau đó hiển thị danh sách kết quả sản phẩm tìm kiếm được lên màn hình cho người dùng |
| Đầu ra | Hiển thị danh sách các sản phẩm phù hợp với từ khóa người dùng nhập vào |

* + 1. Đối với thành viên (người đã đăng ký tài khoản)

1. **Bình luận và đánh giá về sản phẩm đang xem**

|  |  |
| --- | --- |
| Mô tả | Tại trang chi tiết của sản phẩm sẽ có 5 ngôi sao, thành viên có thể đánh giá sản phẩm qua việc lựa chọn số ngôi sao trên màn hình hiển thị, thứ tự ngôi sao tăng dần tương đương về độ hài lòng của khách hàng về sản phẩm( rất kém,ém, bình thường, tốt, rất tốt)  Tại màn hình chi tiết của sản phẩm có phần nhập văn bản, tại đây người dùng có thể nhập văn bản, nêu lên ý kiến của mình về sản phẩm sau đó nhấn vào nút bình luận. Tại đây hệ thống sẽ kiểm tra văn bản, khi văn bản hợp lệ k dài quá 256 ký tự thì hệ thống sẽ lưu bình luận của người dùng vào cơ sở dữ liệu, sau đó hiển thị bình luận của thành viên đến màn hình chi tiết sản phẩm. |
| Đầu vào | Bình luận và đánh giá của thành viên |
| Xử lý | Hệ thống sẽ kiểm tra văn bản và đánh giá, khi văn bản hợp lệ và đánh giá không dài quá 256 ký tự và khác 0 thì hệ thống sẽ lưu bình luận và đánh giá của người dùng vào cơ sở dữ liệu, sau đó hiển thị bình luận và đánh giá của thành viên đến màn hình chi tiết sản phẩm. |
| Đầu ra | Hiển thị bình luận và đánh giá của thành viên trên màn hình chi tiết sản phẩm |

1. **Đặt câu hỏi về sản phẩm**

|  |  |
| --- | --- |
| Mô tả | Tại màn hình chi tiết của sản phẩm có phần nhâp văn bản, tại đây người dùng có thể nhập câu hỏi, nêu lên câu hỏi của mình về sản phẩm sau đó nhấn vào nút đặt câu hỏi. Tại đây hệ thống sẽ kiểm tra văn bản, khi văn bản hợp lệ không dài quá 256 ký tự thì hệ thống sẽ thêm câu hỏi của người dùng vào cơ sở dữ liệu, sau đó hiển thị bình luận của thành viên đến màn hình chi tiết sản phẩm. |
| Đầu vào | Câu hỏi của thành viên |
| Xử lý | Hệ thống sẽ kiểm tra văn bản, khi văn bản hợp lệ, không dài quá 256 ký tự thì hệ thống sẽ lưu câu hỏi của người dùng vào cơ sở dữ liệu, sau đó hiển thị câu hỏi đến màn hình chi tiết sản phẩm |
| Đầu ra | Hiển thị câu hỏi của người dùng lên trên trang chi tiết của sản phẩm |

1. **Trả lời câu hỏi về sản phẩm**

|  |  |
| --- | --- |
| Mô tả | Từ màn hình chi tiết của sản phẩm, dưới mỗi câu hỏi sẽ có dòng chữ nhỏ với nội dung là “trả lời”, khi người dùng nhấn vào đó thì sẽ được nhập text để trả lời cho câu hỏi tương ứng, sau khi nhập nội dung của câu trả lời, người dùng nhấn vào nút “post” hệ thống sẽ kiểm tra văn bản, khi văn bản hợp lệ độ dài không vượt quá 255 ký tự thì câu hỏi sẽ được lưu vào trong cơ sở dữ liệu, quá trình trả lời câu hỏi về sản phẩm được hoàn tất, phần chữ nhỏ với nội dung là “trả lời” sẽ được cập nhật “ trả lời(số câu trả lời)”, khi người dùng nhấn vào nhãn đó, nội dung tất cả câu trả lời cho câu hỏi sẽ được hiển thị ra |
| Đầu vào | Câu hỏi của thành viên về sản phẩm |
| Xử lý | hệ thống sẽ kiểm tra văn bản, khi văn bản hợp lệ độ dài không vượt quá 255 ký tự thì câu hỏi sẽ được lưu vào trong cơ sở dữ liệu, quá trình trả lời câu hỏi về sản phẩm được hoàn tất, phần chữ nhỏ với nội dung là “trả lời” sẽ được cập nhật “ trả lời(số câu trả lời)”, khi người dùng nhấn vào nhãn đó, nội dung tất cả câu trả lời cho câu hỏi sẽ được hiển thị ra |
| Đầu ra | Nhãn “ trả lời” sẽ được cập nhật “ trả lời(số câu trả lời)”, khi người dùng nhấn vào nhãn đó, nội dung tất cả câu trả lời cho câu hỏi sẽ được hiển thị ra |

1. Đăng nhập.

|  |  |
| --- | --- |
| Mô tả | Trong quá trình người dùng sử dụng app, mỗi khi muốn bình luận, đánh giá hoặc trả lời câu hỏi, hệ thống sẽ kiểm tra người dùng này đã có tài khoản và đăng nhập vào hệ thống chưa, nếu chưa thì sẽ được chuyển đến trang đăng nhập. Việc đăng nhập cũng có thể thực hiện thông qua cách chọn chức năng đăng nhập thì mà hình đăng nhập cũng sẽ được hiển thị ra. Tại màn hình đăng nhập người dùng sẽ nhập email và mật khẩu của tài khoản, hệ thống sẽ kiểm tra dữ liệu nhập vào ( kiểm tra kiểu dữ liệu email và mật khẩu có độ dài từ 8- 24 ký tự) nếu sai sẽ thông báo lỗi cho người dùng, nếu đúng hệ thống sẽ tìm kiếm tài khoản này có tồn tại trong cơ sở dữ liệu hay không, nếu tồn tại thì thông báo đăng nhập thành công, sai thì hiển thị thông báo tài khoản không tồn tại |
| Đầu vào | Người dùng nhập tên email và mật khẩu để đăng nhập hệ thống. |
| Xử lý | Kiểm tra kiểu dữ liệu email và mật khẩu có độ dài từ 8- 24 ký tự) nếu sai sẽ thông báo lỗi cho người dùng, nếu đúng hệ thống sẽ tìm kiếm tài khoản này có tồn tại trong cơ sở dữ liệu hay không, nếu tồn tại thì thông báo đăng nhập thành công, sai thì hiển thị thông báo tài khoản không tồn tại |
| Đầu ra | Nếu thông tin đăng nhập đúng thì hiển thị thông báo thành công, sai thì hiển thị thông báo lỗi |

1. Quét mã vạch của sản phẩm

|  |  |
| --- | --- |
| Mô tả | Người dùng dùng camera đưa lại gần mã vạch của sản phẩm để camera có thể nhận diện được mã vạch, hệ thống sẽ xử lý hình ảnh của mã vạch để trả về một mã số, hệ thống sẽ dùng mã số này để tìm kiếm sản phẩm trong cơ sở dữ liệu rồi trả về thông tin chi tiết của sản phẩm sau đó hiển thị thông tin chi tiết của sản phẩm ấy lên màn hình ứng dụng cho người dùng |
| Đầu vào | Camera của điện thoại. |
| Xử lý | Khi camera nhận diện được mã vạch, hệ thống sẽ xử lý hình ảnh của mã vạch để trả về một mã số, hệ thống sẽ dùng mã số này để tìm kiếm sản phẩm trong cơ sở dữ liệu rồi trả về thông tin chi tiết của sản phẩm |
| Đầu ra | Hiển thị thông tin chi tiết của sản phẩm (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |

1. Xem thông tin chi tiết sản phẩm

|  |  |
| --- | --- |
| Mô tả | Ở màn hình trang chủ sẽ có một danh sách các sản phẩm và mô tả ngắn cho từng sản phẩm, bằng việc nhấn chọn vào sản phẩm hoặc mô tả của sản phẩm ấy, người dùng có thể xem thông tin chi tiết của sản phẩm ấy (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |
| Đầu vào | Nhấn chọn vào sản phẩm cần xem |
| Xử lý | Hệ thống sẽ dựa vào mã của sản phẩm mà người dùng chọn để lấy thông tin chi tiết của sản phẩm ấy trong cơ sở dữ liệu sau đó hiển thị lên màn hình ứng dụng cho người dùng |
| Đầu ra | Hiển thị thông tin chi tiết của sản phẩm (Hình ảnh sản phẩm, giá sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả của sản phẩm, bình luận, đánh giá về sản phẩm) |

1. Tìm kiếm sản phẩm

|  |  |
| --- | --- |
| Mô tả | Tại màn hình trang chủ sẽ có khả năng tìm kiếm, khi người dùng nhấn vào sẽ cho nhập văn bản, tại đây người dùng có thể nhập tên sản phẩm hoặc mã vạch của sản phẩm, hệ thống sẽ dựa vào nội dung người dùng nhập vào để tìm kiếm trong cơ sở dữ liệu những sản phẩm có tên hoặc mã vạch đúng hoặc gần đúng với từ khóa người dùng nhập vào, sau đó hiển thị danh sách kết quả sản phẩm tìm kiếm được lên màn hình cho người dùng  Người dùng tìm kiếm sản phẩm bằng tên hoặc bằng mã vạch của sản phẩm |
| Đầu vào | Tên hoặc mã vạch của sản phẩm |
| Xử lý | Hệ thống sẽ dựa vào nội dung người dùng nhập vào để tìm kiếm trong cơ sở dữ liệu những sản phẩm có tên hoặc mã vạch đúng hoặc gần đúng với từ khóa người dùng nhập vào, sau đó hiển thị danh sách kết quả sản phẩm tìm kiếm được lên màn hình cho người dùng |
| Đầu ra | Hiển thị danh sách các sản phẩm phù hơp với từ khóa người dùng nhập vào |

* + 1. Đối với quản trị

1. **Quản lý sản phẩm**
   * **Thêm thông tin sản phẩm**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý sản phẩm và chọn chức năng thêm sản phẩm thì màn hình thêm sản phẩm sẽ được hiển thị ra cho người dùng, tại đây quản trị viên sẽ nhập đầy đủ thông tin liên quan đến sản phẩm mà form thiết kế yêu cầu (Hình ảnh sản phẩm, mã sản phẩm giá sản phẩm, danh mục sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả về sản phẩm ). Sau khi nhập xong thông tin người dùng nhấn nút thêm sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ thêm mới sản phẩm vào cơ sở dữ liệu và hiển thị thông báo thành công |
| Đầu vào | Quản trị nhập thông tin cần thiết cho sản phẩm mới: hình ảnh sản phẩm, giá sản phẩm, danh mục sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả về sản phẩm |
| Xử lý | Hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ thêm mới sản phẩm vào cơ sở dữ liệu và hiển thị thông báo thành công. |
| Đầu ra | Hiển thị thông báo thành công |

* + **Sửa thông sản phẩm.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý sản phẩm, danh sách các sản phẩm sẽ được hiển thị, tại mỗi sản phẩm sẽ có chức năng sửa hoặc xóa ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng sửa sản phẩm, màn hình sửa sản phẩm sẽ được hiển thị. Tại đây người dùng có thể sửa nội dung thông tin liên quan đến sản phẩm mà người dùng vừa chọn (Hình ảnh sản phẩm, mã sản phẩm, giá sản phẩm, danh mục sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả về sản phẩm). Sau khi nhập xong thông tin người dùng nhấn nút sửa sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ cập nhật nội dung vào cơ sở dữ liệu và hiển thị thông báo thành công, sau đó chuyển về trang quản lý sản phẩm. |
| Đầu vào | Người dùng chọn sản phẩm cần sửa, nhấn vào chức năng sửa sản phẩm và chỉnh sửa các thông tin cần thiết cho sản phẩm được chọn |
| Xử lý | Sau khi nhập xong thông tin người dùng nhấn nút sửa sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ cập nhật nội dung vào cơ sở dữ liệu |
| Đầu ra | Hiển thị thông báo thành công và hiển thị lại trang quản lý sản phẩm |

* + **Xóa thông sản phẩm.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý sản phẩm, danh sách các sản phẩm sẽ được hiển thị, tại mỗi sản phẩm sẽ có chức năng sửa hoặc xóa hoặc chặn ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng xóa sản phẩm, sẽ có thông báo được hiện lên “Bạn có muốn xóa sản phẩm này hay không ?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị của sản phẩm thông qua mã của sản phẩm được chọn và cập nhật lại danh sách sản phẩm, nếu chọn không thì trở lại trang danh sách sản phẩm |
| Đầu vào | Người dùng chọn sản phẩm cần xóa và chọn chức năng xóa. |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn xóa của người dùng, nếu người dùng tiếp tục hệ thống sẽ lấy mã của sản phẩm đó và thực hiện việc chuyển trạng thái hiển thị của sản phẩm. |
| Đầu ra | Thông báo đã xóa thành công. |

1. Quản lí tài khoản.
   * **Thêm tài khoản.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý tài khoản và chọn chức năng thêm tài khoản thì màn hình thêm tài sẽ được hiển thị ra cho người dùng, tại đây quản trị viên sẽ nhập đầy đủ thông tin liên quan đến tài khoản mà form thiết kế yêu cầu (Tên đại diện trong khoảng từ 8 -16 ký tự, mật khẩu có độ dài từ 8- 24 ký tự, email phải đúng định dạnh chuẩn của email, số điện thoại phải đúng định dạng). Sau khi nhập xong thông tin người dùng nhấn nút thêm tài khoản, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa và đúng yêu cầu chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa chính xác, nếu nhập đầy đủ thì hệ thống sẽ thêm mới tài khoản vào cơ sở dữ liệu và hiển thị thông báo thành công |
| Đầu vào | Quản trị nhập thông tin cần thiết cho tài khoản: Tên đại diện trong khoảng từ 8 -16 ký tự, mật khẩu có độ dài từ 8- 24 ký tự, email phải đúng định dạnh chuẩn của email, số điện thoại phải đúng định dạng |
| Xử lý | Hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa và đúng yêu cầu chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa chính xác, nếu nhập đầy đủ thì hệ thống sẽ thêm mới tài khoản vào cơ sở dữ liệu và hiển thị thông báo thành công. |
| Đầu ra | Hiển thị thông báo thành công. |

* + **Sửa tài khoản.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý tài khoản, danh sách các tài khoản sẽ được hiển thị, tại mỗi tài khoản sẽ có chức năng sửa hoặc xóa, chặn tài khoản ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng sửa tài khoản, màn hình sửa tài khoản sẽ được hiển thị. Tại đây người dùng có thể sửa nội dung thông tin liên quan đến tài khoản mà người dùng vừa chọn (Tên hiển thị, email, số điện thoại, mật khẩu). Sau khi nhập xong thông tin người dùng nhấn nút sửa tài khoản, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ và chính xác chưa (tên đại diện trong khoảng từ 8 -16 ký tự, mật khẩu có độ dài từ 8- 24 ký tự, email phải đúng định dạnh chuẩn của email, số điện thoại phải đúng định dạng), nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ hoặc chưa đúng định dạng, nếu nhập đầy đủ thì hệ thống sẽ cập nhật thông tin tài khoản vào cơ sở dữ liệu và hiển thị thông báo thành công, sau đó chuyển về trang quản lý tài khoản. |
| Đầu vào | Quản trị chọn tài khoản cần sửa và chính sửa các thông tin cần thiết (Tên hiển thị, email, số điện thoại, mật khẩu) |
| Xử lý | Sau khi nhập xong thông tin người dùng nhấn nút sửa tài khoản, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ và chính xác chưa (tên đại diện trong khoảng từ 8 -16 ký tự, mật khẩu có độ dài từ 8- 24 ký tự, email phải đúng định dạnh chuẩn của email, số điện thoại phải đúng định dạng), nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ hoặc chưa đúng định dạng, nếu nhập đầy đủ thì hệ thống sẽ cập nhật thông tin tài khoản vào cơ sở dữ liệu và hiển thị thông báo thành công, sau đó chuyển về trang quản lý tài khoản. |
| Đầu ra | Hiển thị thông báo thành công và cập nhật lại danh sách tài khoản. |

