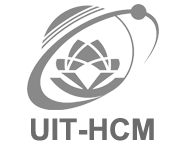
****

BỘ GIÁO DỤC VÀO ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

ĐHQGTPHCM

**BÁO CÁO ĐỒ ÁN CUỐI KÌ**

MÔN PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ

**CHỦ ĐỀ: Phát Triển Module NEWS**

**Cho Mã Nguồn Mở**



**Giảng viên hướng dẫn:** [**Nguyễn Công Hoan**](https://courses.uit.edu.vn/user/view.php?id=378&course=1)

**Danh sách thành viên:**

Ngô Trung Hiếu – 16520396

Đặng Tấn Khải – 17520603

Nguyễn Thanh Thọ – 17521093

**MỤC LỤC**

1. **Tổng quan về lĩnh vực phần mềm**
2. Khái niệm CMS. 4
3. Các chức năng chính 4
4. Cách hoạt động của CMS 4
5. Lý do chọn CMS 5
6. Phân loại CMS 6
7. Các sản phẩm phổ biến hiện nay 6
8. **Phần mềm mã nguồn mở Magento**
9. Lịch sử phát triển 8
10. Các tính năng 9
11. Đặc điểm nổi bật so với với các phần mềm cùng loại 9
12. **Kiến trúc tổng thể của phần mềm**
    * 1. Mô hình kiến trúc : MVC 9
      2. Mô hình kiến trúc tổng thể 12
13. **Mô hình thiết kế Database**
14. Tổng thể mô hình thiết kế Database 14
15. Chi tiết mô hình thiết kế Database 17
16. Mô hình EAV 23
17. **Môi trường cài đặt và xử lý của Magento**
18. Cấu hình tối thiểu 24
19. Yêu cầu phần mềm và các thư viện liên quan 24
20. **Cài đặt**

#### Download 25

#### Cài đặt Magento 2 26

1. **Hiện thực hóa Module Tin tức**
2. Cấu trúc Database Module tin tức 34
3. Sơ đồ phân tích DFD 36
4. Tích hợp module Tin tức 44
5. Giao diện và Tính năng 52
6. **Kết đoạn và hướng phát triển**
7. **Tổng quan về lĩnh vực phần mềm**

## Khái niệm CMS

**CMS (viết tắt của *Content Management System*) là hệ quản trị nội dung của trang web**, điều khiển tất cả hoạt động của trang web**,** có chức năng quản lý các tiến trình tạo và chỉnh sửa các nội dung số mà không cần phải biết lập trình, hoặc bạn chỉ cần có một số những kiến thức cơ bản là đều có thể quản lý được. Thông thường CMS sẽ có một giao diện trực quan để người dùng thao tác mọi phần liên quan của website. Đặc biệt, với các **CMS nâng cao**, người dùng còn có thể chỉnh sửa, phân loại danh mục, thậm chí thay đổi hiển thị giao diện và nhiều thao tác phức tạp khác trên website.

## Các chức năng chính

CMS có vai trò quan trọng trong việc điều khiển và vận hành trang web, đồng thời, CMS có vai trò quan trọng trong việc điều khiển và vận hành trang web.

Các chức năng cơ bản như sau:

* Tạo, lưu trữ các nội dung trên trang web
* Chỉnh sửa, thêm, bớt nội dung
* Chuyển và chia sẻ nội dung
* Quản lí và phân quyền người dùng

Do đó, các doanh nghiệp cần cài đặt CMS trong website để quá trình vận hành doanh nghiệp được thực hiện thuận lợi hơn. Đồng thời góp phần không nhỏ trong việc khẳng định vị thế thương hiệu.

## Cách hoạt động của CMS

Nếu không sử dụng hệ quản trị cơ sở dữ liệu này thì mọi thứ bạn triển khai trên web lên server sẽ hoàn toàn thủ công. Khi xây dựng một trang web, người dùng phải thực hiện 2 bước:

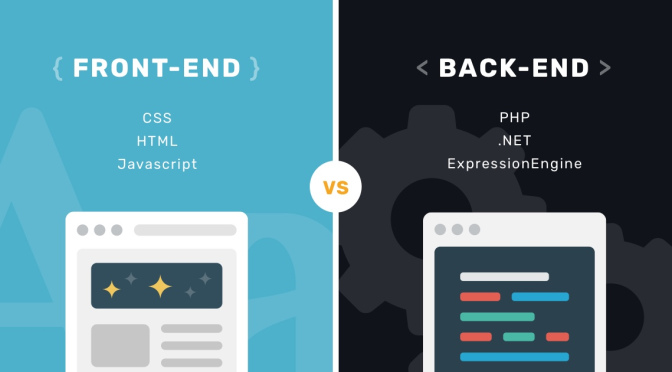
* **Xác lập giao diện bên ngoài (front-end)**
* **Quản lý hệ thống bên trong (back-end)**

Phần **front-end**, khi khách truy cập vào thì sẽ nhìn thấy bài viết, hình ảnh, video trên website. Chức năng này được hình thành từ rất nhiều các ngôn ngữ lập trình khác nhau như: CSS, JavaScript,…

Phần **back-end** sẽ bao gồm các dữ liệu và chức năng sử dụng trong website. Chức năng này được hình thành từ rất nhiều các ngôn ngữ lập trình khác nhau như: PHP, Python, JavaScript, …

Khi bạn sử dụng CMS thì tất cả những phần này đều được thể hiện bằng một giao diện trực quan đã tích hợp các phần ngôn ngữ bên trong. Bạn chỉ cần thao tác click chuột và sử dụng đơn giản và tạo các danh mục, menu mà mình muốn trong thời gian ngắn.

Ngoài ra, CMS sẽ giúp bạn không phải truy cập trực tiếp vào server của mình để chèn nội dung vào mà toàn bộ chỉ thao tác trên giao diện được hình thành. Đặc biệt, có một điều mà CMS mang lại cho người sử dụng nữa đó là chi phí xây dựng CMS tiết kiệm hơn rất nhiều so với việc triển khai từ đầu và làm thủ công.



## Lý do chọn CMS

CMS có vai trò quan trọng trong việc mang lại hiệu quả cho các chiến dịch marketing

* *Giúp website trở thành kênh marketing hiệu quả*

Có thể nói rằng, CMS có vai trò quan trọng trong việc đưa website trở thành kênh marketing hiệu quả. Thông qua hệ thống này, doanh nghiệp có thể dễ dàng thực hiện các hình thức tiếp thị như email marketing, content,… một cách hiệu quả nhất, thu hút được đông đảo khách hàng tiềm năng cũng như quảng bá thương hiệu ra thị trường.

* *Giúp cập nhật nội dung*

Ngày nay, content marketing là hình thức tiếp thị phổ biến và mang lại hiệu quả cao. Vì thế, sự xuất hiện của hệ thống CMS đã góp phần không nhỏ trong việc mang lại hiệu quả cho các chiến dịch content marketing.

* *Với CMS, doanh nghiệp có thể tạo, chỉnh sửa cũng như xuất bản nội dung một cách dễ dàng, nhanh chóng*

Doanh nghiệp có thể tạo, chỉnh sửa cũng như xuất bản nội dung một cách dễ dàng, nhanh chóng ở bất cứ đâu, bất kỳ khi nào chỉ với một vài thao tác nhỏ. Ngoài ra, hệ thống này còn có chức năng lưu nháp nội dung chưa hoàn thành. CMS hỗ trợ khả năng hạn chế truy cập, tự động xuất bản, thêm các ứng dụng video, hình ảnh,… Nhờ vậy mà doanh nghiệp có thể dễ dàng kiểm soát cũng như duy trì nội dung tiếp thị của mình dưới mọi hình thức.

### *Với CMS*, Đảm bảo, duy trì tính nhất quán cho doanh nghiệp

Với CMS, việc chỉnh sửa và xây dựng nội dung trang web được thực hiện một cách dễ dàng. Còn cho phép nhiều người cùng chỉnh sửa trên cùng một giao diện. Điều này không chỉ giúp cho hệ thống quản lý của doanh nghiệp được duy trì nhất quán hơn mà còn tiết kiệm không ít chi phí, thời gian và nhân lực cho doanh nghiệp.

## Phân loại CMS

* **CMS Open Source** (**CMS mã nguồn mở):** là một hệ thống quản trị khá phổ biến và dễ dàng sử dụng. **Ưu điểm của CMS** này có đầy đủ những tính năng cần thiết như quản lý trang, bài viết, tài khoản, hay các liên kết, tag, cấu hình,… ngay sau khi cài đặt lên máy chủ.

### CMS tự code: xây dựng hay dùng Framework: với **CMS** này, khi cài đặt đòi hỏi người dùng phải tự code lại toàn bộ. Điều này sẽ khiến cho việc cài đặt trở nên khó khăn, phức tạp hơn nhưng bù lại, người dùng có thể tùy chọn giao diện thiết kế ưng ý và phù hợp nhất.

* **CMS tính phí:** người dùng phải bỏ tiền cho việc mua bản quyền. Và vì thế, đơn vị cung cấp sẽ tiến hành tất cả mọi thao tác như sửa lỗi, nâng cấp hay vận hành hệ thống,… Đặc biệt, hệ thống này hoạt động ổn định hơn với nhiều tính năng riêng biệt hấp dẫn hơn.

## Các sản phẩm phổ biến hiện nay

Có nhiều CMS khác nhau để người dùng tùy ý lựa chọn.

### **CMS WordPress**

**WordPress** là CMS thông dụng nhất hiện nay. **WordPress** được xây dựng trên hệ thống ngôn ngữ PHP và được sử dụng phổ biến trong các lĩnh vực bất động sản, thương mại điện tử quy mô nhỏ, hay các trang web bán hàng chuyên một ngành hàng. **WordPress** còn được đính kèm với các gói dịch vụ hosting gọi là [**Hosting WordPress**](https://www.matbao.net/hosting/cloud-wordpress-hosting.html?utm_source=matbao&utm_medium=wiki-post&utm_campaign=cms-la-gi#uu-diem-cloud-wordpress-hosting). Đây là những hosting được tối ưu chỉ dành cho hệ quản trị nội dung **CMS WordPress**. Khi sử dụng các gói hosting này, website WordPress của bạn sẽ hoạt động tối ưu nhất.

### **CMS Joomla**

**Joomla** là **CMS phổ biến thứ hai sau WordPress**. Với ưu điểm vượt trội là dễ dàng sử dụng, không đòi hỏi người dùng phải có trình độ chuyên môn cao. Ngày nay, Joomla đang bị mất dần ưu thế nhưng tính thuận tiện của CMS này là hoàn toàn không thể phủ nhận.

Những đặc tính cơ bản của **Joomla** bao gồm: bộ đệm trang (**page caching**) để tăng tốc độ hiển thị, lập chỉ mục, đọc tin RSS (RSS feeds), trang dùng để in, bản tin nhanh, blog, diễn đàn, bình chọn, lịch biểu, tìm kiếm trong Site và hỗ trợ đa ngôn ngữ.

### **Image result for Drupal logo"CMS Drupal**

### **Drupal** được xây dựng trên hệ thống ngôn ngữ PHP và là sự lựa chọn lý tưởng cho các trang web có quy mô trung bình hoặc lớn. Bởi sự linh hoạt, khả năng tùy chỉnh tốt cũng như hệ thống plug-in đa dạng của nó. Thông thường, các trang web trên lĩnh vực tin tức, thương mại điện tử, các trang web theo yêu cầu,…. sẽ sử dụng Drupal trong việc tạo lập và quản lý website của mình.

### Image result for magento logo"

### **CMS Magento**

Tương tự như **WordPress** hay **Drupal**, **Magento** cũng là **CMS** sử dụng ngôn ngữ PHP làm nền tảng và lưu trữ dữ liệu tại MySQL. **Ưu điểm của CMS** này là sử dụng mã nguồn mở, do đó người dùng có thể tự chọn giao diện hay cấu hình tùy theo ý thích.

### **Image result for Opencart logo"CMS Opencart**

**Opencart** là CMS được xây dựng trên nền tảng ngôn ngữ PHP. **CMS Opencart** sử dụng mã nguồn mở. Bên cạnh đó nó còn là sự lựa chọn tối ưu cho các hệ thống bán hàng online.

### **Image result for Typo3 logo"CMS Typo3**

**Typo3** là CMS được viết bằng PHP. Typo3 có thể chạy trên một số máy chủ web, chẳng hạn như Apache hoặc IIS, trên đầu nhiều hệ điều hành, trong đó có Linux, Microsoft Windows, FreeBSD, macOS,…

### **CMS Dotclear**

**Dotclear** là một ứng dụng xuất bản blog nguồn mở được phân phối theo GNU GPLv2. Được phát triển ban đầu bởi Olivier Meunier từ năm 2002, Dotclear hiện đã thu hút được một nhóm các nhà phát triển vững chắc và được ứng dụng rộng rãi tại Pháp.

Ngoài ra, còn có một số CMS khác bao gồm:

* *Chamilo*
* *DotNetNuke (ASP)*
* *Kentiko (ASP)*
* *Liferay (JSP)*
* *Mambo (PHP)*
* *NukeViet (PHP)*
* *PHP-Nuke (PHP)*
* *Rainbow (ASP)*
* *Xoops (PHP)*
* *Cake*
* *Shopify,*
* *…*

1. **Phần mềm mã nguồn mở Magento**
2. **Lịch sử phát triển**

Varien - California, Mỹ - đã phát triển Magento bởi vì công ty này đã nhận ra nhiều hạn chế của nền tảng thương mại điện tử OSCommerce. Nền tảng này thiếu tính ổn định, các tính năng độc đáo và tính linh hoạt do đó họ muốn phát triển một nền tảng thương mại điện tử tốt hơn.

Năm 2007**, Varien** bắt đầu xây dựng nền tảng thương mại điện tử mới mang tên “Bento”, xây dựng trên nền tảng của **Zend Framework**. Nhưng nó đã được đăng ký nhãn hiệu bởi một công ty khác. Do đó, họ thống nhất nó với Mage để khởi động Magento. Ngày 31/8/2008, Varien chính thức phát hành phiên bản beta Magento cộng đồng.

Magento trở nên phổ biến hơn sau năm 2009. Nó thậm chí còn phổ biến hơn chính Thương mại Điện tử. Vào ngày 15 tháng 4 năm 2009, Magento Enterprise Edition, được xuất bản từ Magento Community Edition, đã được phát hành. Năm 2010, Magento cho Điện thoại di động đã được phát triển

Tháng 2 năm 2011, eBay tuyên bố đã đầu tư vào Magento và sở hữu 49% cổ phần sở hữu của công ty. Và vào tháng 6 cuối năm đó, eBay đã mua lại 100% Magento, với kế hoạch hợp nhất nó với những tính năng mới. Nền tảng này sẽ trở thành một danh mục phát triển mới của X.Commerce.

3/11/2015, **Magento** lại tách khỏi eBay và trở thành một công ty độc lập dưới sự đầu tư của quỹ **Permira**. Magento 2.0 đã được phát hành vào cuối năm 2015 với mục đích cung cấp những phương thức mới để nâng cao sự tương tác của người dùng, điều hướng thông suốt, tỷ lệ chuyển đổi và tạo ra doanh thu tổng thể. Hiện nay, **Magento 1.0** và **Magento 2.0** đều đang được phát triển đồng thời để cung cấp cho người dùng nhiều lựa chọn hơn.

Theo Aheadworks trong 2016, có tới 29,8% **website thương mại điện tử** được xây dựng từ **Magento**. Rất nhiều tên tuổi lớn đã sử dụng **Magento** để tạo **website thương mại điện tử** như: *Fraport AG, Dyson, MotherCare, Orsay, Tilbury, Poundland*,…

Tính đến 2018, trong tổng số 457.964 website thương mại điện tử thì có hơn 342.218 trang web đã hoặc đang sử dụng Magento. Nhóm nước phát triển có xu hướng sử dụng Magento nhiều hơn so với những quốc gia khác. Trong đó, Mỹ, Anh và Úc là những nước dùng Magento nhiều nhất.

1. **Các tính năng**

Magento được đánh giá là có đầy đủ các chức năng và công cụ để phát triển một trang website thương mại điện tử thành công.

1. **Tính năng cơ bản**

* Quản lý sản phẩm: với nhiều hình ảnh, tùy chọn nhận xét đánh giá sản phẩm, danh mục sản phẩm ưa thích, inventory.
* Quản lý category: dễ dàng tìm kiếm và chọn lọc sản phẩm theo category.
* Quản lý kho hàng: quản lý sản phẩm tồn kho, nhập xuất…
* Quản lý User: tạo tài khoản, tình trạng tài khoản, lịch sử giao dịch, danh mục ưa thích, địa chỉ, giỏ hàng…
* Dịch vụ khách hàng: tăng cường các tính năng, hình thức liên hệ khách hàng, theo dõi toàn diện, dịch vụ email.
* Quản lý đơn hàng.
* Thanh toán: nhiều phương thức thanh toán như thẻ tín dụng, Paypal, Authorize.net, Google –Checkout, hổ trợ các module thanh toán bên ngoài như CyberSource, ePay, eWAY,…
* Công nghệ tìm kiếm: nhanh chóng, thân thiện, trổ trợ tìm kiếm trên Google SiteMap.
* Hổ trợ đa ngôn ngữ, tiền tệ.
* Công cụ khuyên mãi và tiếp thị: phiếu giảm giá, khuyến mãi và nhiều tùy chọn.
* Phân tích và báo cáo: tích hợp với dịch vụ Google Analytics và cung cấp nhiều báo cáo.

1. **Tính năng nổi bật**

* Magento đã có những bước đột phá, vượt qua khỏi những hạn chế của mã nguồn mở và có được các tính năng hữu ích nhất trong phiên bản miễn phí.
* Wishlists: chia sẽ, lưu trữ sản phẩm, danh sách sản phẩm cho mỗi user.
* Multi store/mall feature: nhiều cửa hàng, kho hàng, điểm bán hàng.
* Email lists: danh sách Email, lọc, sắp xếp và lưu trữ, phục vụ cho việc marketing.
* Compare products: so sánh, đối chiếu sản phẩm.
* Bundle/Grouped/Digital products & more: gói, nhóm sản phẩm, thuộc tính riêng biệt cho từng sản phẩm.
* CMS system for static pages: trang tĩnh giúp thiết kế độc lập với hệ thống sản phẩm, tạo ra sự tùy biến cho từng chiến lược bán hàng.
* SEO performance: tối ưu sẵn có với các công cụ tìm kiếm, tích hợp sẵn SEO.
* Advanced searching: tìm kiếm nâng cao và tùy chỉnh tìm kiếm, giúp bạn điều hướng được người sử dụng.
* Layered/Filter navigation: tự tạo ra các bộ lọc thông minh, tùy chỉnh bộ lọc theo từng tình huống cụ thể giúp cho việc thể hiện sản phẩm, tìm kiếm sản phẩm một cách thông minh nhất.

1. **Đặc điểm nổi bật so với với các phần mềm cùng loại**

#### **Đồng bộ**

Việc làm thế nào để đồng bộ hóa dữ liệu giữa phần mềm bạn đã có với một website mới toanh? Magento đặc biệt ưu ái phát triển hệ thống này, chỉ sau một vài thao tác nhỏ ở dạng addon là bạn có ngay kết quả theo yêu cầu mong muốn. Đây là điểm khác biệt với các hệ thống mã nguồn mở khác và nó lại bị các doanh nghiệp Việt Nam bỏ ngỏ.

1. ***Cấu trúc***

Magento được thiết kế rất tối ưu ngay từ phiên bản đầu tiên, và có sự phân biệt rõ ràng giữa code “core” và code “community”. Điều này có nghĩa là bạn có thể cài đặt phần mở rộng khác bằng các modules addon mà không sợ bị ảnh hưởng đến nhân Magento khi nâng cấp trong tương lai. Đây là lý do tuyệt vời đã được cộng đồng chấp nhận, nhiều người sẽ dễ dàng chuyển đổi từ một mã nguồn mở trước đó qua việc sử dụng Magento mà không gặp trở ngại gì.