* + **Chặn tài khoản.**

|  |  |
| --- | --- |
| Mô tả | Trong quá trình sử dụng app, có một số tài khoản bình luận hoặc đánh giá những nội dung không liên quan hoặc k phù hợp, ta cần chặn các tài khoản này để tránh việc ảnh hưởng xấu đến ứng dụng. Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý tài khoản, danh sách các tài khoản sẽ được hiển thị, tại mỗi tài khoản sẽ có chức năng sửa hoặc xóa, chặn tài khoản ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng chặn tài khoản sẽ có thông báo được hiện lên “Bạn có muốn chặn tài khoản này hay không?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị của tài khoản thông qua mã của tài khoản được chọn và cập nhật lại danh sách tài khoản, nếu chọn không thì trở lại trang danh sách tài khoản |
| Đầu vào | Quản trị chọn tài khoản muốn chặn và nhấn nút chặn |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn chặn tài khoản của quản trị, nếu quản trị tiếp tục hệ thống sẽ lấy mã của sản phẩm đó và thực hiện việc chuyển trạng thái hiển thị của tài khoản. |
| Đầu ra | Thông báo kết quả thành công và hiển thị lại danh sách tài khoản |

1. Quản lí bình luận, câu hỏi, câu trả lời.
   * Ẩn bình luận

|  |  |
| --- | --- |
| Mô tả | Trong quá trình sử dụng app, có một số tài khoản bình luận hoặc nội dung không liên quan hoặc k phù hợp, ta cần ẩn các bình luận này để tránh việc ảnh hưởng xấu đến ứng dụng. Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý bình luận, câu hỏi, trả lời, và click vào quản lý bình luận danh sách các bình luận sẽ được hiển thị, tại mỗi bình luận sẽ có chức năng ẩn bình luận ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng ẩn bình luận sẽ có thông báo được hiện lên “Bạn có muốn ẩn bình luận này hay không?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị bình luận thông qua mã của bình luận được chọn và cập nhật lại danh sách bình luận, nếu chọn không thì trở lại trang danh sách bình luận. |
| Đầu vào | Quản trị chọn bình luận cần ẩn và ấn nút ẩn bình luận. |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn ẩn bình luận của quản trị, nếu quản trị tiếp tục hệ thống sẽ lấy mã của sản phẩm đó và thực hiện việc chuyển trạng thái hiển thị của bình luận. |
| Đầu ra | Thông báo kết quả thành công và hiển thị lại danh sách bình luận |

* + Ẩn câu hỏi

|  |  |
| --- | --- |
| Mô tả | Trong quá trình sử dụng app, có một số tài khoản đặt câu hỏi hoặc nội dung không liên quan hoặc k phù hợp, ta cần ẩn các câu hỏi này để tránh việc ảnh hưởng xấu đến ứng dụng. Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý bình luận, câu hỏi, trả lời, và click vào quản lý câu hỏi, danh sách các câu hỏi sẽ được hiển thị, tại mỗi câu hỏi sẽ có chức năng ẩn câu hỏi ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng ẩn câu hỏi sẽ có thông báo được hiện lên “Bạn có muốn ẩn câu hỏi này hay không?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị câu hỏi thông qua mã của câu hỏi được chọn và cập nhật lại danh sách câu hỏi, nếu chọn không thì trở lại trang danh sách câu hỏi. |
| Đầu vào | Quản trị chọn câu hỏi cần ẩn và ấn nút ẩn câu hỏi. |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn ẩn câu hỏi của quản trị, nếu quản trị tiếp tục hệ thống sẽ lấy mã của câu hỏi đó và thực hiện việc chuyển trạng thái hiển thị của câu hỏi. |
| Đầu ra | Thông báo kết quả thành công và hiển thị lại danh sách câu hỏi |

* + Ẩn câu trả lời

|  |  |
| --- | --- |
| Mô tả | Trong quá trình sử dụng app, có một số tài khoản đặt câu trả lời hoặc nội dung không liên quan hoặc k phù hợp, ta cần ẩn các câu hỏi này để tránh việc ảnh hưởng xấu đến ứng dụng. Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý bình luận, câu hỏi, trả lời, và click vào quản lý câu trả lời, danh sách các trả lời sẽ được hiển thị, tại mỗi câu trả lời sẽ có chức năng ẩn câu hỏi ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng ẩn câu trả lời sẽ có thông báo được hiện lên “Bạn có muốn ẩn câu trả lời này hay không?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị câu trả lời thông qua mã của câu trả lời được chọn và cập nhật lại danh sách câu trả lời, nếu chọn không thì trở lại trang danh sách câu trả lời. |
| Đầu vào | Quản trị chọn câu trả lời cần ẩn và ấn nút ẩn câu câu trả lời. |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn ẩn câu trả lời của quản trị, nếu quản trị tiếp tục hệ thống sẽ lấy mã của câu trả lời đó và thực hiện việc chuyển trạng thái hiển thị của câu trả lời. |
| Đầu ra | Thông báo kết quả thành công và hiển thị lại danh sách câu trả lời |

1. Quản lí danh mục sản phẩm.
   * **Thêm danh mục sản phẩm.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý danh mục sản phẩm và chọn chức năng thêm danh mục sản phẩm thì màn hình thêm danh mục sản phẩm sẽ được hiển thị ra cho người dùng, tại đây quản trị viên sẽ nhập đầy đủ thông tin liên quan đến sản phẩm mà form thiết kế yêu cầu (Tên danh mục, mã danh mục, mô tả, danh mục cha (nếu có)). Sau khi nhập xong thông tin người dùng nhấn nút thêm danh mục sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ thêm mới danh mục sản phẩm vào cơ sở dữ liệu và hiển thị thông báo thành công. |
| Đầu vào | Quản trị nhập thông tin cần thiết cho sản phẩm mới: Tên danh mục, mã danh mục, mô tả, danh mục cha (nếu có). |
| Xử lý | Hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ thêm mới danh mục sản phẩm vào cơ sở dữ liệu và hiển thị thông báo thành công. |
| Đầu ra | Hiển thị thông báo thành công và chuyển về trang danh sách danh mục sản phẩm |

* + **Sửa danh mục sản phẩm.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý danh mục sản phẩm, danh sách các danh mục sản phẩm sẽ được hiển thị, tại mỗi danh mục sản phẩm sẽ có chức năng sửa hoặc xóa ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng sửa danh mục sản phẩm, màn hình sửa danh mục sản phẩm sẽ được hiển thị. Tại đây người dùng có thể sửa nội dung thông tin liên quan đến danh mục sản phẩm mà người dùng vừa chọn (Tên danh mục, mã danh mục, mô tả, danh mục cha (nếu có)). Sau khi nhập xong thông tin người dùng nhấn nút sửa danh mục sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ cập nhật nội dung vào cơ sở dữ liệu và hiển thị thông báo thành công, sau đó chuyển về trang quản lý danh mục sản phẩm. |
| Đầu vào | Người dùng chọn danh mục sản phẩm cần sửa, nhấn vào chức năng sửa danh mục sản phẩm và chỉnh sửa các thông tin cần thiết cho danh mục sản phẩm được chọn |
| Xử lý | Sau khi nhập xong thông tin người dùng nhấn nút sửa danh mục sản phẩm, hệ thống sẽ kiểm tra dữ liệu đã được nhập đầy đủ chưa, nếu chưa thì thông báo dữ liệu nhập vào chưa đầy đủ, nếu nhập đầy đủ thì hệ thống sẽ cập nhật nội dung vào cơ sở dữ liệu |
| Đầu ra | Hiển thị thông báo thành công và hiển thị lại trang quản lý danh mục sản phẩm |

* + **Xóa danh mục sản phẩm.**

|  |  |
| --- | --- |
| Mô tả | Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng quản lý danh mục sản phẩm, danh sách các danh mục sản phẩm sẽ được hiển thị, tại mỗi danh mục sản phẩm sẽ có chức năng sửa hoặc xóa ở ngay phía bên cạnh, khi người dùng nhấn vào chức năng xóa danh mục sản phẩm, sẽ có thông báo được hiện lên “Bạn có muốn xóa danh mục sản phẩm này hay không ?” nếu chọn có thì hệ thống sẽ chuyển trạng thái hiển thị của danh mục sản phẩm thông qua mã của danh mục sản phẩm được chọn và cập nhật lại danh sách danh mục sản phẩm, nếu chọn không thì trở lại trang danh sách danh mục sản phẩm |
| Đầu vào | Người dùng chọn danh mục sản phẩm cần xóa và chọn chức năng xóa. |
| Xử lý | Hệ thống xác nhận lại yêu cầu muốn xóa của người dùng, nếu người dùng tiếp tục hệ thống sẽ lấy mã của danh mục sản phẩm đó và thực hiện việc chuyển trạng thái hiển thị của danh mục sản phẩm. |
| Đầu ra | Thông báo đã xóa thành công. |

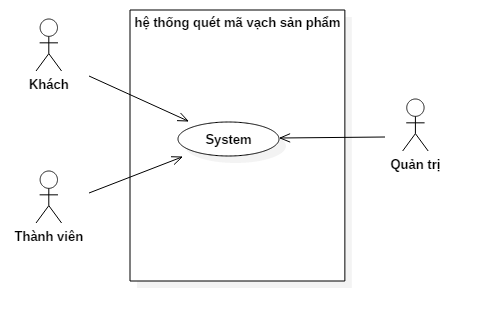
1. **Xem thống kê số lượt truy cập, số bình luận, số câu hỏi của app theo tên danh muc, tên sản phẩm trong khoảng thời gian do người dùng tự chọn**

|  |  |
| --- | --- |
| Mô tả | Trong quá trình người dùng sử dụng ứng dụng cần phải biết được thống kê cơ bản về số lượt truy cập, số bình luận, số câu hỏi của app theo thời gian để so sánh khả năng tương tác và khả năng được sử dụng của app để đưa ra những cải thiện hay nâng cấp cần thiết để giúp người dùng có trải nghiệm tốt hơn. Quá trình khi đăng nhập vào hệ thống quản trị của ứng dụng sẽ được hiển thị danh sách các chức năng, khi nhấn vào chức năng xem thống kê, màn hình thống kê sẽ được hiện ra. Tại đây quản trị chọn tên danh mục, tên sản phẩm và chọn khoảng thời gian để xem thống kê số lượt truy cập, số bình luận, số câu hỏi sau đó nhấn nút hiển thị. Kết quả trả về sẽ là thống kê tổng thể về số bình luận, số lượt truy cập ứng dụng, số câu hỏi, trả lời của sản phẩm hoặc của danh mục trong khoảng thời gian mà người dùng chọn |
| Đầu vào | tên danh mục, tên sản phẩm và chọn khoảng thời gian |
| Xử lý | Hệ thống kiểm tra các điều kiện, trường hợp dữ liệu trống thì sẽ hiển thị ra tất cả sản phẩm, tổng lượt truy cập app, số lượt truy cập sản phẩm, tổng số bình luận, tổng số câu hỏi, tổng số câu trả lời, trường hợp một vài trường dữ liệu thiếu thì việc tìm kiếm vẫn lấy dữ liệu trong cơ sở dữ liệu đối với những trường đã được nhập, sau đó hiển thị dữ liệu trả về lên giao diện thống kê |
| Đầu ra | Hiển thị danh sách và các thông số cần thống kê ( tổng số lượt truy cập app, số lượt truy cập sản phẩm, tổng số bình luận, câu hỏi, trả lời về sản phẩm, ) của các sản phẩm thỏa mãn điều kiện đầu vào |

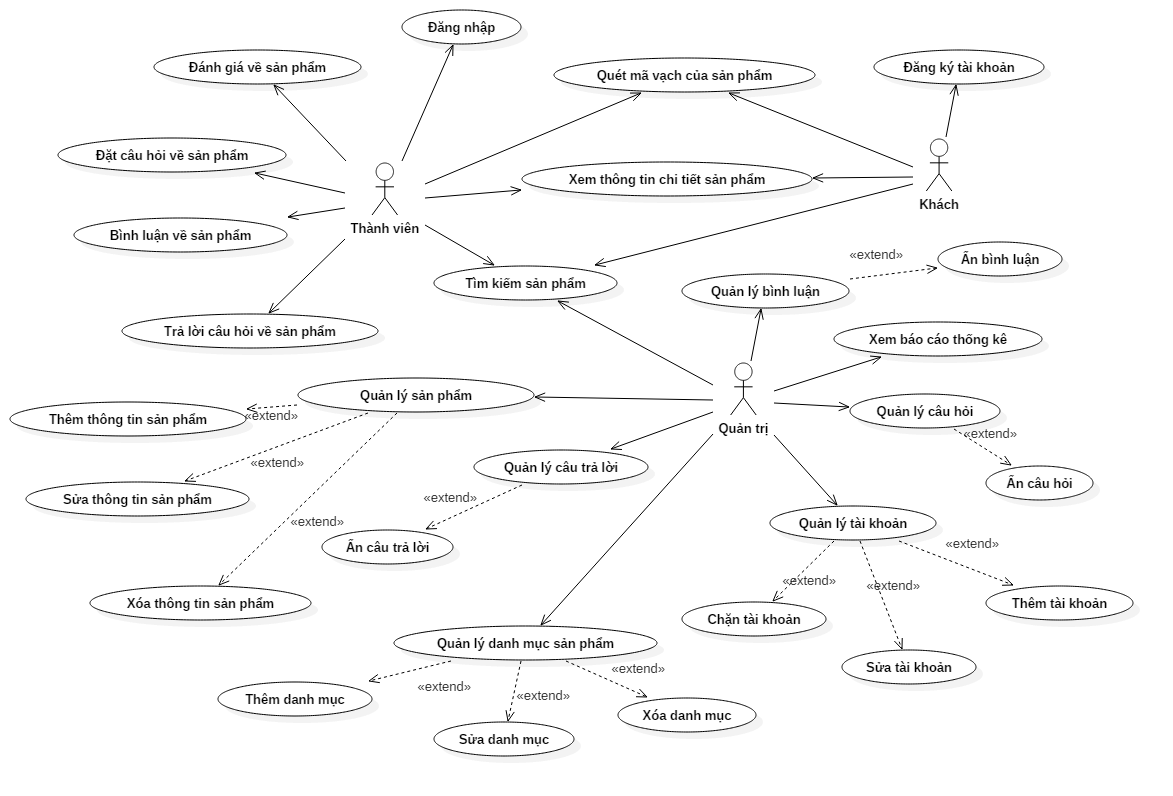
1. PHÂN TÍCH HỆ THỐNG.
   1. Mô tả bài toán.

Từ thực tế kiểm tra mã vạch để phân biệt hàng thật hàng giả và sự phổ biến của điện thoại thông minh hiện nay thì việc xây dựng ứng dụng quét mã vạch của sản phẩm ngay trên chính chiếc điện thoại để kiểm tra sản phẩm là hàng chính hãng hay hàng giả là cần thiết, đáp ứng các nhu cầu của xã hội, hỗ trợ cho việc kiểm tra thông tin sản phẩm ngay từ phía người tiêu dùng giúp việc mua bán diễn ra một cách nhanh chóng, tiết kiệm.