#### **Cộng đồng**

Magento có một cộng đồng rất tích cực, cả người sử dụng và nhà phát triển, nhiều người trong số họ là những cuyên gia Magento. Đã có một số lượng lớn các phần mở rộng đã được xây dựng cho Magento, cả hai điều miễn phí và thương mại, cho phép nâng cấp cửa hàng của bạn trong nhiều cách với rất nhiệu dịch vụ bên thứ 3 và từ nhà phát triển hệ thống.

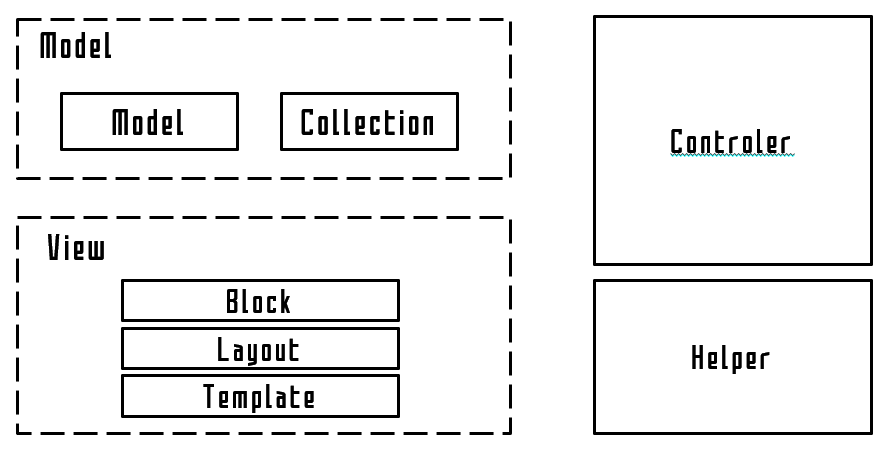
1. **Kiến trúc tổng thể của phần mềm**
2. **Mô hình kiến trúc : MVC**

Mô hình MVC (Model – View – Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nó giúp cho các developer tách ứng dụng của họ ra 3 thành phần khác nhau Model, View và Controller. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

1. ***Cấu trúc mô hình kiến trúc MVC***

Mô hình MVC được chia làm 3 lớp xử lý gồm Model – View – Controller :

* **Model:** là nơi chứa những nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (mysql, mssql… ); nó sẽ bao gồm các class/function xử lý nhiều nghiệp vụ như kết nối database, truy vấn dữ liệu, thêm – xóa – sửa dữ liệu…
* **View:** là nới chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.
* **Controller:** là nới tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View.



### Image result for MVC***Sự tương tác giữa các thành phần***

* **Controller – View** sẽ lấy những hình ảnh, nút bấm…hoặc hiển thị dữ liệu được trả ra từ Controller để người dùng có thể quan sát và thao tác. Trong sự tương tác này cũng có thể không có dữ liệu được lấy từ Model và khi đó nó chỉ chịu trách nhiệm hiển thị đơn thuần như hình ảnh, nút bấm…
* **Controller – Model** là luồng xử lý khi controller tiếp nhận yêu cầu và các tham số đầu vào từ người dùng, controller sẽ sử dụng các lớp/hàm trong Model cần thiết để lấy ra những dữ liệu chính xác.
* **View – Model** có thể tương tác với nhau mà không qua Controller, nó chỉ đảm nhận hiển thị dữ liệu chứ không phải qua bất kỳ xử lý nghiệp vụ logics nào. Nó giống như các vùng dữ liệu hiển thị tĩnh trên các website như block slidebar…

1. ***Ưu điểm và nhược điểm***

**Ưu điểm**

* Trình tự xử lý rất rõ ràng
* Mô hình MVC quy hoạch các class/function vào các thành phần riêng biêt **Controller - Model - View**, việc đó làm cho quá trình phát triển - quản lý - vận hành - bảo trì web diễn ra thuận lợi hơn, tạo ra được các chức năng chuyên biệt hoá đồng thời kiểm soát được luồng xử lý.
* Tạo mô hình chuẩn cho dự án, khi người có chuyên môn ngoài dự án tiếp cận với dự án dễ dàng hơn.
* Mô hình đơn giản, dễ hiểu, xử lý những nghiệp vụ đơn giản, và dễ dàng triển khai với các dự án nhỏ.

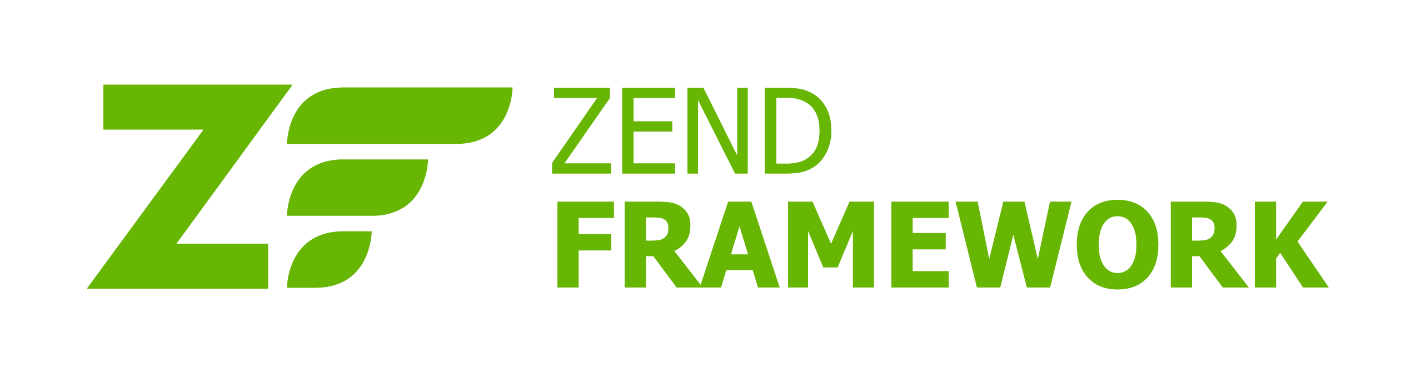
**Nhược điểm**

* Đối với các dự án có tính phức tạp cao thì mô hình MVC trở nên không khả dụng.

1. ***Zend Framework***

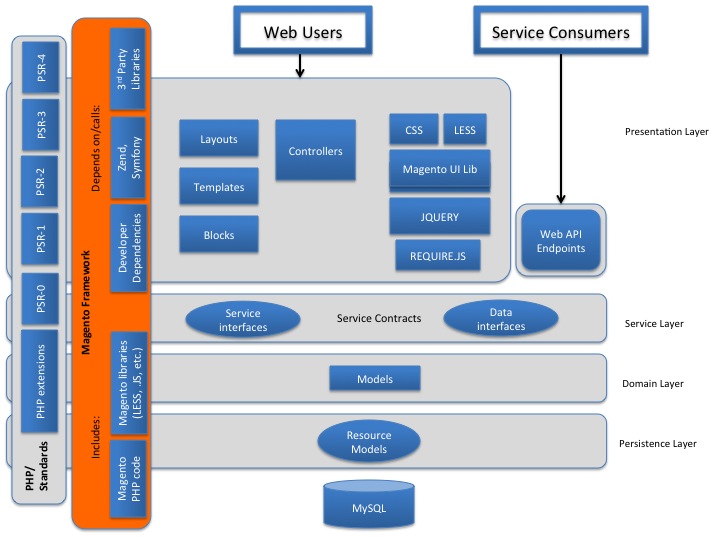
Bộ thư viện (object\_oriented library và base code) của **Zend Framework:**

* **Model**: gồm class PHP Model, là 1 phần của module logic MVC. Dùng để viết các câu lệnh truy vấn trực tiếp với cơ sở dữ liệu.
* **Block**: gồm class PHP View, là 1 phần của module logic MVC. Là nơi dùng để load dữ liệu, điều chỉnh dữ liệu từ database trước khi cho hiển thị dữ liệu ra template.
* **Controller**: gồm class PHP Controller, là 1 phần của module logic MVC. Nhận yêu cầu từ phía người dùng từ http sau đó sẽ chuyển yêu cầu tới các lớp xử lý.
* etc: gồm các file xml dùng để config cho module. Tùy theo mỗi module mà có những file xml khác nhau:
* Config.xml: dùng để khai báo model, helper, block…
* System.xml: cầu hình tạo ra một số field
* Adminhtml.xml: dùng để cấu hình hiển thị trên menu chính của phần quản trị.
* **Setup**: Bao gồm class cho cấu trúc dữ liệu và cài thiết lập dữ liệu. Dữ liệu này sẽ được truyền vào khi cài đặt hoặc nâng cấp



1. **Mô hình kiến trúc tổng thể**

Khi tương tác với giao diện web của Magento, thì bạn đang tương tác với **layer presentation**



**Layer Presentation:**

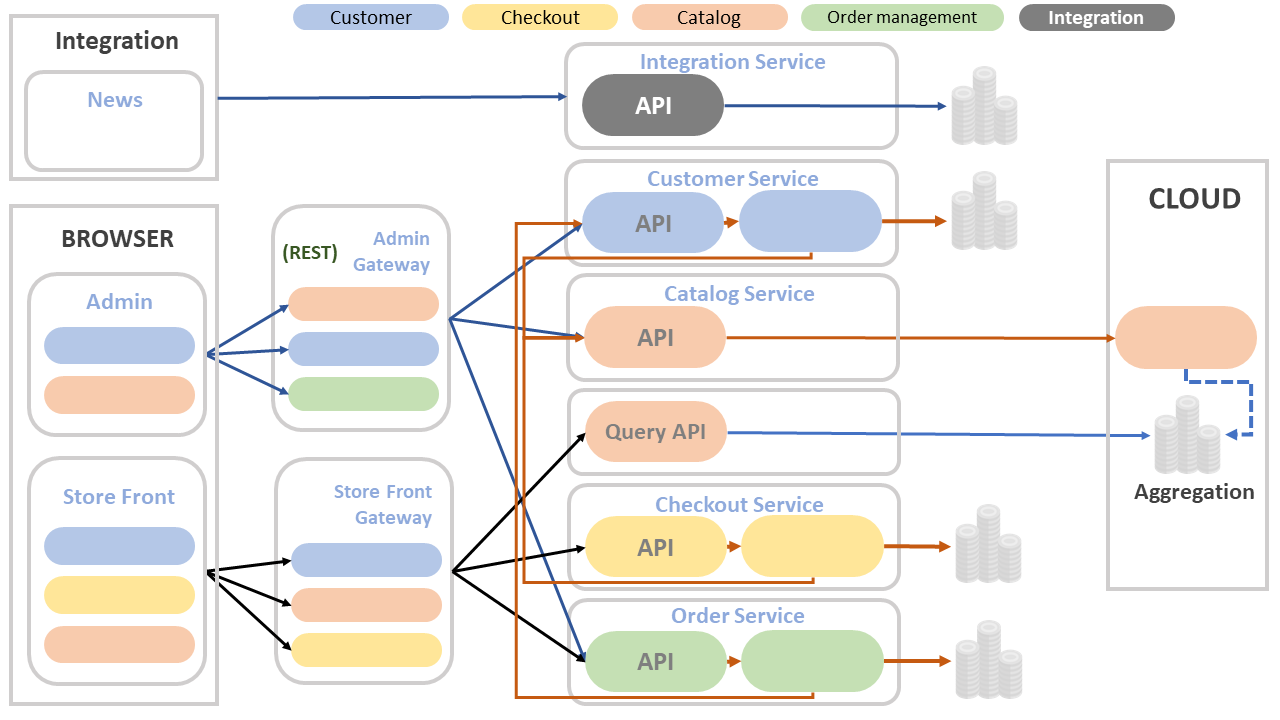
* Nằm trên cùng trong số 4 layer (presentation, service, domain, và persistence layers). Presentation layer chứa cả 2 yếu tố View và Controller
* Bao gồm cả 2 yếu tố View và Controllers, xử lý các lệnh và gửi đến user interface. Code Presentation đảm nhiệm xử lý các tương tác người dùng và sản phẩm, giao diện web. Developer có thể tùy biến giao diện người dùng qua file HTML, CSS, và PHTML để tùy chỉnh các yếu tố trong layer presentation

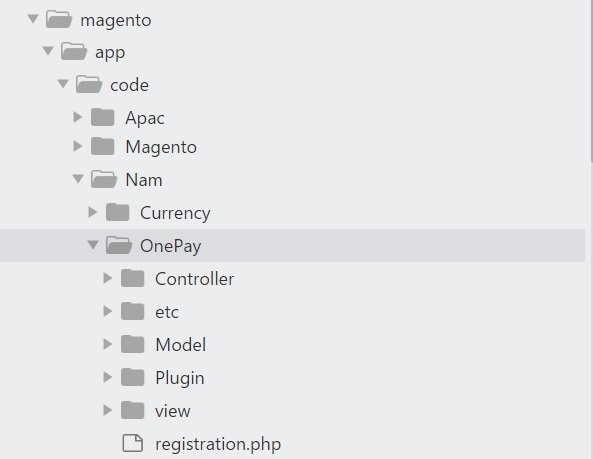
**Service Layer:**

* Nằm dưới Presentation Layer và Domain Layer
* Chứa các Service Contract, xác định cách implements hoạt động
* Dễ dàng truy cập đến
* Cung cấp API ổn định cho các module khác code của REST/SOAP API framework. Đồng thời có thể liên kết Service Contract vào Web Service API trong file hệ thống – không yêu cầu phải code thêm

**Domain Layer:**

* Chứa Bussiness Logic Layer của module Magento. Layer này không chứa tài nguyên hay thông tin dữ liệu.
* Chức năng chính là thực hiện các yêu cầu thêm, sửa, xóa, và cập nhập. Chứa SQL code để thực hiện các yêu cầu này. Thực hiện Bussiness Logic



[](http://namluu.com/wp-content/uploads/2018/07/m2.jpg)

Với Magento 2, 1 module được đóng gói tập trung tại 1 nơi, bao gồm tất cả các chức năng: khai báo, xử lý chức năng, giao diện và dịch thuật ngôn ngữ

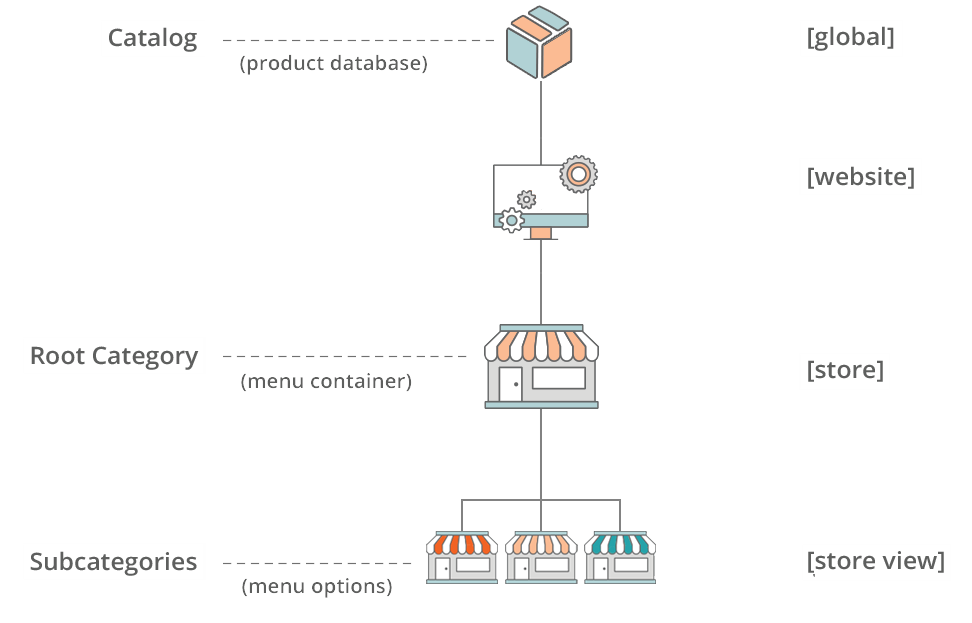
1. **Mô hình thiết kế Database**
2. **Tổng thể mô hình thiết kế Database**
3. Mô hình phân cấp

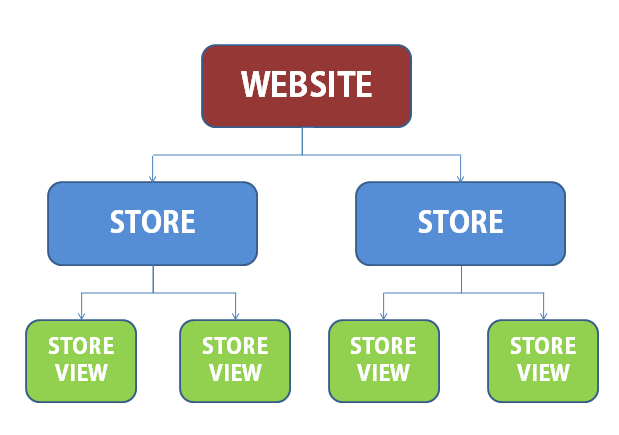
Hệ thống phân cấp của magento 2 giúp cải thiện trải nghiệm mua sắm của khách hàng. Với hệ thống này, bạn có thể dễ dàng sắp xếp nội dung trang, giao diện trang, menu, … Magento cho phép người dùng tạo nhiều website và cửa hàng chỉ với 1 lần cài đặt.

Trong mô hình phân lớp của Magento, chia thành 4 phần “**Global**”, “**Website**”, “**Store**”, “**Store View**”:

* **Global** là phần cài đặt chung Magento. Có thể sử dụng Global Domain để tạo ra Website
* **Stores** là thành phần con của website. Mỗi Website có ít nhất một hoặc nhiều Store để hiển thị Catalog
* **Store views** là một kiểu hiển thị khác trong front-end. Thường được dùng để hiển thị đa ngôn ngữ

Hệ thống phân cấp giúp chúng ta thiết lập nhiều Website, mỗi website có thể có nhiều Store, mỗi Store có thể có nhiều Store View

[](https://docs.magento.com/m1/ce/user_guide/Resources/Images/scope-catalog.png)



Những thứ CÓ THỂ và KHÔNG THỂ làm vs mô hình phân cấp

**Multi-store view:**

* Có thể tạo nhiều theme khác nhau cho từng Store View
* Có thể điều chỉnh quy ước về phương thức thanh thoán / vận chuyểm
* Có thể dễ dàng thay đổi tiêu đề và nội dung của sản phẩm
* Có thể thấy đơn hàng đã qua xử lý từ Store View
* Có thể dễ dàng quản lý đơn hàng dựa trên Store View
* Không thể thay đổi cấu trúc Category/Tax giữa các Store View
* Không thể thấy báo các từ Store View

**Multi-store:**

* Có thể quản lý hoàn toàn Catalog,Category giữa các Store
* Không thể thay đổi Cấu trúc Tax giữa các store
* Không thể quản lý kho hàng không nằm trong Store
* Không thể quản lý cấu hình giữa các Srore