* 1. Mô hình vai trò người dùng

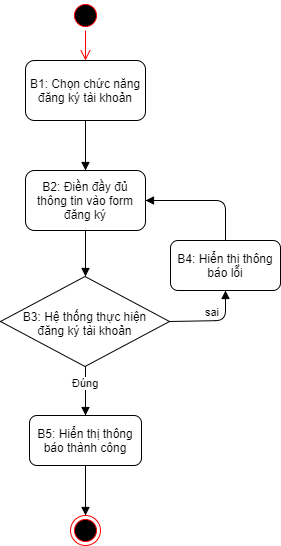


* 1. Tổng quan các usecase của hệ thống



* 1. Sơ đồ quy trình
     1. Đối với khách

1. **Đăng ký tài khoản**

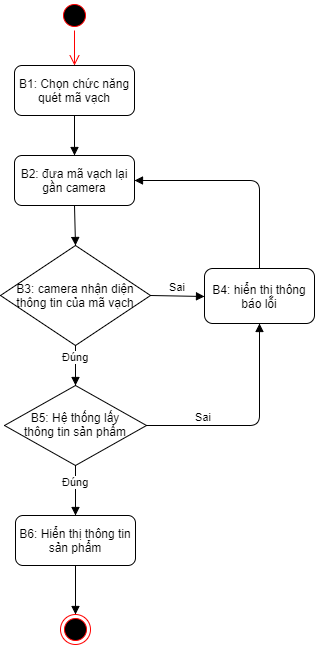
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng đăng kí tài khoản.
* **B2:** Người dùng điền tên đại diện, mật khẩu, email, số điện thoại vào form đăng kí.
* **B3:** Hệ thống kiểm tra dữ liệu đầu vào. Nếu đúng sẽ thêm dữ liệu vào cơ sở dữ liệu và chuyển chuyển sang bước 5. Nếu có lỗi xảy ra thì chuyển sang bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiện thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết tại chương 3 mục 3.4.1.a)**

1. **Quét mã vạch sản phẩm**

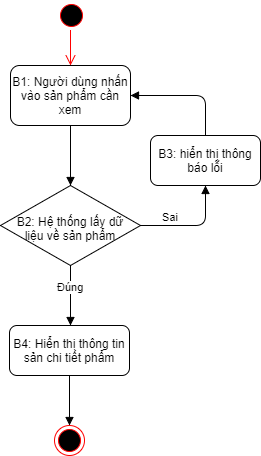
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng quét mã vạch.
* **B2:** Người dùng đưa mã vạch của sản phẩm lại gần camera .
* **B3:** Camera của hệ thống sẽ kiểm tra dữ liệu đầu vào. Kiểm tra dữ liệu trả về từ camera có phải là mã vạch hay không (cách kiểm tra mã vạch xem tại mục 2.5.4.b) Nếu đúng là mã vạch chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiển thị thông báo.
* **B5:** Lấy dữ liệu trong cơ sở dữ liệu. Nếu đúng chuyển sang bước 6. Nếu sai chuyển sang bước 4.
* **B6:** Hiển thị thông tin sản phẩm
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.1.b)**

1. **Xem thông tin chi tiết sản phẩm**

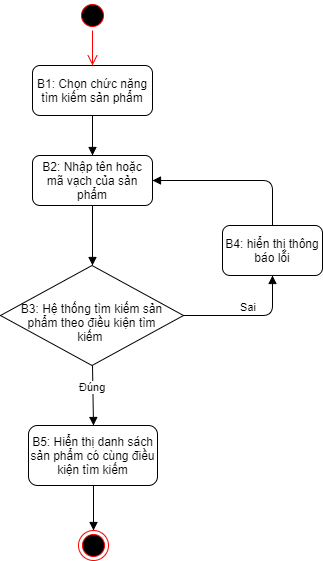
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng nhấn vào sản phẩm cần xem
* **B2:** Hệ thống lấy dữ liệu về sản phẩm được chọn từ cơ sở dữ liệu nếu thành công thì chuyển đến bước 4 ngược lại sẽ chuyển sang bước 3
* **B3:** Hiển thị thông báo lỗi
* **B4:** Hiển thị thông tin chi tiết sản phẩm lên giao diện cho người dùng
* **B5:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.1.c)**

1. **Tìm kiếm sản phẩm**

****

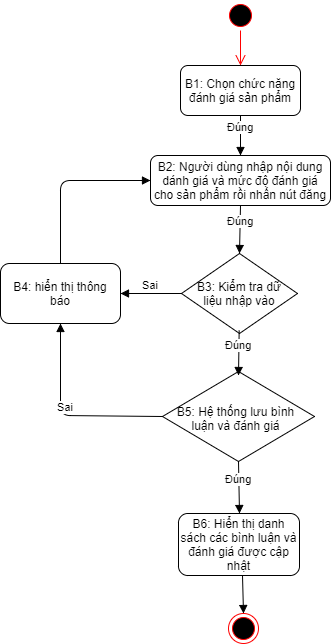
**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng tìm kiếm sản phẩm.
* **B2:** Người dùng nhập tên hoặc mã vạch của sản phẩm và nhấn vào nút tìm kiếm.
* **B3:** Hệ thống lấy dữ liệu phù hợp với từ khóa tìm kiếm trong cơ sở dữ liệu. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiển thị thông báo lỗi.
* **B5:** Hiển thị danh sách sản phẩm có cùng điều kiện tìm kiếm
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.1.d)**

* + 1. Đối với thành viên

1. Đánh giá sản phẩm và bình luận về sản phẩm đang xem

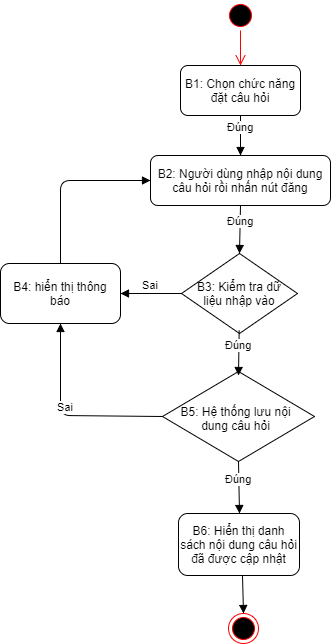


**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng đánh giá sản phẩm.
* **B2:** Người dùng nhập nội dùng đánh giá và mức độ đánh giá cho sản phẩm rồi nhấn vào nút đăng.
* **B3:** Hệ thống kiểm tra dữ liệu đầu vào. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Lưu bình luận và đánh giá vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6 ngược lại chuyển sang bước 4.
* **B6:** hiển thị danh sách các bình luận và đánh giá của từng thành viên đã được cập nhật.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.2.a)**

1. **Đặt câu hỏi về sản phẩm đang xem**

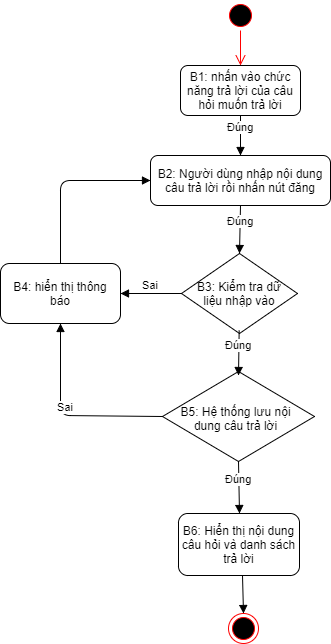
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn tab đặt câu hỏi cho sản phẩm
* **B2:** Người dùng nhập nội dung câu hỏi rồi nhấn nút đăng
* **B3:** Hệ thống kiểm tra dữ liệu do người dùng nhập vào. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ lưu dữ liệu vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6, ngược lại sẽ quay lại bước 4.
* **B6:** Hiển thị danh sách câu hỏi đã được cập nhật.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.2.b)**

1. **Trả lời câu hỏi về sản phẩm**

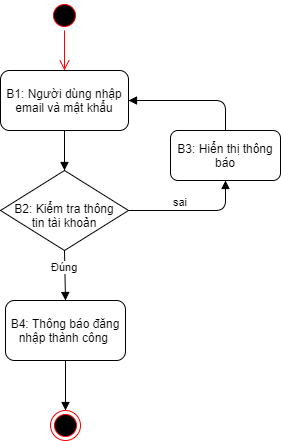
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng nhấn vào chức năng trả lời của câu hỏi muốn trả lời
* **B2:** Người dùng nhập nội dung câu trả lời rồi nhấn icon đăng
* **B3:** Hệ thống kiểm tra dữ liệu do người dùng nhập vào. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ lưu dữ liệu vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị câu hỏi và danh sách câu trả lời đã được cập nhật.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.2.c)**

1. **Đăng nhập**

****

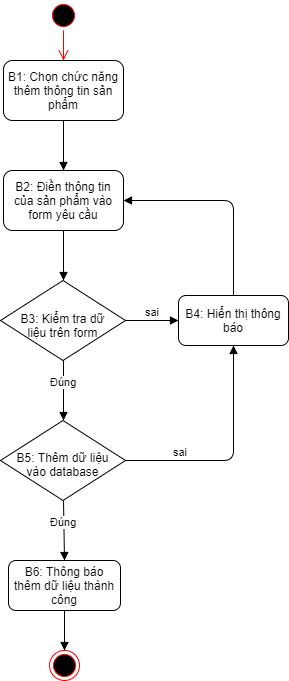
**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng nhập email và mật khẩu
* **B2:** Hệ thống sẽ thực hiện việc kiểm tra thông tin tài khoản. Nếu đúng chuyển sang bước 4. Nếu sai chuyển sang bước 3.
* **B3:** Hiện thông báo lỗi.
* **B4:** Hiển thị thông báo đăng nhập thành công.
* **B5:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.2.d)**

* + 1. Đối với quản trị

1. **Thêm thông tin sản phẩm**

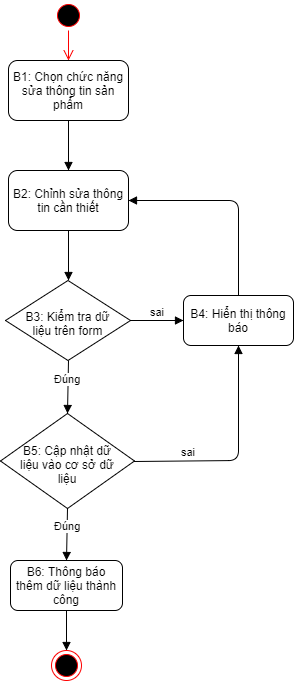
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng thêm thông tin sản phẩm
* **B2:** Người dùng điền đầy đủ thông tin của sản phẩm (hình ảnh sản phẩm, giá sản phẩm, danh mục sản phẩm, nhà sản xuất, địa chỉ mua hàng, mô tả về sản phẩm) vào form yêu cầu
* **B3:** Hệ thống kiểm tra dữ liệu trên form. Nếu đúng chuyển sang bước 5. Nếu sai chuyển sang bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ cập nhật dữ liệu sản phẩm vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị thông báo thêm dữ liệu thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.a)**

1. **Sửa thông tin sản phẩm**

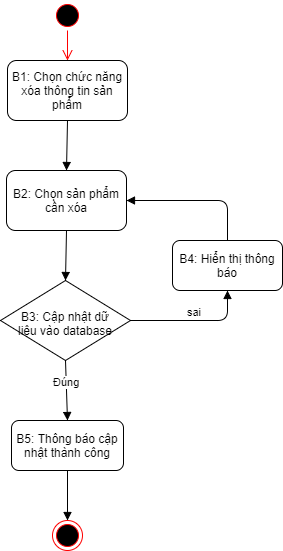
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng sửa thông tin sản phẩm
* **B2:** Người dùng chỉnh sửa thông tin cần thiết về sản phẩm
* **B3:** Hệ thống kiểm tra dữ liệu trên form. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ cập nhật dữ liệu vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu sai quay lại bước 4.
* **B6:** Hiển thị thông báo thêm dữ liệu thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.a)**

1. **Xóa thông tin sản phẩm**

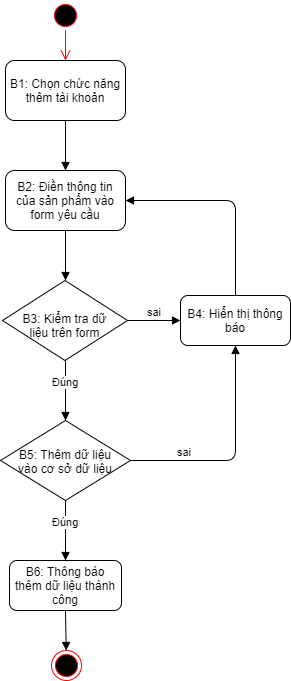
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng xóa thông tin sản phẩm
* **B2:** Người dùng chọn sản phẩm cần xóa
* **B3:** Hệ thống sẽ thực hiện việc xóa thông tin sản phẩm. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.a)**

1. **Thêm tài khoản**

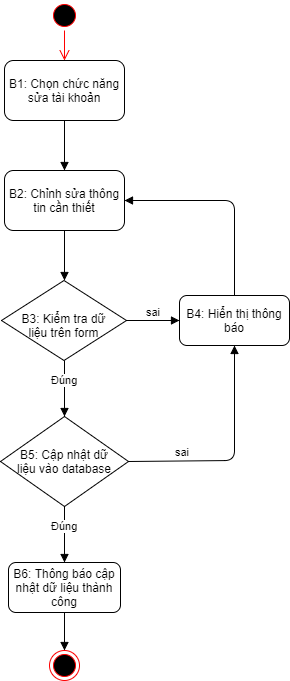
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng thêm tài khoản
* **B2:** Người dùng điền thông tin của người dùng vào form yêu cầu ( Tên đại diện trong khoảng từ 8 -16 ký tự, mật khẩu có độ dài từ 8- 24 ký tự, email phải đúng định dạnh chuẩn của email, số điện thoại phải đúng định dạng)
* **B3:** Hệ thống kiểm tra dữ liệu trên form. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ thêm dữ liệu vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị thông báo cập nhật dữ liệu thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.b)**

1. **Sửa tài khoản**

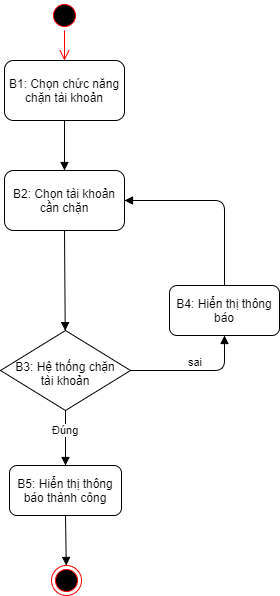
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng sửa tài khoản
* **B2:** Người dùng chỉnh sửa thông tin cần thiết trên form (tên đại diện, mật khẩu, email, số điện thoại)
* **B3:** Hệ thống kiểm tra dữ liệu trên form. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống sẽ cập nhật dữ liệu vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị thông báo cập nhật dữ liệu thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.b)**

1. **Chặn tài khoản**

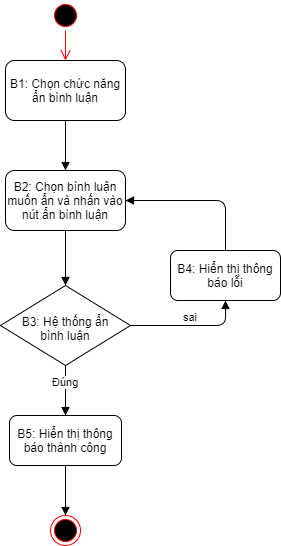
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng chặn tài khoản
* **B2:** Người dùng chọn tài khoản muốn chặn và nhấn vào nút chặn tài khoản
* **B3:** Hệ thống sẽ thực hiện việc chặn tài khoản. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.b)**

1. **ẩn bình luận**

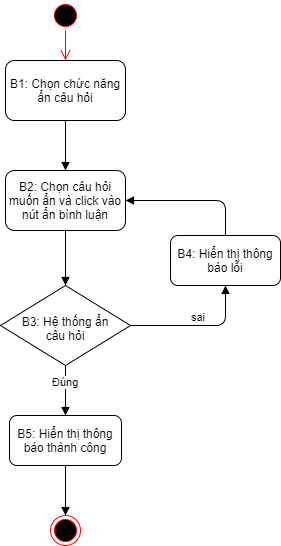
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng ẩn bình luận
* **B2:** Người dùng chọn bình luận muốn ẩn và nhấn vào nút ẩn bình luận
* **B3:** Hệ thống sẽ thực hiện việc ẩn bình luận. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.c)**

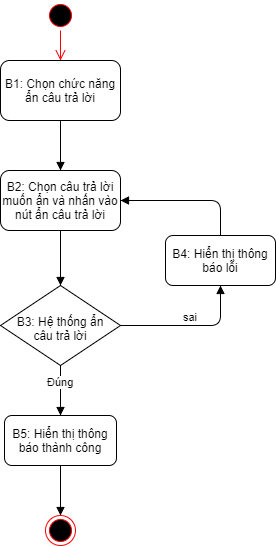
1. **ẩn câu hỏi**

****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng ẩn câu hỏi
* **B2:** Người dùng chọn câu hỏi muốn ẩn và nhấn vào nút ẩn câu hỏi
* **B3:** Hệ thống sẽ thực hiện việc ẩn câu hỏi. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc
* **(Chi tiết xem tại chương 3 mục 3.4.3.c)**

1. **ẩn câu trả lời**

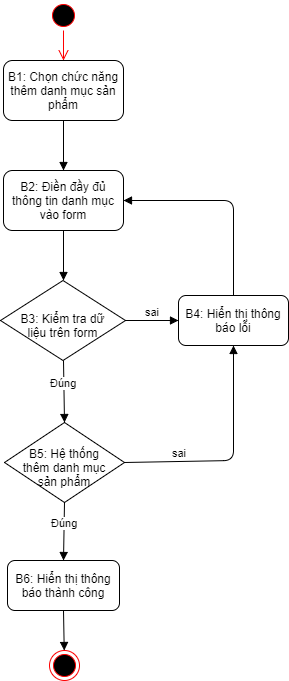
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng ẩn câu trả lời
* **B2:** Người dùng chọn câu trả lời muốn ẩn và nhấn vào nút ẩn câu trả lời
* **B3:** Hệ thống sẽ thực hiện việc ẩn câu trả lời. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.c)**

1. **thêm danh mục sản phẩm**

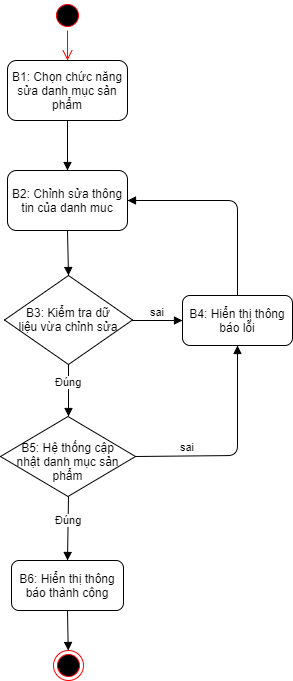
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng thêm danh mục sản phẩm
* **B2:** Người dùng điền đầy đủ thông tin danh mục (Tên danh mục, mã danh mục, mô tả, danh mục cha (nếu có)) vào form
* **B3:** Hệ thống kiểm tra dữ liệu trên form. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống thêm danh mục sản phẩm vào cơ sở dữ liệu. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị thông báo thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.d)**

1. **sửa danh mục sản phẩm**

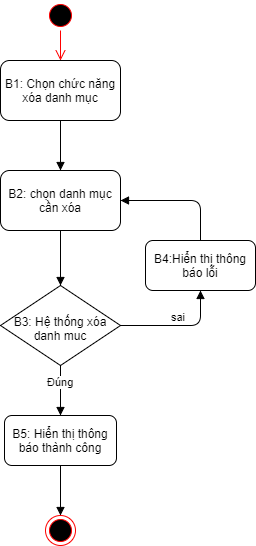
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng sửa danh mục sản phẩm
* **B2:** Người dùng chỉnh sửa thông tin của danh mục (Tên danh mục, mã danh mục, mô tả, danh mục cha) cần chỉnh sửa
* **B3:** Hệ thống kiểm tra dữ liệu vừa chỉnh sửa. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống cập nhật thông tin của danh mục được chọn. Nếu thành công chuyển sang bước 6. Nếu không thành công quay lại bước 4.
* **B6:** Hiển thị thông báo thành công.
* **B7:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.d)**

1. **xóa danh mục sản phẩm**

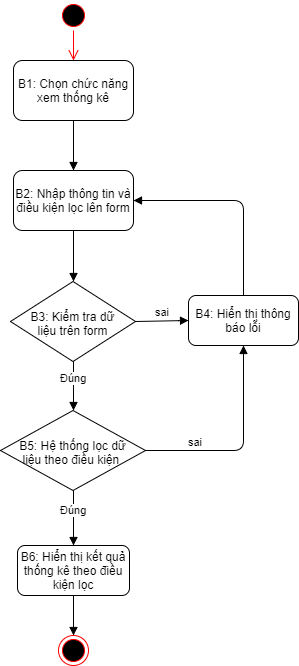
****

**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng xóa danh mục sản phẩm
* **B2:** Người dùng chọn danh mục cần xóa
* **B3:** Hệ thống sẽ thực hiện việc xóa danh mục. Nếu thành công chuyển sang bước 5. Nếu không thành công quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hiển thị thông báo thành công.
* **B6:** Kết thúc

**(Chi tiết xem tại chương 3 mục 3.4.3.d)**

1. **xem thống kê**

****

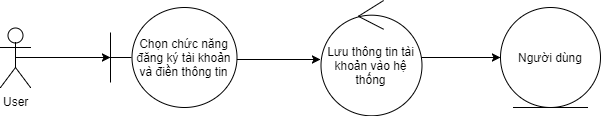
**Mô tả:**

* **B0:** Bắt đầu.
* **B1:** Người dùng chọn chức năng xem thống kê
* **B2:** Người dùng nhập thông tin và điều kiện lọc dữ liệu (tên danh mục, tên sản phẩm và chọn khoảng thời gian) về thống kê muốn xem
* **B3:** Hệ thống kiểm tra dữ liệu mà người dùng nhập trên form. Nếu đúng chuyển sang bước 5. Nếu sai quay lại bước 4.
* **B4:** Hiện thông báo lỗi.
* **B5:** Hệ thống lọc dữ liệu theo điều kiện. Nếu thành công chuyển sang bước 6. Nếu không thành công chuyển sang bước 4.
* **B6:** Hiển thị kết quả thống kê theo điều kiện lọc.
* **B7:** Kết thúc

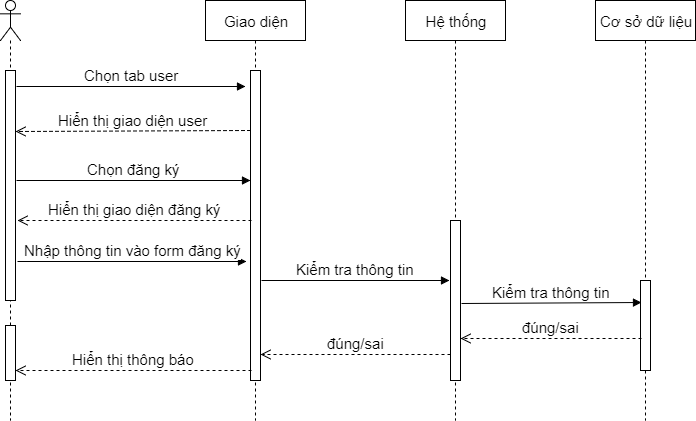
**(Chi tiết xem tại chương 3 mục 3.4.3.e)**

1. PHÂN TÍCH THIẾT KẾ.
   1. Biểu đồ lớp
      1. Đối với khách truy cập vào app

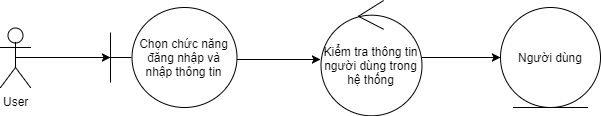
+ Đăng ký tài khoản:



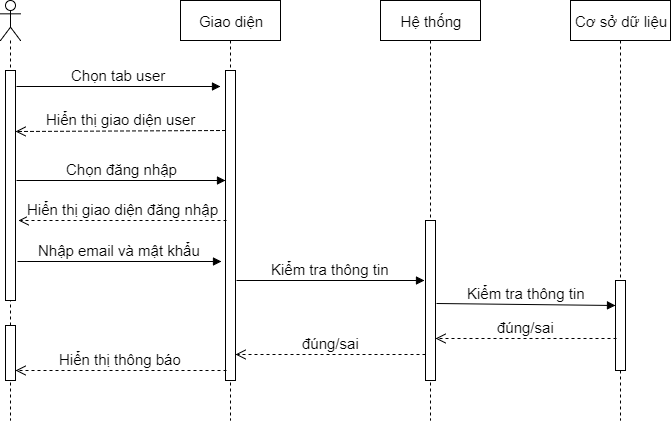
Biểu đồ tuần tự



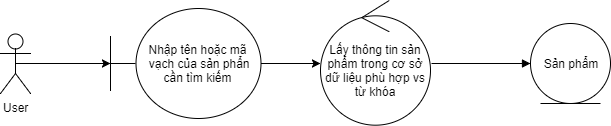
+ Đăng nhập vào hệ thống:



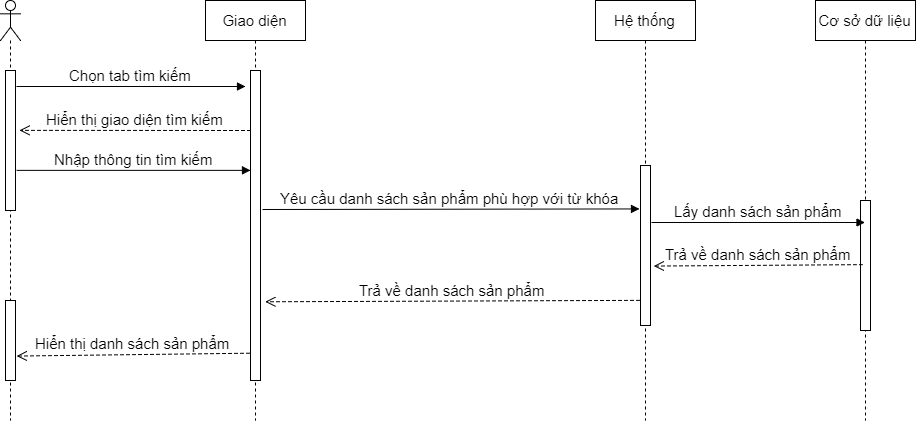
Biểu đồ tuần tự



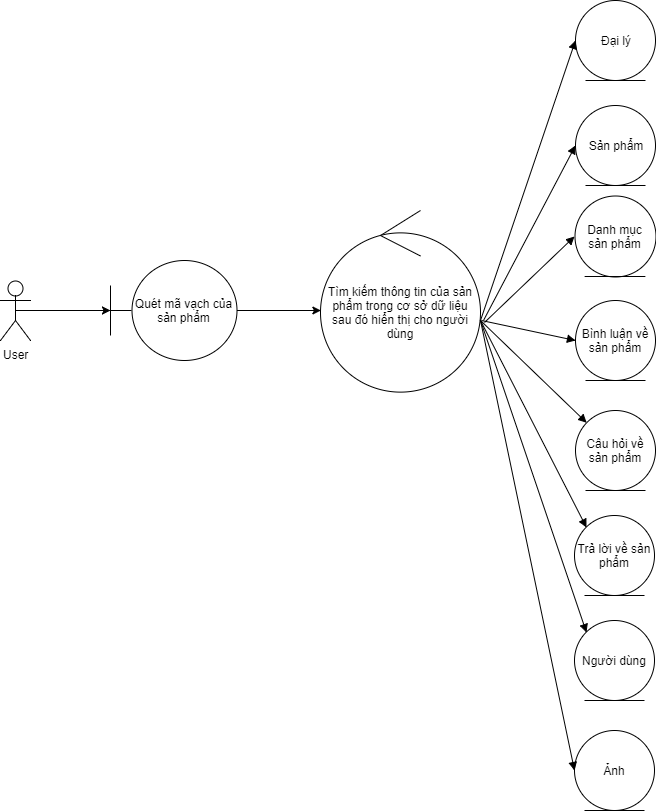
+ Tìm kiếm sản phẩm



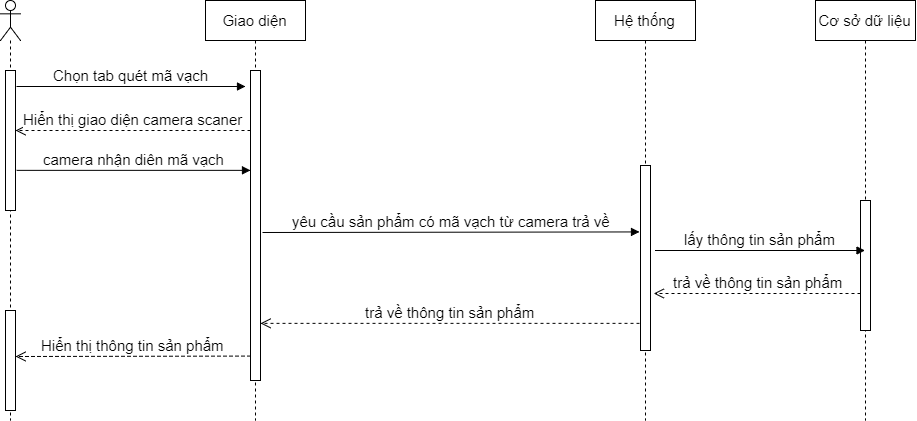
Biểu đồ tuần tự



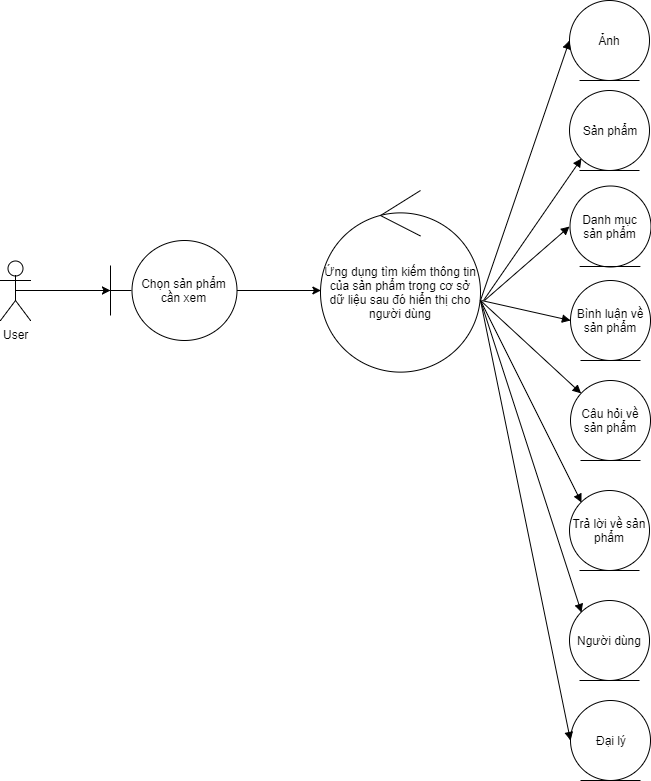
+ Quét mã vạch của sản phẩm:



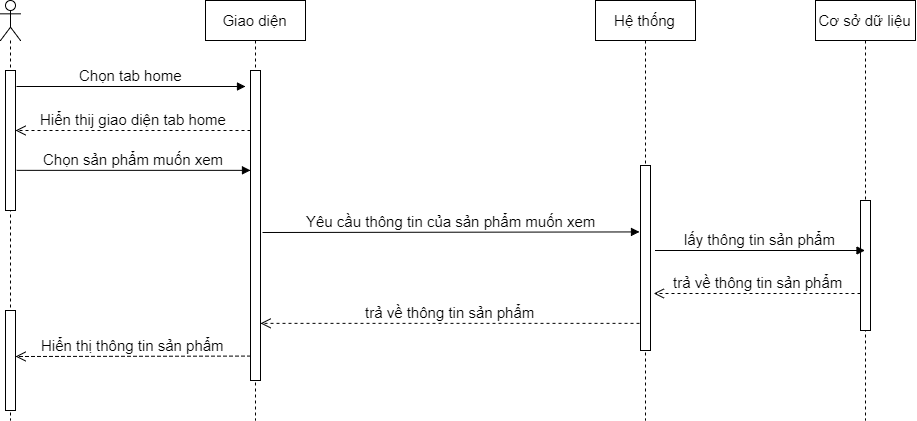
Biểu đồ tuần tự



+ Xem thông tin chi tiết sản phẩm:

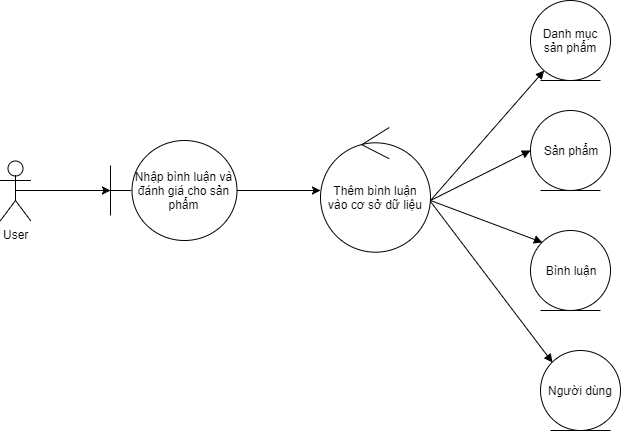


Biểu đồ tuần tự

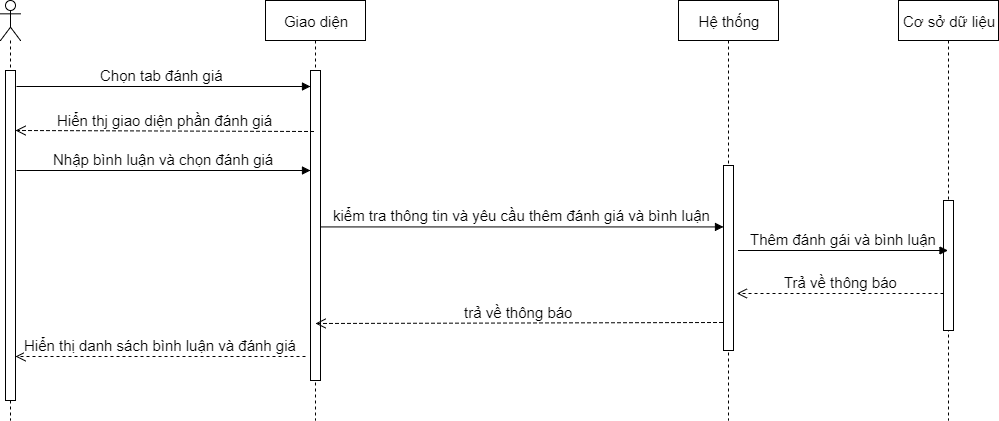


* + 1. Thành viên( người đã đăng ký tài khoản với hệ thống)