**Multi website:**

* Có thể làm hầu hết tất cả mọi thứ ở phần này

**Lợi ích**

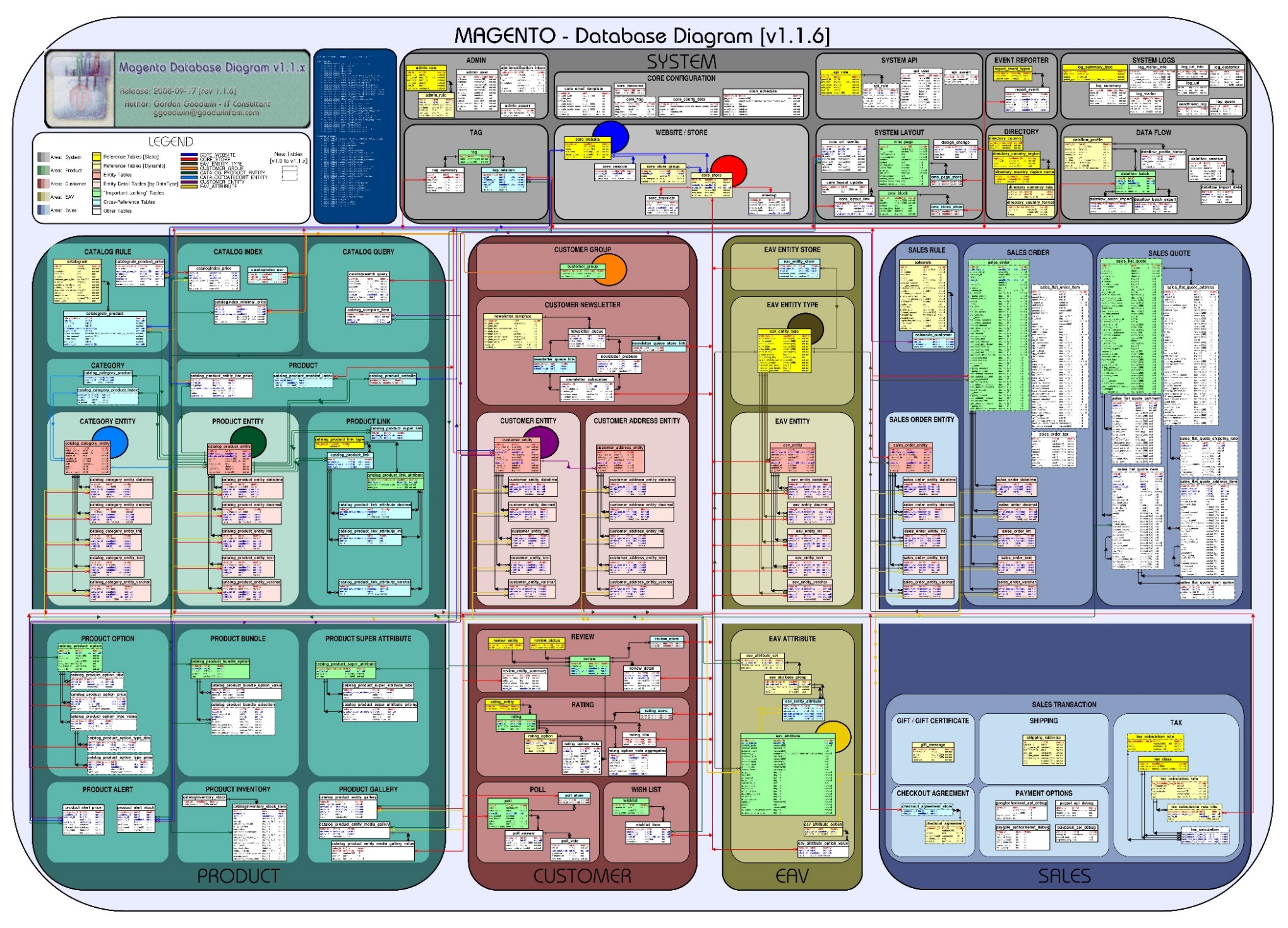
Hệ thống phâ tầng của Magento là hệ thống có thể mở rộng, linh hoạt, và than thiện với người dung. Giúp người bán hàng có thể dễ dàng theo dõi, quản lý sự phát triển của Store chỉ với 1 lần cài đặt Magento

Trải nghiệm của khách hàng sẽ được cải thiện đáng kể, Khách hàng sẽ dễ dàng và nhanh chóng truy cập tới trang bán hang

Việc quản lý cửa hàng sẽ dễ dàng hơn rất nhiều, Có thể kinh doanh nhiều loại hàng hóa mà ko lo về số lượng, tính đa dạng, thị trường

* + **Multi stores**: giúp việc buôn bán thuận lợi hơn, và dễ dàng hơn, Giúp người bán có thể bán nhiều loại hàng hóa đa dạng hơn, giúp phân loại sản phẩm dễ dàng hơn
* + **Multi-store views:** thuận lợi hơn cho việc buôn bán trong và ngoài nước. Buôn bán xuyên biên giới dẽ dàng hơn với ngôn ngữ và đơn vị tiền tệ trong nước
* + **Multi website**: tăng khả năng thay đổi mọi khia cạnh của trải nghiệm khách hàng
* + Thuận tiện hơn khi chia sẽ data sản phẩm và khách hàng, thuận tiện cho việc báo cáo doanh thu / thông tin

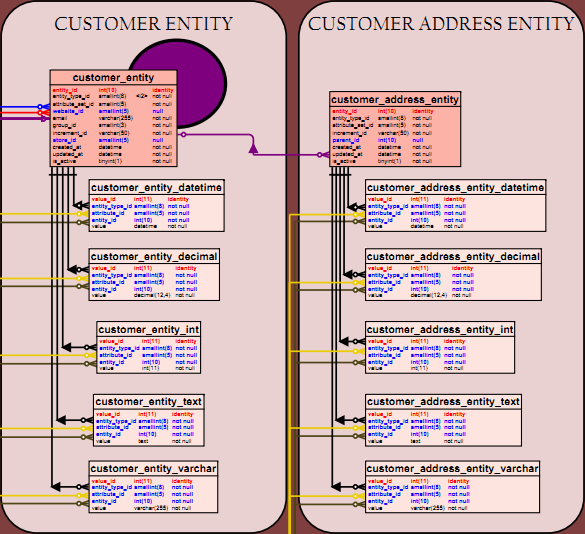
1. **Chi tiết mô hình thiết kế Database**

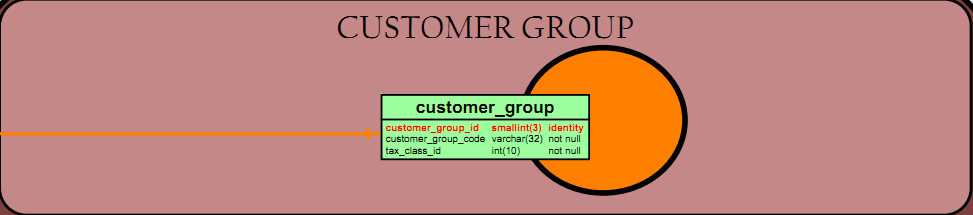


*Những Module quan trọng của Magento:* ***Custormer, Catalog, Sales***

* 1. **Customer**

Dữ liệu Custormer của Magento được lưu trũ trong model EAV







Bảng customer\_entity là nơi lưu giữ thông tin của khách hàng:

* entity\_id: Khóa chính
* email: Email khách hàng
* create\_at, update\_at: Thời gian tạo, và lần cuối update
* is\_active: Trạng thái hoạt động
* website\_id, store\_id, group\_id: các key liên kết tới website, store, group
* entity\_type\_id: Khóa chính của eav\_entity
* attribute\_set\_id: Khóa chính của của bảng eav\_attribute\_set

Những bảng như customer\_entity\_datetime, customer\_entity\_decimal, customer\_entity\_int, customer\_entity\_text, customer\_entity\_varchar chứa các thông tin của các thuộc tính để tiện cho việc phân loại.

Trong bảng này có một số thuộc tính cần quan tâm như:

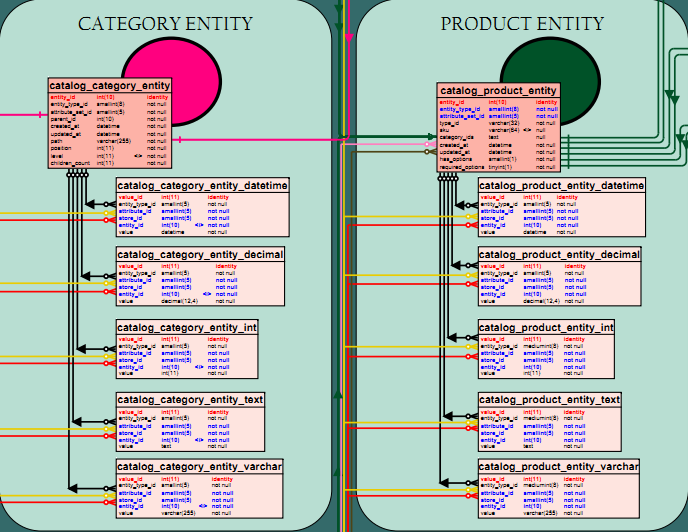
* entity\_type\_id: Khóa chính của bảng eav\_entity\_type
* attribute\_id: Khóa chính của bảng eav\_attribute.
* entity\_id: Khóa chính của bảng customer\_entity

Bảng địa chỉ của customer có cách lưu trữ tương tự với customer\_address\_entity, bên cạnh đó parent\_id sẽ tạo link tới bảng customer\_entity.

* 1. **Product**

Cách tốt nhất để diễn tả module catalog trong Magento là sử dụng mô hình E.A.V.

Dữ liệu của module này được chia thành 2 phần**: category** và **product**



Cách lưu trữ của table này tương tự như bảng customer: bảng chính là catalog\_category\_entity, bảng phụ là một trong những tiền tố như catalog\_category. Parent\_id chứa thông tin về quan hệ giữa các categories (parent category).

Bảng catalog\_category\_product dung để biểu diễn mối quan hệ giữa product và category.

Dữ liệu của product khá lớn, điều này khá phổ biến với các platform liên quan tới E-Commerce như Magento.