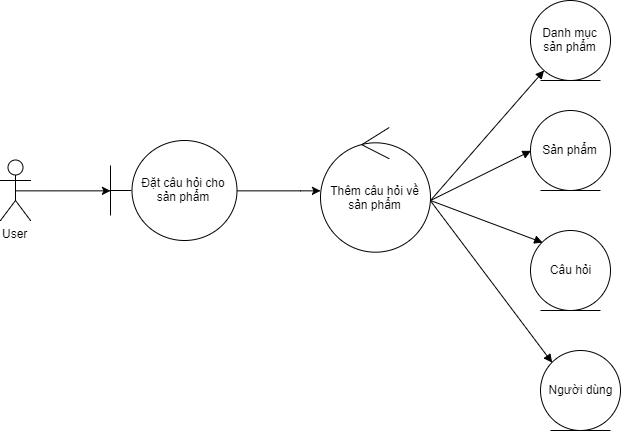
**+ Đánh giá và bình luận về sản phẩm đang xem**

****

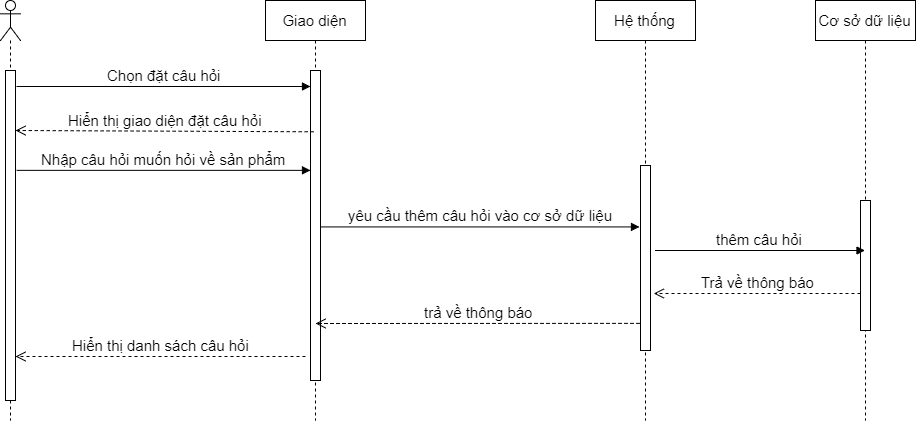
**Biểu đồ tuần tự**

****

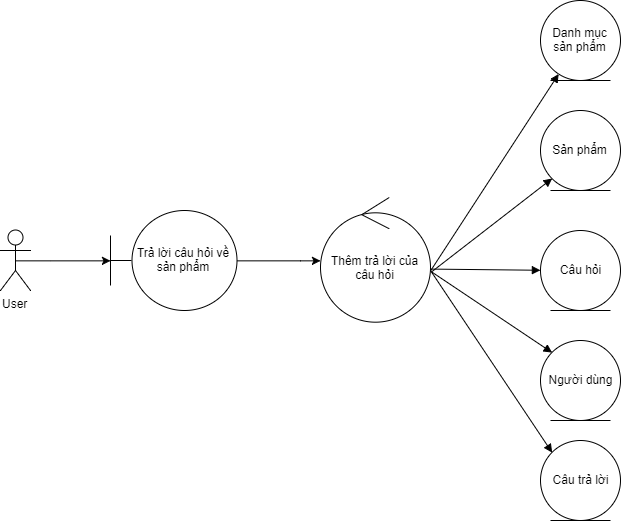
**+ Đặt câu hỏi về sản phẩm đang xem**

****

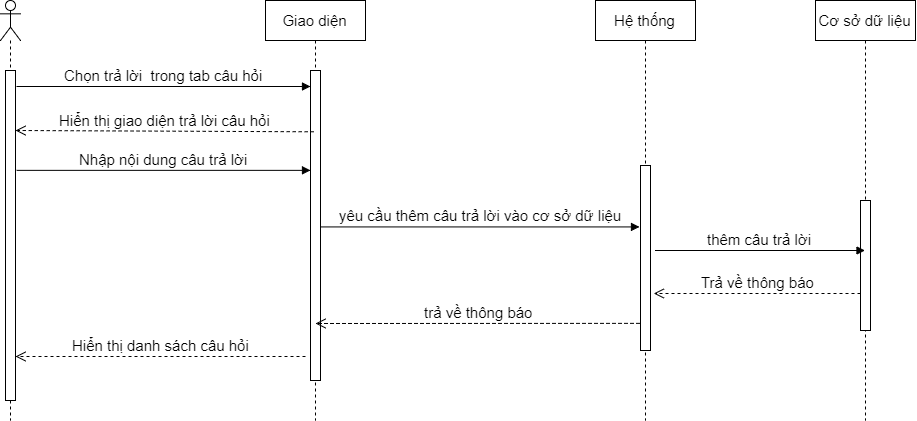
**Biểu đồ tuần tự**

****

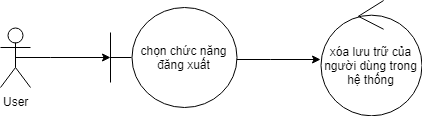
**+ Trả lời câu hỏi về sản phẩm đang xem**

****

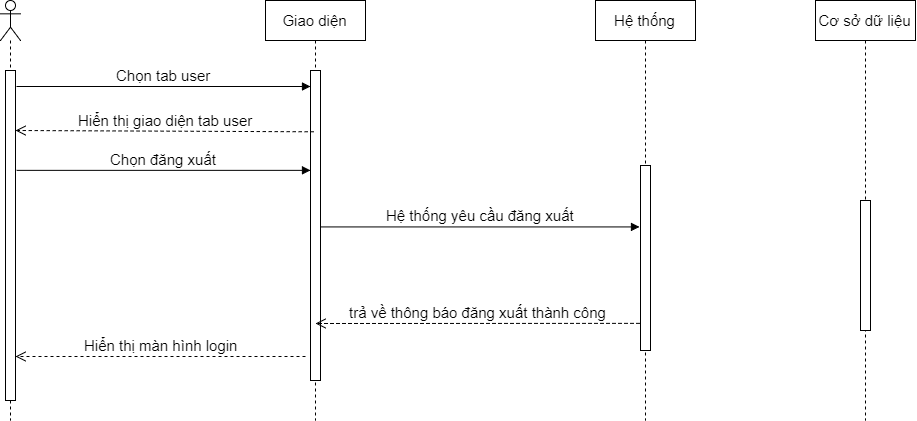
**Biểu đồ tuần tự**

****

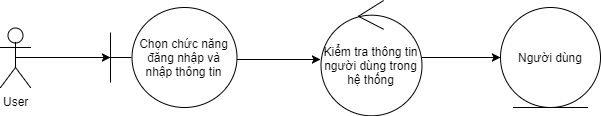
**+ Đăng xuất tài khoản khỏi hệ thống**

****

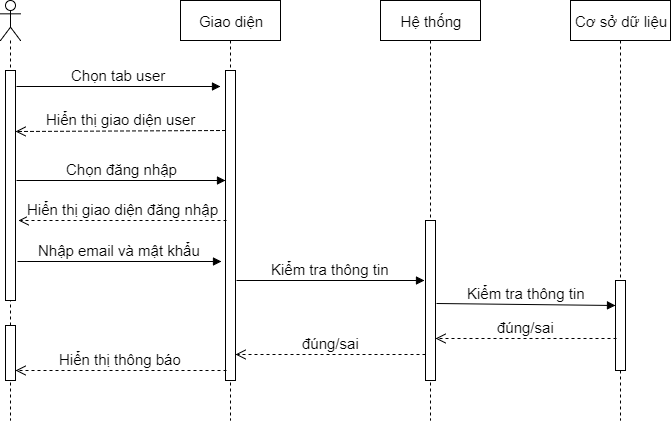
**Biểu đồ tuần tự**

****

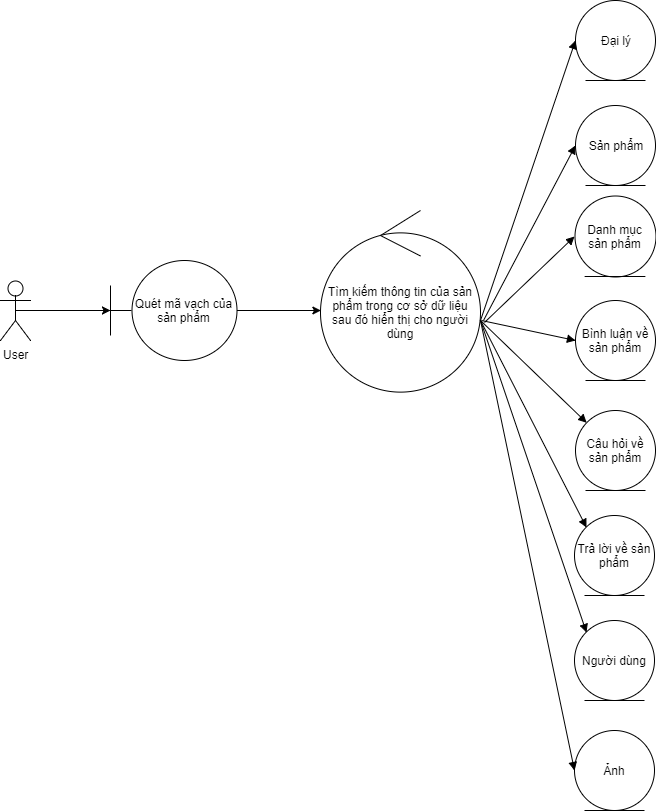
**+ Đăng nhập vào hệ thống**

****

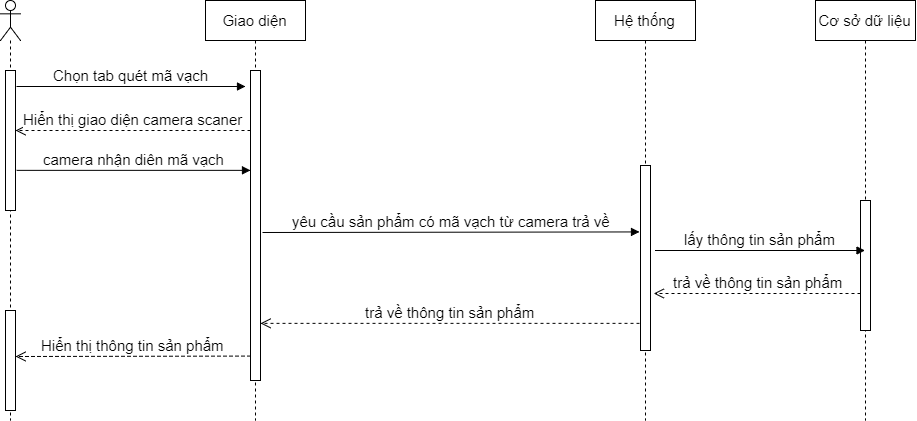
**Biểu đồ tuần tự**

****

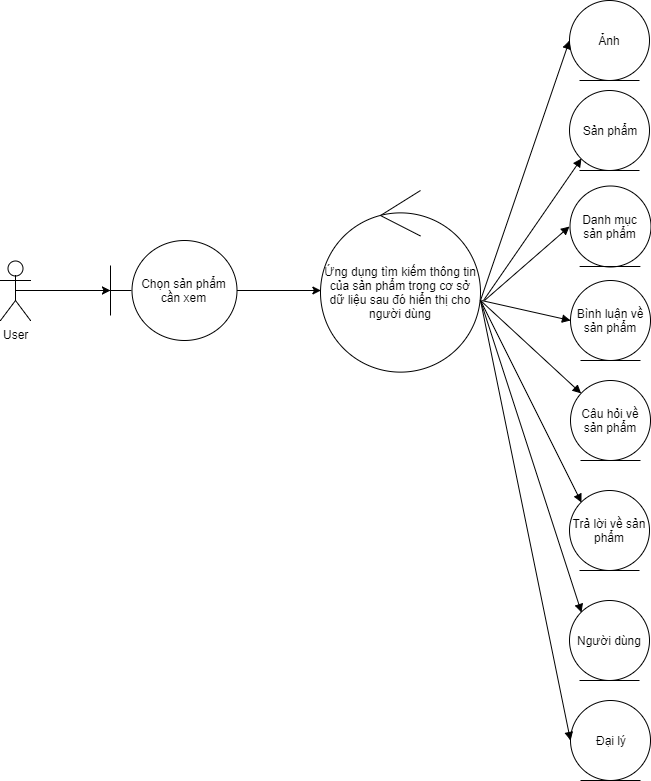
**+ Quét mã vạch của sản phẩm**

****

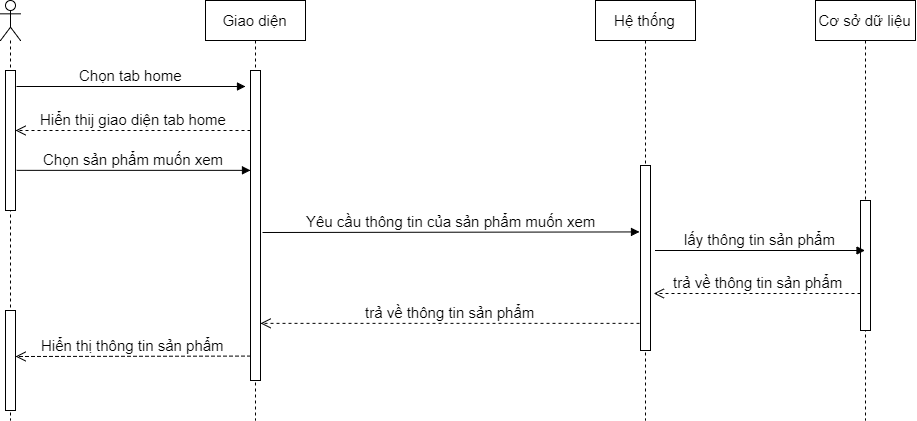
**Biểu đồ tuần tự**

****

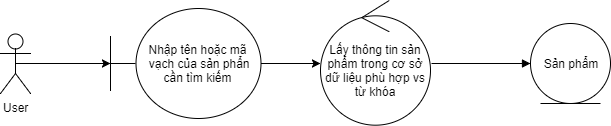
**+ Xem thông tin chi tiết sản phẩm**

****

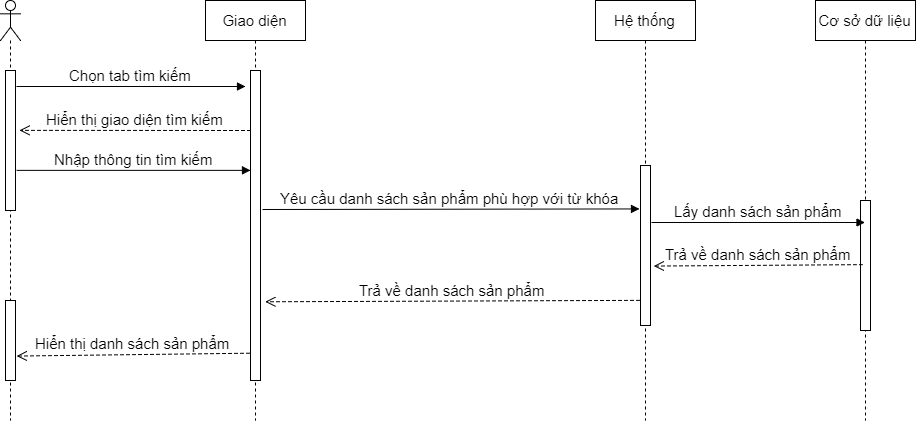
**Biểu đồ tuần tự**

****

**+ Tìm kiếm sản phẩm**

****

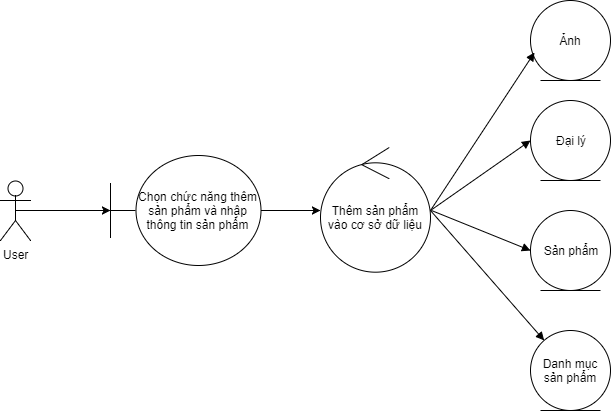
**Biểu đồ tuần tự**

****

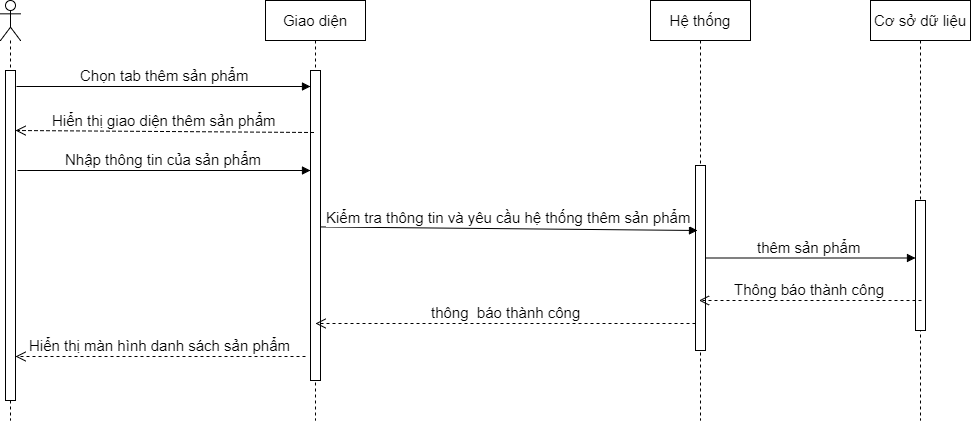
* + 1. Quản trị hệ thống

**+ Quản lý sản phẩm**

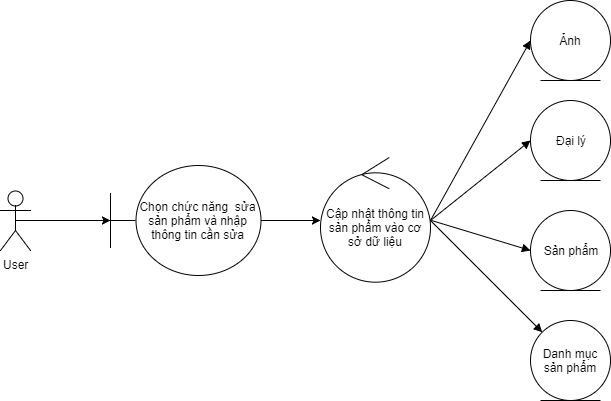
* **Thêm sản phẩm**

****

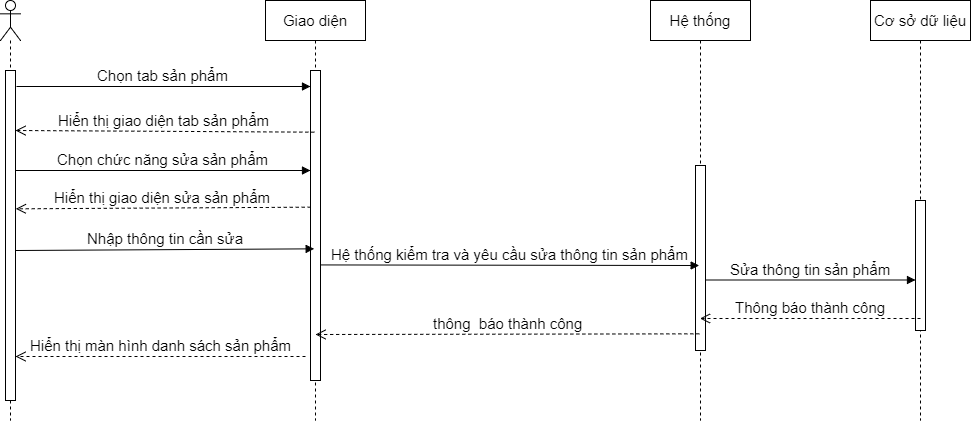
**Biểu đồ tuần tự**

****

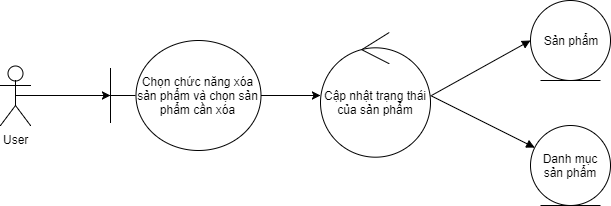
* **Sửa sản phẩm**

****

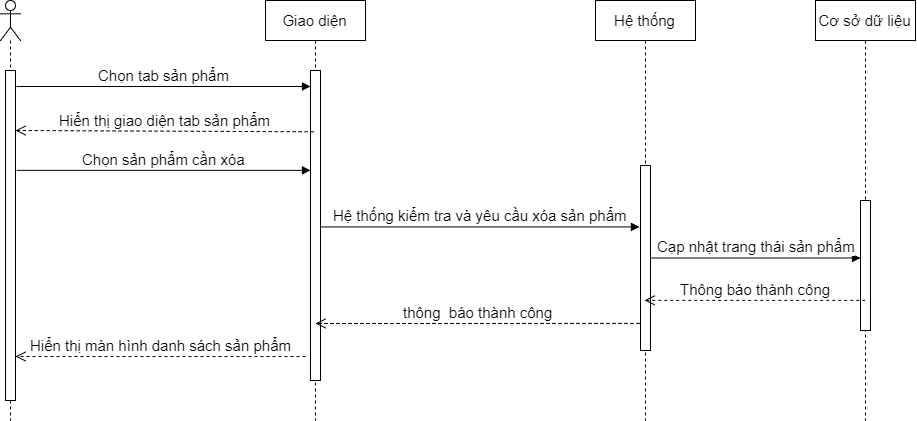
**Biểu đồ tuần tự**

****

* **Xóa sản phẩm**

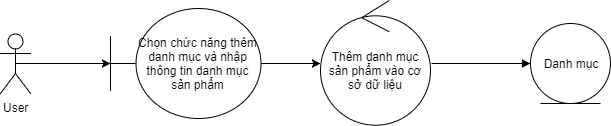
****

**Biểu đổ tuần tự**

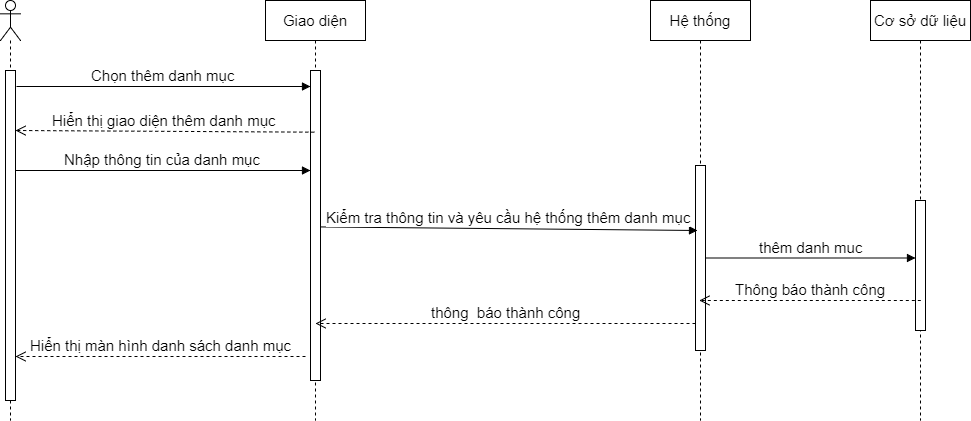
****

**+ Quản lý danh mục sản phẩm**

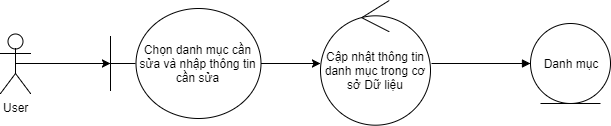
* **Thêm danh mục**

****

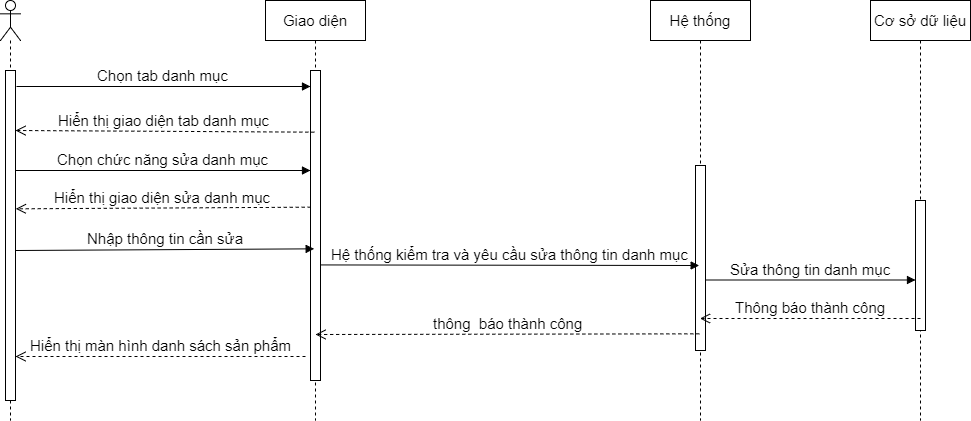
**Biểu đồ tuần tựu**

****

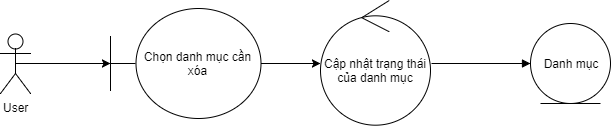
* **Sửa danh mục**

****

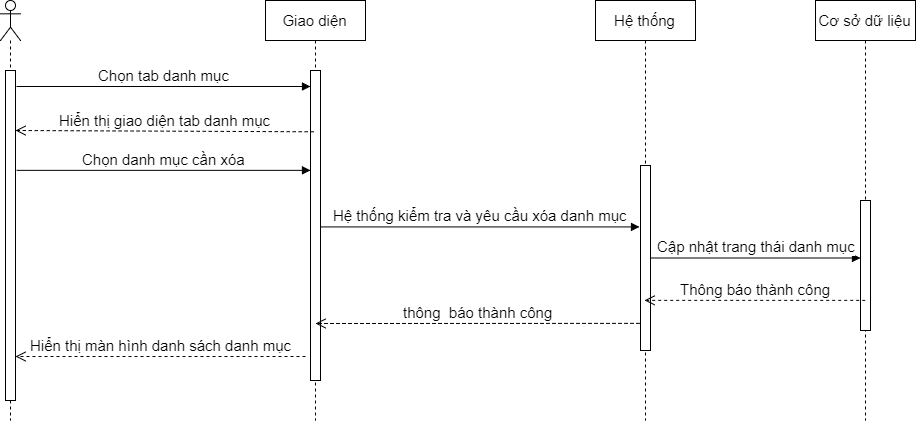
**Biểu đồ tuần tự**

****

* **Xóa danh mục**

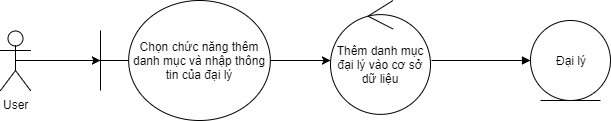
****

**Biểu đồ tuần tự**

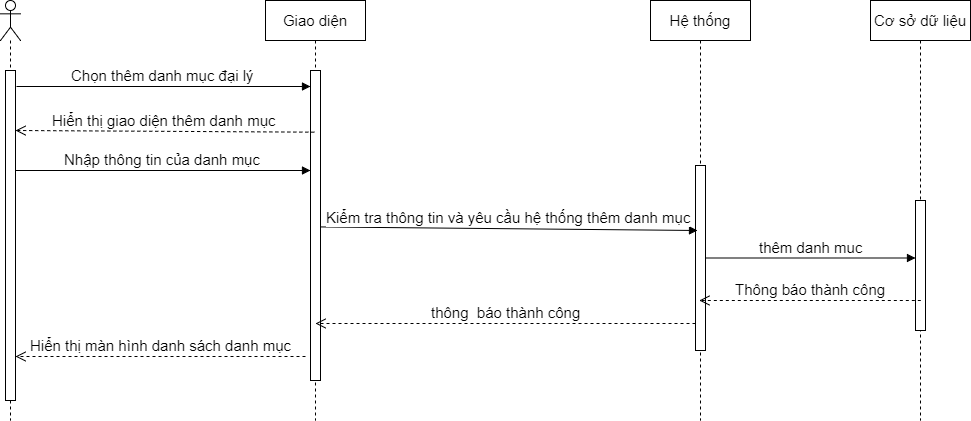
****

**+ Quản lý danh mục đại lý**

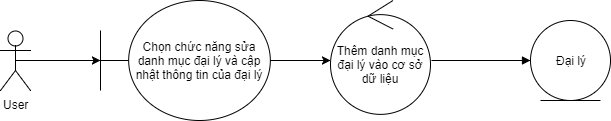
* **Thêm danh mục**

****

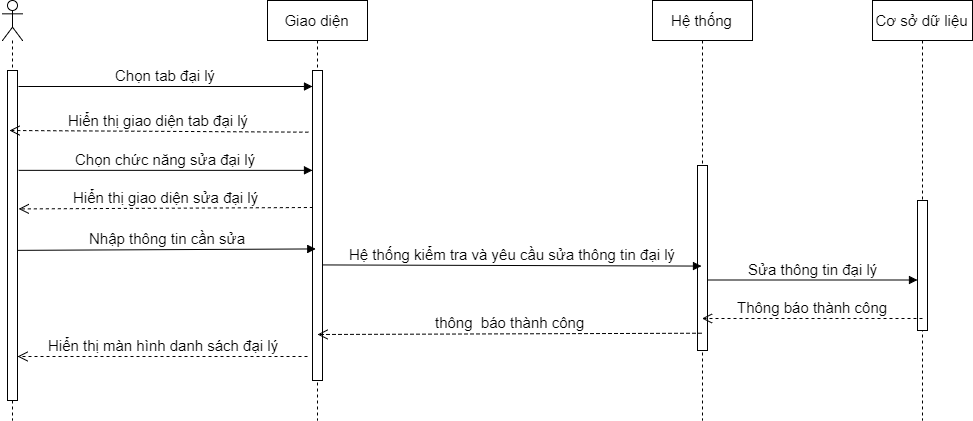
**Biểu đồ tuần tự**

****

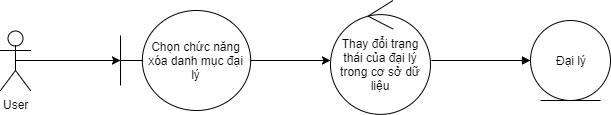
* **Sửa danh mục**

****

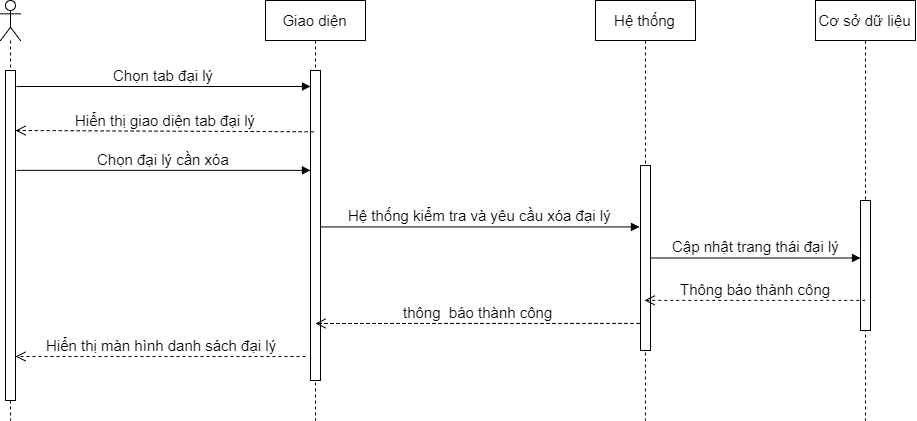
**Biểu đồ tuần tự**

****

* **Xóa danh mục**

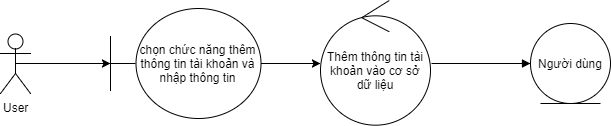
****

**Biểu đồ tuần tự**

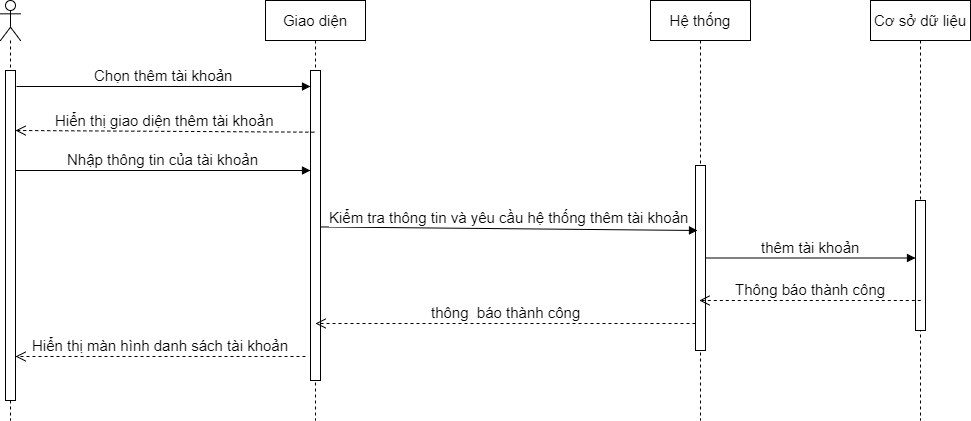
****

**+ Quản lý tài khoản**

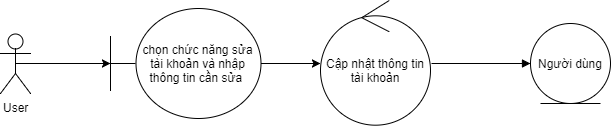
* **Thêm tài khoản**

****

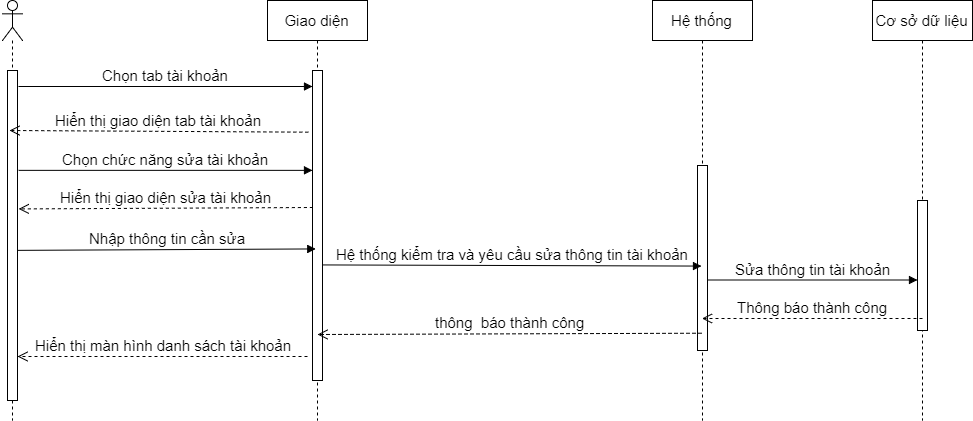
**Biểu đồ tuần tựu**

****

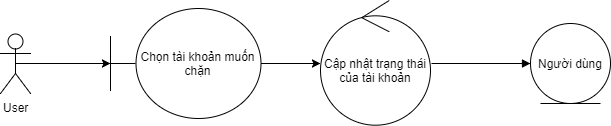
* **Sửa tài khoản**

****

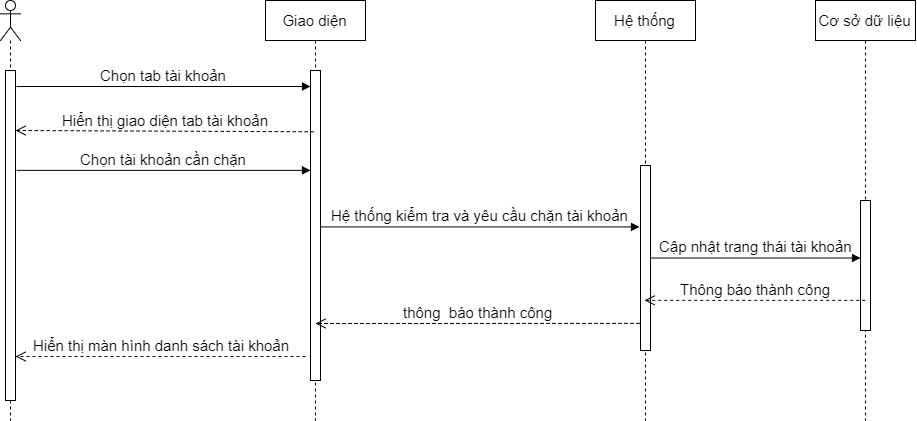
**Biểu đồ tuần tựu**

****

* **Chặn tài khoản**

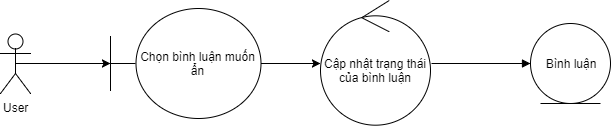
****

**Biểu đồ tuần tựu**

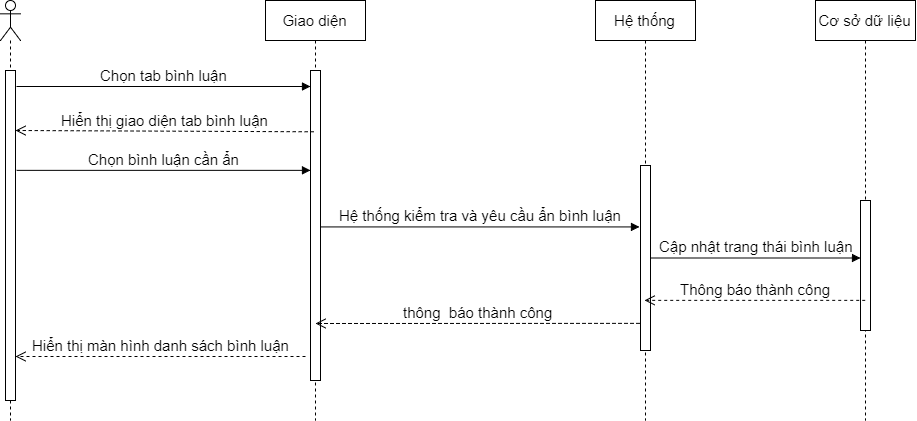
****

**+ Quản lý bình luận, câu hỏi, câu trả lời**

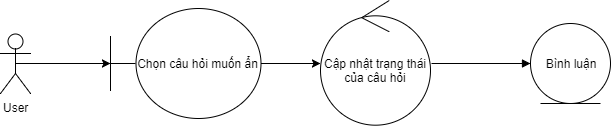
* **Ẩn bình luận**

****

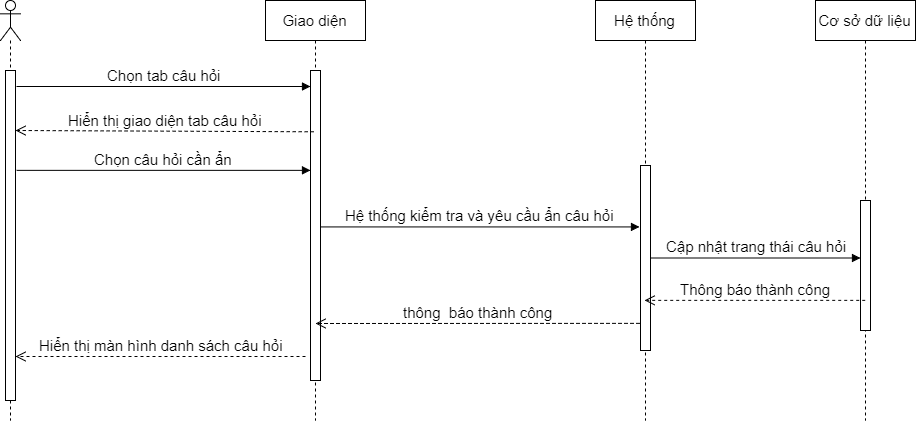
**Biểu đồ tuần tự**

****

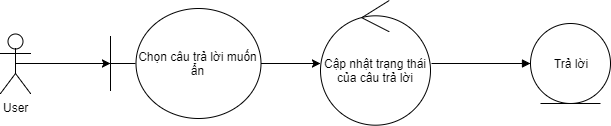
* **Ẩn câu hỏi**

****

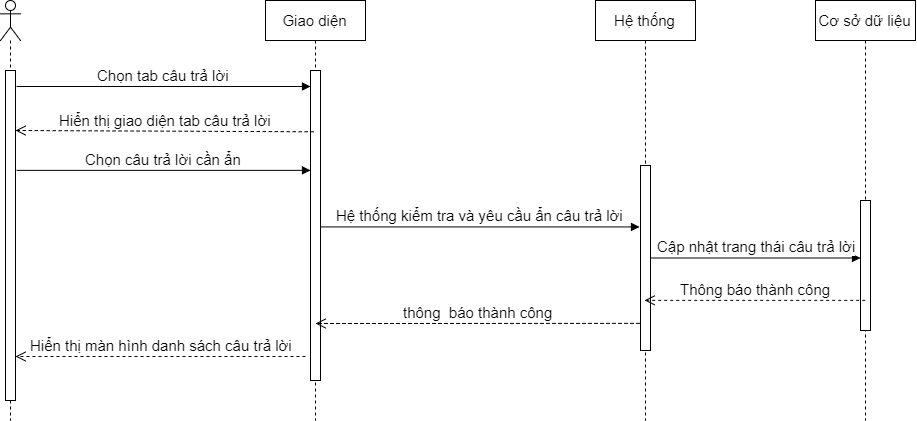