Tương tự như customer, bảng lưu trữ chính là catalog\_product\_entity bao gồm các bảng sau:

* entity\_id: Khóa chính (primary key)
* create\_at, update\_at: Thời gian tạo, và lần cuối update
* entity\_type\_id: Khóa chính của bảng eav\_entity\_type
* attribute\_set\_id: Khóa chính của bảng eav\_attribute\_set
* type\_id: Loại hàng
* sku: Mã hàng
* …

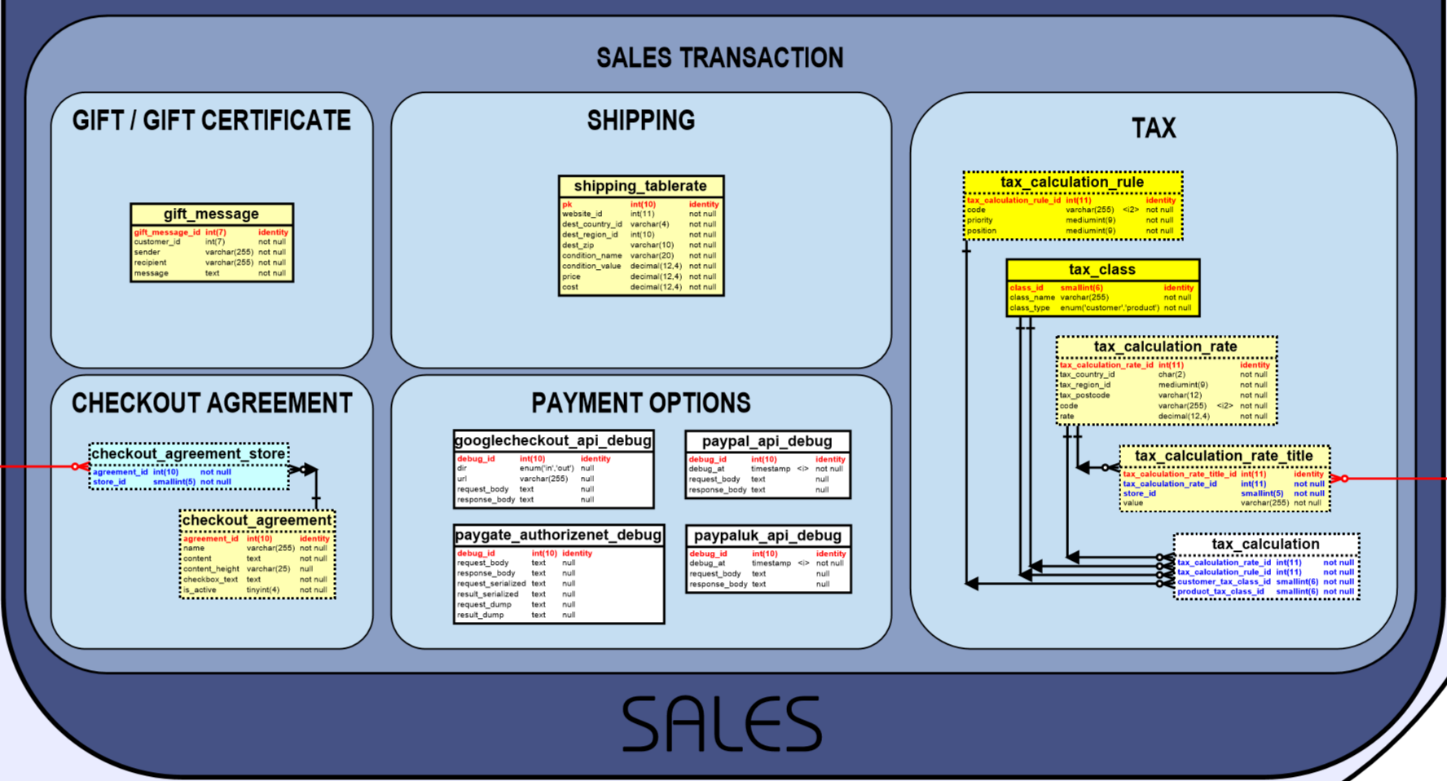
Những bảng kế thừa catalog\_product\_entity lưu trữ những thuộc tính sản phẩm

* Bảng catalog\_product\_link: Diễn tả mối quan hệ giữa các sản phẩm.
* Bảng catalog\_product\_link\_type: Lưu các mối quan hệ
* Bảng catalog\_product\_option: Chứa thông tin về các option mà sản phẩm có thể có. Bảng này link tới catalog\_product\_bundle\_option.
* Bảng catalog\_product\_bundle\_website: Hiển thị quan hệ giữa sản phẩm và website
* Bảng catalog\_product\_index
* …

Bên cạnh đó, các bảng catalog\_category\_flat\_store\_x, catalog\_product\_flat\_1 (x=1,2,3….) hiển thị mối quan hệ giữa các category và product.

Tuy nhiên, Magento có nhược điểm: khi tạo một store mới, 2 bảng tương tự được máy chủ tự động tạo ra. Gây bất lợi với những website có nhiều store, server không có nhiều bộ nhớ cần thiết.

* 1. **Sale**

****

Dữ liệu của module sale chứa những thông tin cần thiết khi khách muốn mua hàng: Quote, Order, Invoice, Shipment, Creditmemo. Không như 2 module Customer và Catalog đã đề cập bên trên, các bảng trong module sale được lưu trong dữ liệu của model.

**Quote:** chứa thông tin khi khách hàng hoặc admin tạo đơn hàng

* Bảng sales\_flat\_quote: Khóa chính
* Bảng sales\_flat\_quote\_address: Chứa thông tin cụ thể về địa chỉ
* Bảng sales\_flat\_quote\_item: Chứa thông tin cụ thể về thông tin đơn hàng
* Bảng sales\_flat\_quote\_payment: Chứa thông tin cụ thể về phương thức thanh toán

**Order:** chứa thông tin sau khi khách hàng đã xác nhận đơn hàng. Tương tự như Quote, bảng chính là sales\_flat\_order, bảng phụ lưu giữ những thông tin cần thiết

**Invoice:** Xử lý thông tin hóa đơn.

* Bảng sales\_flat\_invoice: Khóa chính
* Bảng sales\_flat\_invoice\_item: Lưu trữ chi thông tin của sản phẩm trong hóa đơn
* Bảng sales\_flat\_invoice\_grid: Lưu giữ các báo cáo và phân tích cần thiết

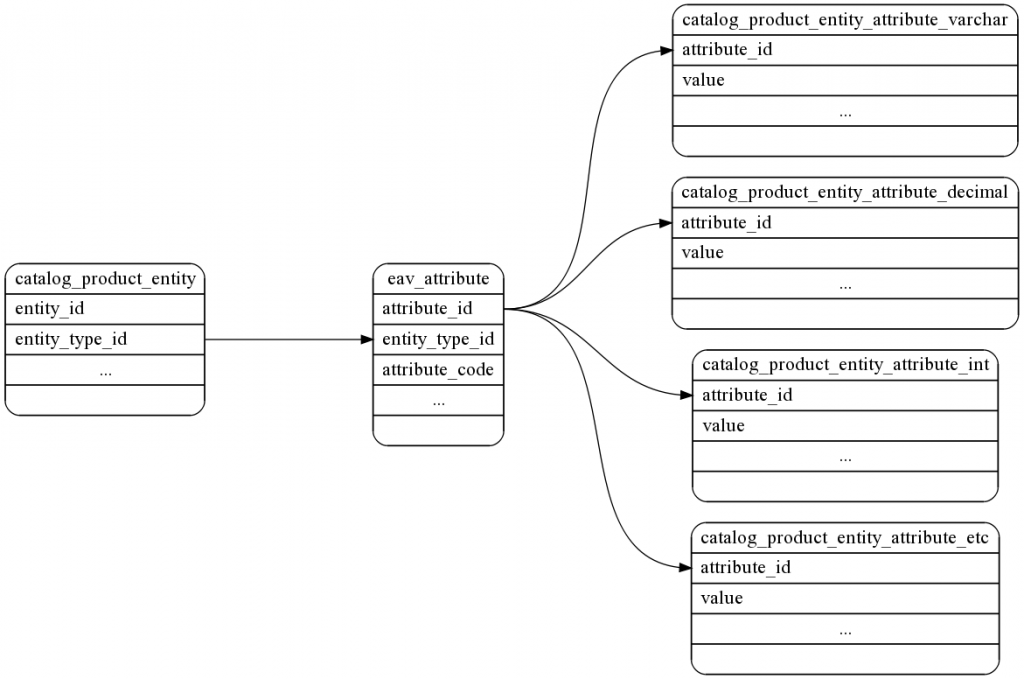
**Shipment & Creditmemo:** các bảng này được thiết kế dựa trên bảng invoice, chứa thông tin xử lý đơn hàng (giao hàng hoặc hủy đơn hàng). Cấu trúc bảng tương tự như bảng Invoice

Bên cạnh đó, có những bảng quan trong khác

* Bảng sales\_order\_status: Thông tin trạng thái
* Bảng sales\_order\_tax: Các loại thuế trong đơn đặt hàng
* Bảng salesrule: Các rang buộc và khuyến mãi của từng đơn hàng
* Bảng sales\_recurring\_profile: ……
  1. **Mô hình EAV**

Entity-Attribute-Value viết tắt là EAV, là 1 mô hình dữ liệu, làm việc với các thực thể (entity) có số lượng các thuộc tính (attribute) có thể mở rộng:

* Entity: Trong Magento, đây là một mục dữ liệu. Ví dụ điển hình như Categories, Products và Customers
* Attribute: Các hạng mục con bên trong Entity. Ví dụ điểm hình như: Description, Price, …
* Value: Giá trị của từng hạng mục

Magento xây dựng hệ thống EAV:  
[](http://namluu.com/wp-content/uploads/2017/09/eav.png)  
Phân tích theo mô hình trên:

* catalog\_product\_entity chính là thực thể (entity) chứa dữ liệu của sản phẩm – tương đương bảng product
* Còn cái đám catalog\_product\_entity\_attribute\_varchar, -int… Chính là chứa giá trị của thuộc tính (tương đương attribute\_value). Tuy nhiên trong Magento, mỗi kiễu dữ liệu của attribute lại có bảng riêng (varchar, int, decimal, text, datetime …)
* eav\_attribute là bảng chứa thông tin chính của thuộc tính (attribute)
* **Lí do tách attribute\_value ra theo kiễu dữ liệu:**

Đó là vì Magento dễ dàng kiểm soát các attribute của mình hơn.