**Biểu đồ tuần tự**

****

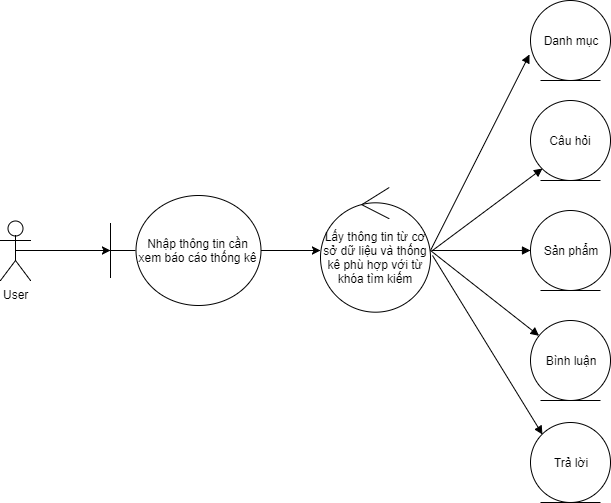
* **Ẩn câu trả lời**

****

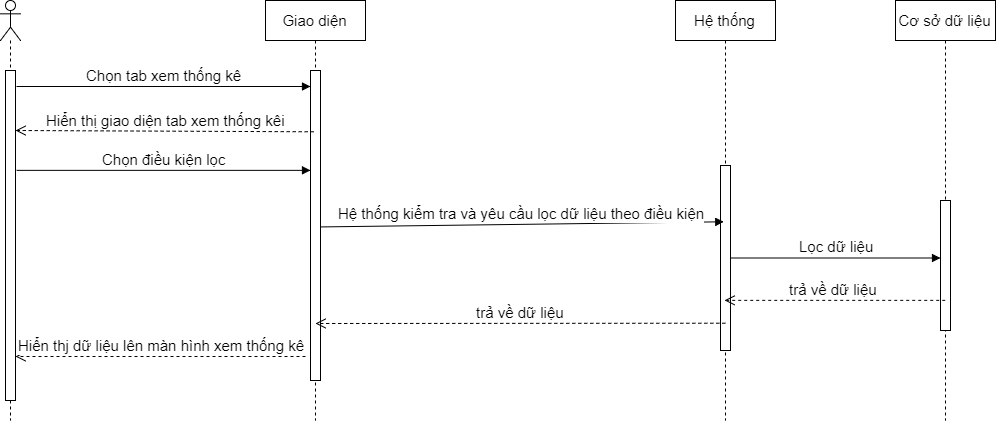
**Biểu đồ tuần tự**

****

**+ Xem thống kê về số lượt truy cập app, số lượt bình luận và đánh giá cho từng sản phẩm dựa vào điều kiện tìm kiếm (tên sản phẩm, danh mục thời gian)**

****

**Biểu đồ tuần tựu**

****

* 1. Đồng bộ hóa các thực thể

Từ biểu đồ lớp ta rút ra các thực thể còn lại sau khi đồng bộ là:

1. Thành viên, sản phẩm, doanh nghiệp, quốc gia, danh mục sản phẩm, ảnh, đại lý
2. thực thể yếu: câu hỏi, câu trả lời, bình luận
   * 1. Các đối tượng chính trong hệ thống
3. Thành viên (User): chứa thông tin các tài khoản đăng nhập vào hệ thống

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID\_USER | Id của người dùng |
| USER\_GROUP | Nhóm người sử dụng (0: người dùng; 1: quản trị) |
| USER\_NAME | Tên tài khoản |
| PASSWORD | Mật khẩu người dùng |
| SEX | Giới tính |
| EMAIL | Email người dùng |
| DESCRIPTION | Mô tả ngắn về người dùng |
| AVATAR | Ảnh đại diện của người dùng |
| PHONE | Số điện thoại của người dùng |
| ADDRESS | Địa chỉ của người dùng |
| STATUS | Ngày tạo tài khoản |
| UP\_DATE | Ngày cập nhật tài khoản gần nhất |
| CRT\_DATE | Ngày tạo tài khoản |

1. Câu hỏi (Question): lưu câu hỏi về sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của câu hỏi |
| QUESTION\_CONTENT | Nội dung câu hỏi |
| STATUS | Trạng thái hiển thị của bình luận |
| UP\_DATE | Ngày cập nhật câu hỏi |
| CRT\_DATE | Ngày tạo câu hỏi |

1. Câu trả lời (Answer): lưu câu trả lời của câu hỏi về sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của câu trả lời |
| ANSWEAR\_CONTENT | Id của thành viên |
| STATUS | Trạng thái của câu trả lời |
| UP\_DATE | Ngày cập nhật câu hỏi |
| CRT\_DATE | Ngày tạo câu hỏi |

1. Bình luận (Review): lưu bình luận về sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của bình luận |
| CONTENT\_REVIEW | Nội dung bình luận |
| APPROVAL | Đánh giá của người dùng về sản phẩm |
| STATUS | Trạng thái của bình luận |
| UP\_DATE | Ngày cập nhật bình luận |
| CRT\_DATE | Ngày tạo bình luận |

1. Sản phẩm (Product): lưu thông tin về sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của sản phẩm |
| PRODUCT\_CODE | Mã của sản phẩm |
| PRODUCT\_NAME | Tên sản phẩm |
| PRICE | Giá cả |
| APPROVAL | Đánh giá tổng quát của người dùng về sản phẩm |
| TITLE | Tiêu đề về sản phẩm |
| DESCRIPTION | Mô tả về sản phẩm |
| STATUS | Trạng thái của sản phẩm |
| UP\_DATE | Ngày cập nhật sản phẩm |
| CRT\_DATE | Ngày tạo sản phẩm |

1. Danh mục sản phẩm (Category): lưu danh mục của sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của danh mục |
| TYPE\_CATEGOTY | Mã danh mục |
| NAME\_CATEGORY | Tên danh mục |
| IMAGE | hình ảnh của danh mục |
| STATUS | Trạng thái của danh mục |
| DESCRIPTION | Mô tả cho danh mục |
| UP\_DATE | Ngày cập nhật danh mục |
| CRT\_DATE | Ngày tạo danh mục |