Khi tạo mới 1 attribute, Magento sẽ hỏi attribute input type là gì? Text, Date, Yes/No hay Dropdown, Price… Với mỗi loại, sẽ hiển thị tiếp chỗ nhập giá trị tương ứng với các validate thích hợp.

Ví dụ: attribute dạng text như: name, description… thì validate khác do với attribute dạng decimal như price, weight…, attribute dạng dropdown thì lại hiện danh sách các field option bên trong…

Đó là 1 cải tiến khá hay của Magento so với EAV truyền thống,

* **Hạn chế**

Khi truy xuất dữ liệu, tốc độ câu truy vấn bị giảm đáng kể vì phải thu thập dữ liệu từ nhiều bảng khác nhau

1. **Môi trường cài đặt của Magento**
2. **Cấu hình tối thiểu**

* Hệ điều hành: Linux x86-64(RHEL, CentOS, Ubuntu, Debian…).
* RAM: tối thiểu 2GB RAM. Bạn có thể sử dụng SWAP nếu hệ thống nhỏ hơn 2GB RAM.
* Webserver: Apache 2.2/2.4 hoặc Nginx 1.8 trở lên.
* Composer: phiên bản ổn định mới nhất 1.4.2.
* Các kĩ thuật cache: sử dụng PHP OPcache. Bên cạnh đó, có thể sử dụng [Redis](https://hocvps.com/cai-dat-redis-tren-centos/) 3.0 hoặc [Varnish](https://hocvps.com/series/varnish/) 3.5 trở lên để page caching. Sử dụng Redis hoặc Memcached để session storage.
* Mail server: Mail Transfer Agent (MTA) hoặc SMTP server

1. **Yêu cầu phần mềm và các thư viện liên quan**

* Giới hạn bộ nhớ PHP nên để ít nhất là 768MB cho tốc độ vận hành bình thường hoặc 2GB cho testing
* Supported Web Servers: Apache 2.2, Apache 2.4
* nginx 1.8 trở lên
* PHP Compatibility: 5.6.x, 5.5.x,7.0.2 up to 7.1.0, except for 7.0.5
* PHP Extensions: bc-math,curl, gd, ImageMagick 6.3.7 (hoặc cao hơn), intl, mbstring, mcrypt, mhash, openssl, PDO/MySQL, SimpleXML, soap, xml, xsl, zip
  + - PHP 7:json, iconv
    - MySQL: MySQL 5.6 (Oracle or Percona)
* SSL: Yêu cầu Transport Layer Security (TLS)

1. **Cài Đặt**

#### **Download**

Khi truy cập vào <https://magento.com/tech-resources/download> , chọn mục ***Full Release with Sample Data*** rồi chọn phiên bản như hình rồi ấn nút ***Download***



Khi ấn nút ***Download***, trang web sẽ yêu cầu bạn đăng nhập, nếu bạn đã có tài khoản rồi thì đăng nhập rồi download, không thì phải đăng ký lấy 1 tài khoản mới down được nhé, bước đăng ký mình không hướng dẫn chi tiết vì nó cũng cơ bản rồi.

Sau khi download về thì bạn giải nén vào đường dẫn gốc của Wamp là ***www*** và đặt tên cho folder chưa toàn bộ mã nguồn giải nén là ***magento2***

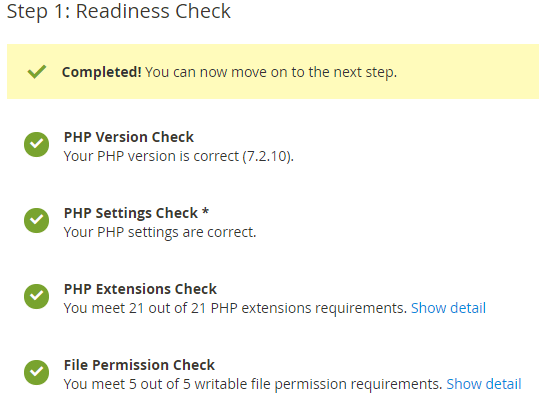
#### **Cài đặt Magento 2**

Sau đó các bạn truy cập vào trình duyệt đường dẫn sau : localhost/magento2

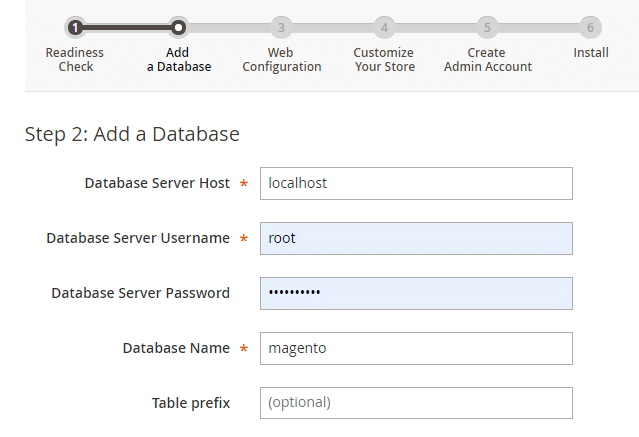
Nhấp vào nút ***Agree and Setup Magento***



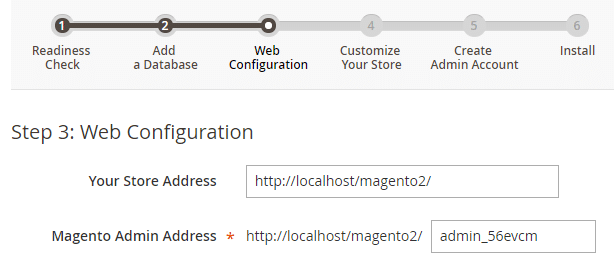
Step 1 : Nhấp vào ***Start Readiness Check,*** hệ thống sẽ kiểm trả cấu hình, nếu tick xanh hết là ok, rồi ấn ***Next***



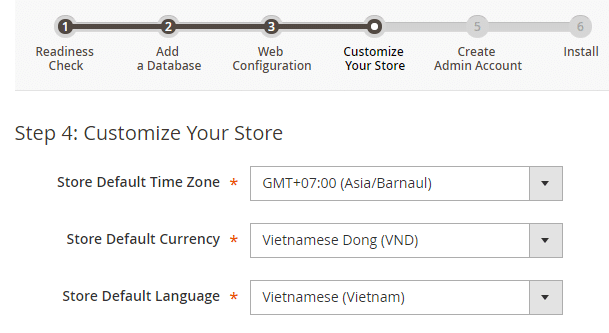
Step 2 : Thêm thông số database của bạn rồi ấn ***Next*** (các bạn lưu ý là phải vào phpmyadmin để tạo 1 database mới rồi lấy tên điền vào mục ***Database name***)



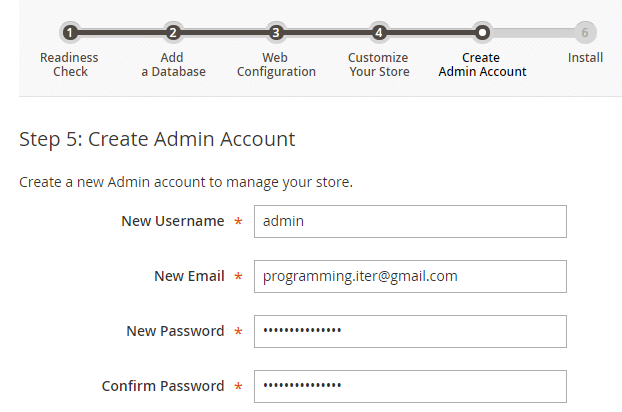
Step 3 : Các thông số trang frontend và đăng nhập admin của Magento, bạn cần nhớ nó để lần sau vào, sau đó ấn ***Next***



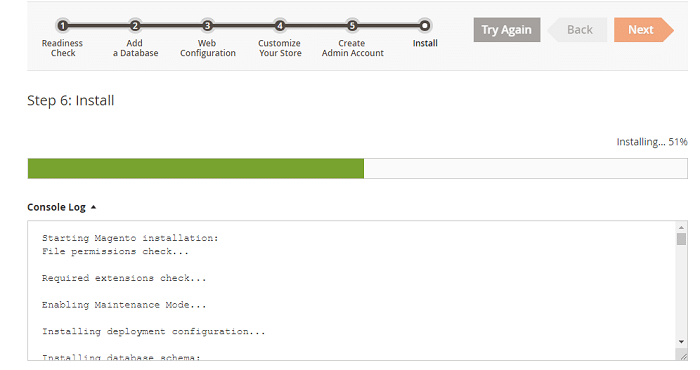
Step 4 : Thiết lập thông số cho website Magento của bạn, rồi ấn ***Next***



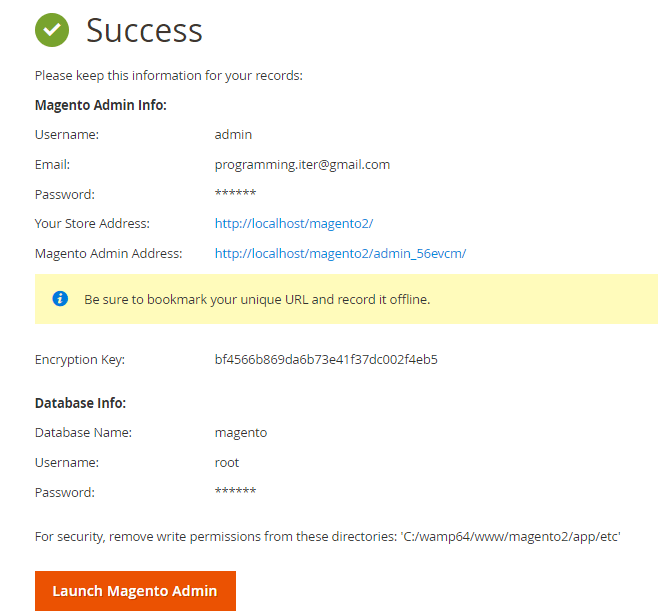
Step 5 : Tạo tài khoản để đăng nhập trang quản trị, rồi ấn ***Next***