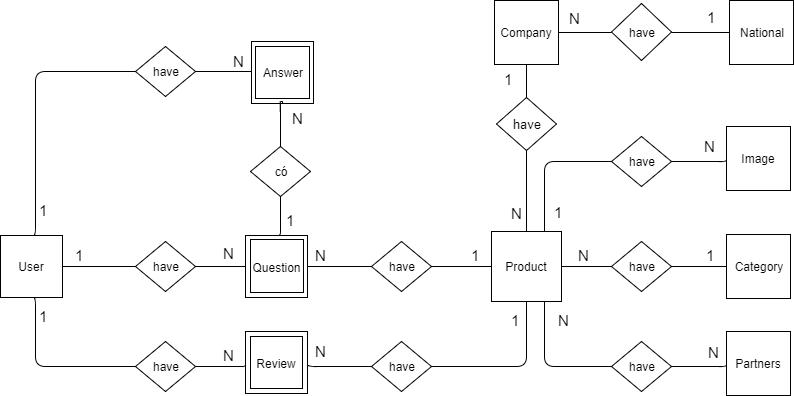
1. Ảnh (Image): nơi lưu ảnh của sản phẩm và danh mục

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của ảnh |
| IMAGE | Hình ảnh |
| STATUS | Trạng thái của hình ảnh |
| DESCRIPTION | Mô tả cho hình ảnh |
| UP\_DATE | Ngày cập nhật hình ảnh |
| CRT\_DATE | Ngày tạo hình ảnh |

1. Đại lý (Partner): nơi lưu địa chỉ phân phối sản phẩm

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của đại lý |
| NAME\_PARTNER | Tên đại lý |
| EMAIL | Email |
| PHONE | Số điện thoại |
| FAX | fax |
| ADDRESS | Địa chỉ |
| LINK | Link trang web |
| STATUS | Trạng thái của đại lý |
| DESCRIPTION | Mô tả cho đại lý |
| UP\_DATE | Ngày cập nhật đại lý |
| CRT\_DATE | Ngày tạo đại lý |

**Sơ đồ quan hệ thực thể ERD**



* Phân tích mối quan hệ các thực thể

+ Xét quan hệ giữa sản phẩm (Product) và danh mục sản phẩm (Category) là quan hệ 1-N (1 sản phẩm thuộc 1 danh mục, 1 danh mục có nhiều sản phẩm)

* Đưa khóa chính từ bảng danh mục sản phẩm (Category) sang làm khóa ngoại của bảng sản phẩm (Product)

+ Xét quan hệ giữa sản phẩm (Product) và doanh nghiệp (Company) là quan hệ 1-N (1 doanh nghiệp có nhiều sản phẩm, 1 sản phẩm thuộc về 1 doanh nghiệp)

* Đưa khóa chính từ bảng doanh nghiệp (Company) sang làm khóa ngoại của bảng sản phẩm (Product)

+ Xét quan hệ giữa doanh nghiệp (Company) và quốc gia (National) là quan hệ 1-N (1 doanh nghiệp thuộc về 1 quốc gia,1 quốc gia có nhiều doanh nghiệp)

* Đưa khóa chính từ bảng doanh nghiệp (Company) sang làm khóa ngoại của bảng sản phẩm (Product)

+ Xét quan hệ giữa sản phẩm (Product) và đại lý (Patrner) là quan hệ N-N (1 sản phẩm được cung cấp bởi nhiều đại lý và 1 đại lý sẽ cung cấp nhiều sản phẩm)

* Thêm bảng mới sản phẩm- đại lý (Product-Parent)

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của bảng Product\_Partner |
| ID\_PRODUCT | Id của sản phẩm |
| ID\_PARTNER | Id của đại lý |
| DESCRIPTION | Mô tả về bản ghi |
| STATUS | Trạng thái của bản ghi |
| UP\_DATE | Ngày cập nhật bản ghi |
| CRT\_DATE | Ngày tạo bản ghi |

+ Xét quan hệ giữa sản phẩm (product) và ảnh (image) là quan hệ 1-N (1 sản phẩm có nhiều ảnh, 1 ảnh thuộc về 1 ản phẩm)

* Đưa khóa chính từ bảng sản phẩm (product) sang làm khóa ngoại của bảng hình ảnh (image)

+ Xét quan hệ giữa bảng sản phẩm (product) và bảng câu trả lời (Answer) là quan hệ 1-N (1 sản phẩm có nhiều câu trả lời, 1 câu trả lời chỉ thuộc về 1 sản phẩm)

* Đưa khóa chính từ bảng sản phẩm (product) sang làm khóa ngoại của bảng câu trả lời (Answer)

+ Xét quan hệ giữa bảng sản phẩm (product) và bảng câu hỏi (Question) là quan hệ 1-N (1 sản phẩm có nhiều câu hỏi, 1 câu hỏi chỉ thuộc về 1 sản phẩm)

* Đưa khóa chính từ bảng sản phẩm (product) sang làm khóa ngoại của bảng câu hỏi (Question)

+ Xét quan hệ giữa bảng sản phẩm (product) và bảng bình luận (Review) là quan hệ 1-N (1 sản phẩm có nhiều bình luận, 1 bình luận chỉ thuộc về 1 sản phẩm)

* Đưa khóa chính từ bảng sản phẩm (product) sang làm khóa ngoại của bảng bình luận (Review)

+ Xét quan hệ giữa bảng câu hỏi (question) và bảng trả lời (Answer) là quan hệ 1-N (1 câu hỏi có nhiều câu trả lời, 1 câu trả lời chỉ thuộc về 1 câu hỏi)

* Đưa khóa chính từ bảng câu hỏi (question) sang làm khóa ngoại của bảng trả lời (Answer)

+ Xét quan hệ giữa bảng thành viên (User) và bảng trả lời (Answer) là quan hệ 1-N (1 thành viên có nhiều câu trả lời, 1 câu trả lời chỉ thuộc về 1 thành viên)

* Đưa khóa chính từ bảng thành viên (User) sang làm khóa ngoại của bảng trả lời (Answer)

+ Xét quan hệ giữa bảng thành viên (User) và bảng câu hỏi (question) là quan hệ 1-N (1 thành viên có nhiều câu hỏi, 1 câu hỏi chỉ thuộc về 1 thành viên)

* Đưa khóa chính từ bảng thành viên (User) sang làm khóa ngoại của bảng câu hỏi (question)

+ Xét quan hệ giữa bảng thành viên (User) và bảng bình luận (Review) là quan hệ 1-N (1 thành viên có nhiều bình luận, 1 bình luận chỉ thuộc về 1 thành viên)

* Đưa khóa chính từ bảng thành viên (User) sang làm khóa ngoại của bảng bình luận (Review)
  + - 1. Danh sách các bảng cùng thuộc tính sau khi đã được chuẩn hóa

1. **thành viên:**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID\_USER | Id của người dùng |
| USER\_GROUP | Nhóm người sử dụng (0: người dùng; 1: quản trị) |
| USER\_NAME | Tên tài khoản |
| PASSWORD | Mật khẩu người dùng |
| SEX | Giới tính |
| EMAIL | Email người dùng |
| DESCRIPTION | Mô tả ngắn về người dùng |
| AVATAR | Ảnh đại diện của người dùng |
| PHONE | Số điện thoại của người dùng |
| ADDRESS | Địa chỉ của người dùng |
| STATUS | Ngày tạo tài khoản |
| UP\_DATE | Ngày cập nhật tài khoản gần nhất |
| CRT\_DATE | Ngày tạo tài khoản |

1. **Câu hỏi (Question)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của câu hỏi |
| ID\_USER | Id của thành viên |
| ID\_PRODUCT | ID của sản phẩm |
| CONTERNT\_QUESTION | Nội dung câu hỏi |
| STATUS | Trạng thái hiển thị của bình luận |
| UP\_DATE | Ngày cập nhật câu hỏi |
| CRT\_DATE | Ngày tạo câu hỏi |

1. **Câu trả lời (Answer)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của câu trả lời |
| ID\_QUESTION | Id của sản phẩm |
| ID\_USER | Nội dung câu trả lời |
| CONTENT\_ANSWEAR | Id của thành viên |
| STATUS | Trạng thái của câu trả lời |
| UP\_DATE | Ngày cập nhật câu hỏi |
| CRT\_DATE | Ngày tạo câu hỏi |

1. **Bình luận (Review)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của bình luận |
| ID\_PRODUCT | Id của sản phẩm |
| ID\_USER | Id của thành viên |
| CONTENT\_REVIEW | Nội dung bình luận |
| APPROVAL | Đánh giá của người dùng về sản phẩm |
| STATUS | Trạng thái của bình luận |
| UP\_DATE | Ngày cập nhật bình luận |
| CRT\_DATE | Ngày tạo bình luận |

1. **Sản phẩm (Product)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của sản phẩm |
| PRODUCT\_CODE | Mã của sản phẩm |
| ID\_CATEGORY | Mã danh mục sản phẩm |
| ID\_PARTNER | Id của đại lý |
| PRODUCT\_NAME | Tên sản phẩm |
| PRICE | Giá cả |
| APPROVAL | Đánh giá tổng quát của người dùng về sản phẩm |
| TITLE | Tiêu đề về sản phẩm |
| ID\_IMAGE | Mã hình ảnh |
| DESCRIPTION | Mô tả về sản phẩm |
| STATUS | Trạng thái của sản phẩm |
| UP\_DATE | Ngày cập nhật sản phẩm |
| CRT\_DATE | Ngày tạo sản phẩm |

1. **Danh mục sản phẩm (Category)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của danh mục |
| PARENT\_ID | Danh mục cha |
| TYPE\_CATEGOTY | Mã danh mục |
| NAME\_CATEGORY | Tên danh mục |
| IMAGE | hình ảnh của danh mục |
| STATUS | Trạng thái của danh mục |
| DESCRIPTION | Mô tả cho danh mục |
| UP\_DATE | Ngày cập nhật danh mục |
| CRT\_DATE | Ngày tạo danh mục |

1. **Ảnh (Image)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của ảnh |
| ID\_PRODUCT | Id sản phẩm |
| IMAGE | Hình ảnh |
| STATUS | Trạng thái của hình ảnh |
| DESCRIPTION | Mô tả cho hình ảnh |
| UP\_DATE | Ngày cập nhật hình ảnh |
| CRT\_DATE | Ngày tạo hình ảnh |

1. **Đại lý (Partner)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của đại lý |
| NAME\_PARTNER | Tên đại lý |
| EMAIL | Email |
| PHONE | Số điện thoại |
| FAX | fax |
| ADDRESS | Địa chỉ |
| LINK | Link trang web |
| STATUS | Trạng thái của đại lý |
| DESCRIPTION | Mô tả cho đại lý |
| UP\_DATE | Ngày cập nhật đại lý |
| CRT\_DATE | Ngày tạo đại lý |

1. **Sản phẩm - Đại lý (Product\_Partner)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của bảng Product\_Partner |
| ID\_PRODUCT | Id của sản phẩm |
| ID\_PARTNER | Id của đại lý |
| DESCRIPTION | Mô tả về bản ghi |
| STATUS | Trạng thái của bản ghi |
| UP\_DATE | Ngày cập nhật bản ghi |
| CRT\_DATE | Ngày tạo bản ghi |

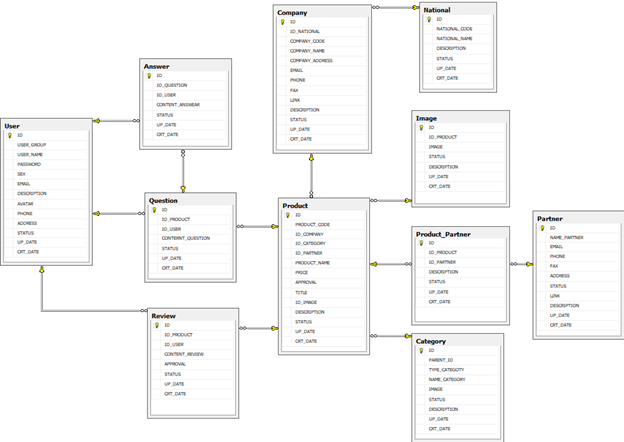
1. **Doanh nghiệp (Company)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của doanh nghiệp |
| ID\_NATIONAL | Mã quốc gia |
| COMPANY\_CODE | Mã doanh nghiệp |
| COMPANY\_NAME | Tên doanh nghiệp |
| COMPANY\_ADDRESS | Địa chỉ doanh nghiệp |
| EMAIL | Email của doanh nghiệp |
| PHONE | Số điện thoại của doanh nghiệp |
| FAX | Fax của doanh nghiệp |
| LINK | Link trang web của doanh nghiệp |
| DESCRIPTION | Mô tả về doanh nghiệp |
| STATUS | Trạng thái của của bản ghi |
| UP\_DATE | Ngày cập nhật bản ghi |
| CRT\_DATE | Ngày tạo bản ghi |

1. **Quốc gia (National)**

|  |  |
| --- | --- |
| Tên | Mô tả |
| ID | Id của quốc gia |
| NATIONAL\_CODE | Mã quốc gia |
| NATIONAL\_NAME | Tên quốc gia |
| DESCRIPTION | Mô tả về quốc gia |
| STATUS | Trạng thái của bản ghi |
| UP\_DATE | Ngày cập nhật bản ghi |
| CRT\_DATE | Ngày tạo bản ghi |

**Mô hình quan hệ giữa các bảng**



1. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN PHẦN MỀM
   1. Kết luận

Với sự hướng dẫn nhiệt tình của các thầy cô trong Khoa Công nghệ thông tin, đặc biệt là sự chỉ bảo tận tình của thầy Nguyễn Văn Nam trong thời gian làm đồ án của mình, nay em đã hoàn thành đồ án và đạt được một số kết quả sau:

* Nắm được cơ bản về một số quy trình khi tiến hành xây dựng một hệ thống trong thực tế (khảo sát thực tế, thu thập dữ liệu… triển khai, thử nghiệm).
* Tìm hiểu về phân tích và thiết kế cơ sở dữ liệu. Từ đó, áp dụng vào việc phân tích bài toán “*xây dựng ứng dụng quét mã vạch hiển thị thông tin sản phẩm*”.
* Tìm hiểu được các công nghệ phát triển platform và tìm hiểu được các công nghệ mới như Ionic 2, cordova, AngularJS 2, ASP.NET
* Xây dựng ứng dụng quét mã vạch trên mobile có các chức năng sau:
* Xem thông tin sản phẩm thông qua việc quét mã vạch
* Tìm kiếm sản phẩm theo tên hoặc theo mã barcode với giao diện dễ dàng thân thiện với người dùng

Mặc dù đã có nhiều cố gắng trong suốt quá trình làm đồ án, nhưng do thời gian có hạn và thiếu kinh nghiệm trong thực tế nên cũng không thể tránh khỏi những nhược điểm và thiếu sót. Vì vậy, em mong nhận được sự góp ý của quý thầy cô và các bạn để em có thể phát triển hệ thống này hoàn thiện hơn.

* 1. Hướng phát triển đề tài

Sau khi hoàn thành đồ án và xây dựng được ứng dụng quét mã vạch của sản phẩm, phương hướng của em sẽ cố gắng xây dựng để đưa ứng dụng với tính thực tế cao hơn, chức năng hoàn thiện hơn.

Một số hướng để em có thể phát triển hoàn thiện hơn đó là:

* + Tìm hiểu sâu hơn về ASP.NET, AngularJS 2, Ionic 2,SQL SERVER, để có thể đáp ứng nhiều hơn nữa nhu cầu của người sử dụng, phát triển và tối ưu hóa hệ thống.
  + Xây dựng website cho hệ thống và thêm chức năng mới
* Khảo sát chi tiết hơn nhu cầu của người dùng để hoàn thiện các chức năng của hệ thống.

**TÀI LIỆU THAM KHẢO.**

Các tài liệu tham khảo online:

1. <http://www.w3schools.com/jsref/jsref_length_array.asp><http://php.net/>
2. [http://www.google.com.vn](http://www.google.com.vn/imghp?hl=vi&tab=wi)
3. <https://github.com/>
4. <https://ionicframework.com/docs>
5. <https://cordova.apache.org>
6. <http://www.anthinh.com/tin-tuc/chi-tiet/nhung-dieu-can-biet-ve-ma-vach-66/>
7. <https://vi.wikipedia.org/wiki/M%C3%A3_v%E1%BA%A1ch>
8. <http://baohothuonghieu.com>
9. <https://stackoverflow.com/>

Giáo trình

1. Vũ Tiến Thái, *Bài giảng Phân tích thiết kế hệ thống thông tin.*

2. Nguyễn Văn Lệ, *Bài giảng Công nghệ phần mềm.*

3. Phạm Thị Hoàng Nhung, *Cơ sở dữ liệu I.*

4. Phạm Thị Hoàng Nhung, *Cơ sở dữ liệu II.*

5. Phạm Thị Hoàng Nhung, *Hệ quản trị cơ sở dữ liệu SQL Server 2000.*

6. Đặng Văn Đức, Nguyễn Thị Phương Trà, *Giao diện Người – Máy.*