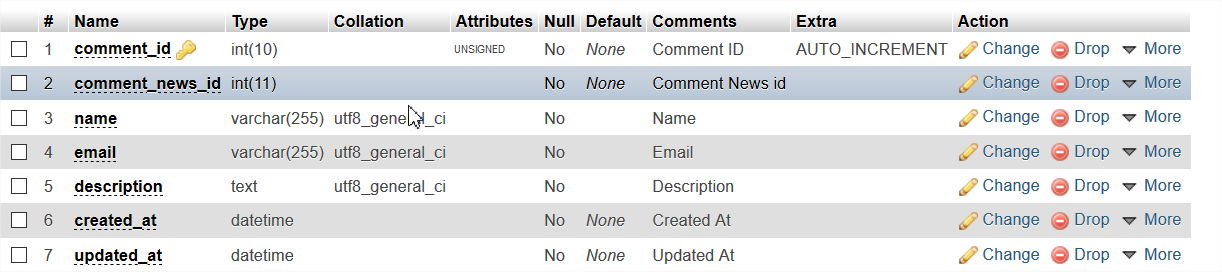
Step 6 : Ấn nút ***Install Now*** để quá trình cài đặt diễn ra, quá trình cài đặt sẽ mất vài phút, ban kiên nhẫn đợi nhé

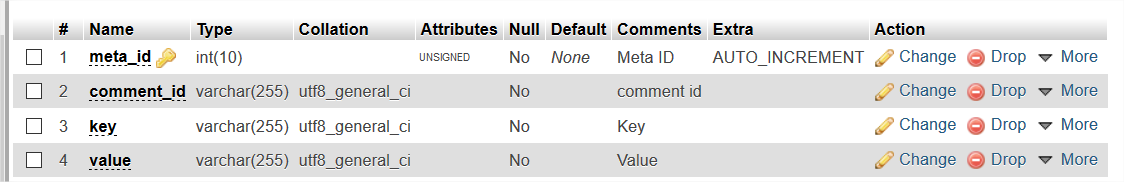


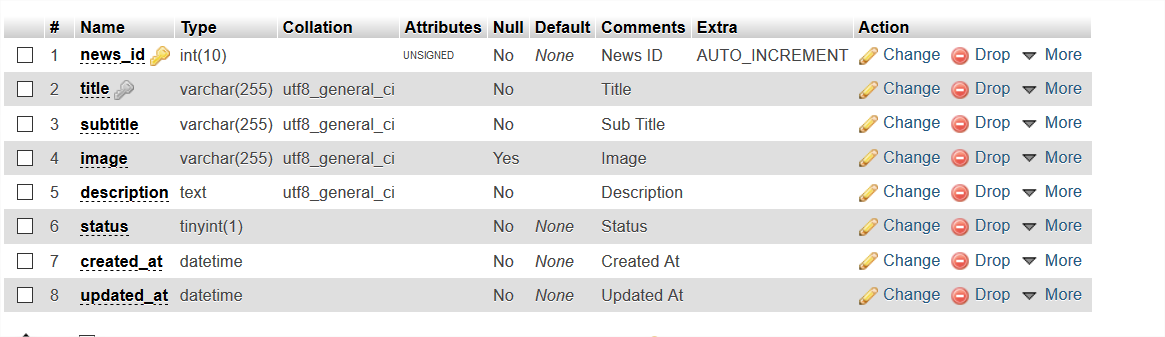
Step 7 : Giao diện cài đặt Magento thành công

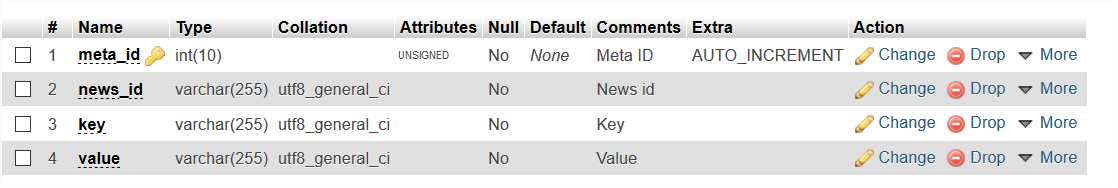


1. **Hiện thực hóa Module tin tức**
2. **Cấu trúc Database Module tin tức**
   * + - Cấu trúc

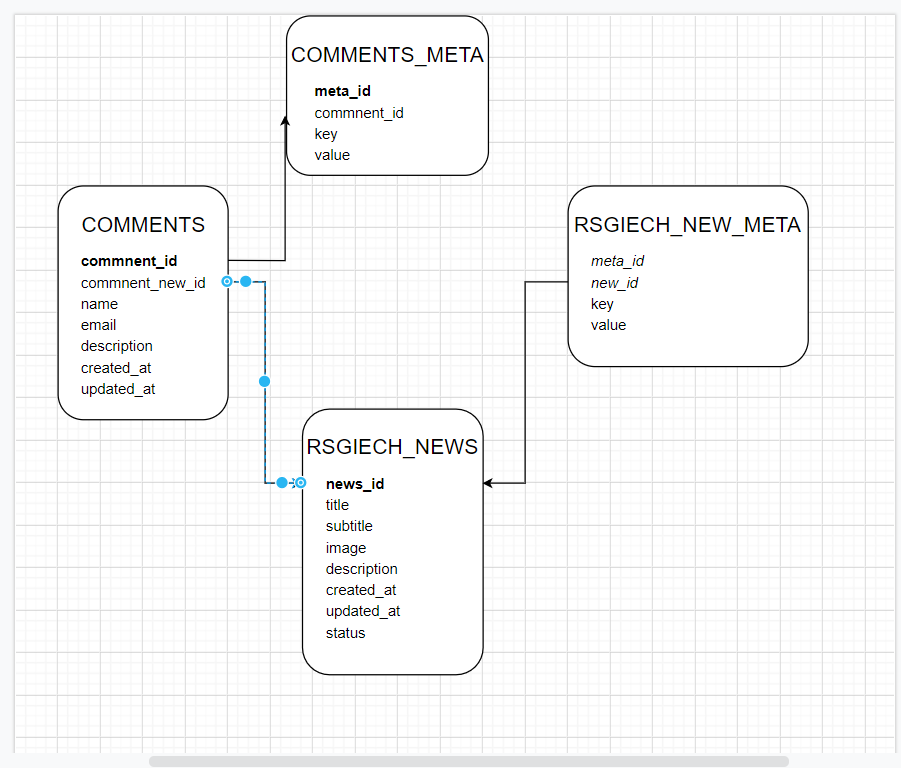
**Table “comments”**

**Table “comments\_meta”**

**Table “rsgiech\_news”**

**Table “rsgitech\_news\_meta”**

* + - * Sơ đồ thiết kế

****

1. **Sơ đồ phân tích DFD**

* Tạo News

Thuật toán:

B1: Nhận D1 từ người dùng.

B2: Kết nối cơ sở dữ liệu.

B3: Đọc D3 từ bộ nhớ phụ.

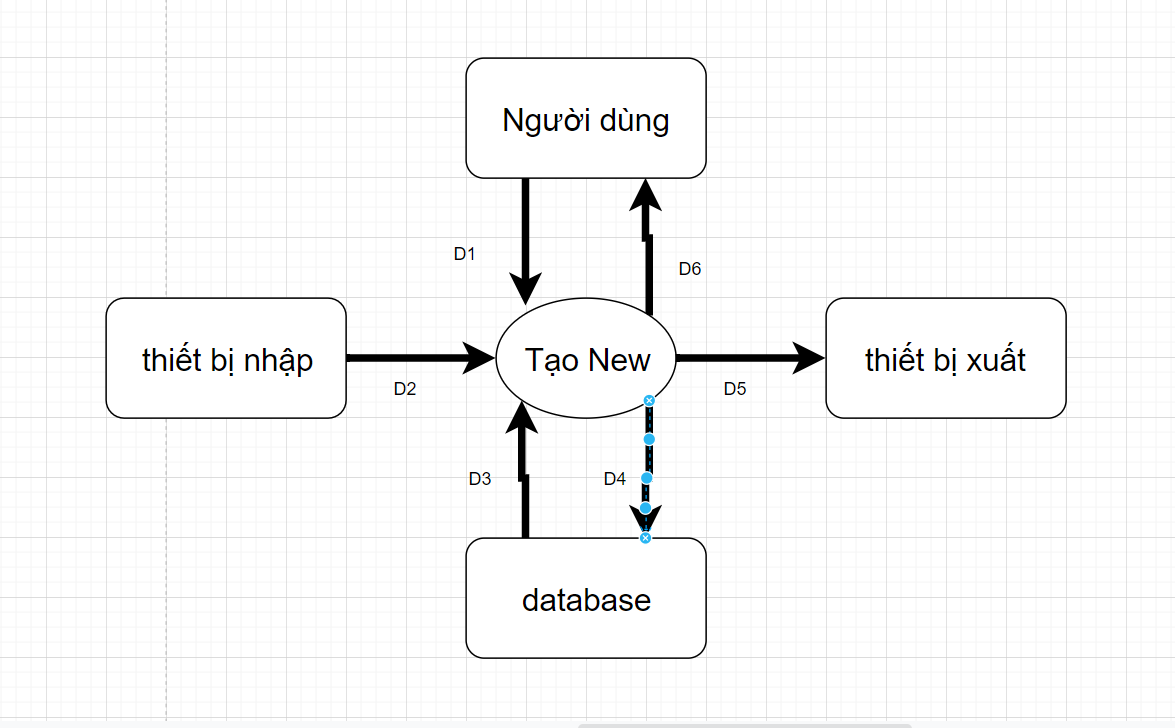
B4: Kiểm tra D1 có phù hợp với điều kiện D3 không.

B5: Nếu không phù hợp thì tới B7.

B6: Lưu D4 xuống bộ nhớ phụ.

B7: Đóng kết nối cơ sở dử liệu.

B8: Kết thúc



D1: Nhập vào tiêu đề, nội dung và chế độ hiển thị

D2: không có

D3: kiểm tra tiêu đề, nội dung có rõ không và

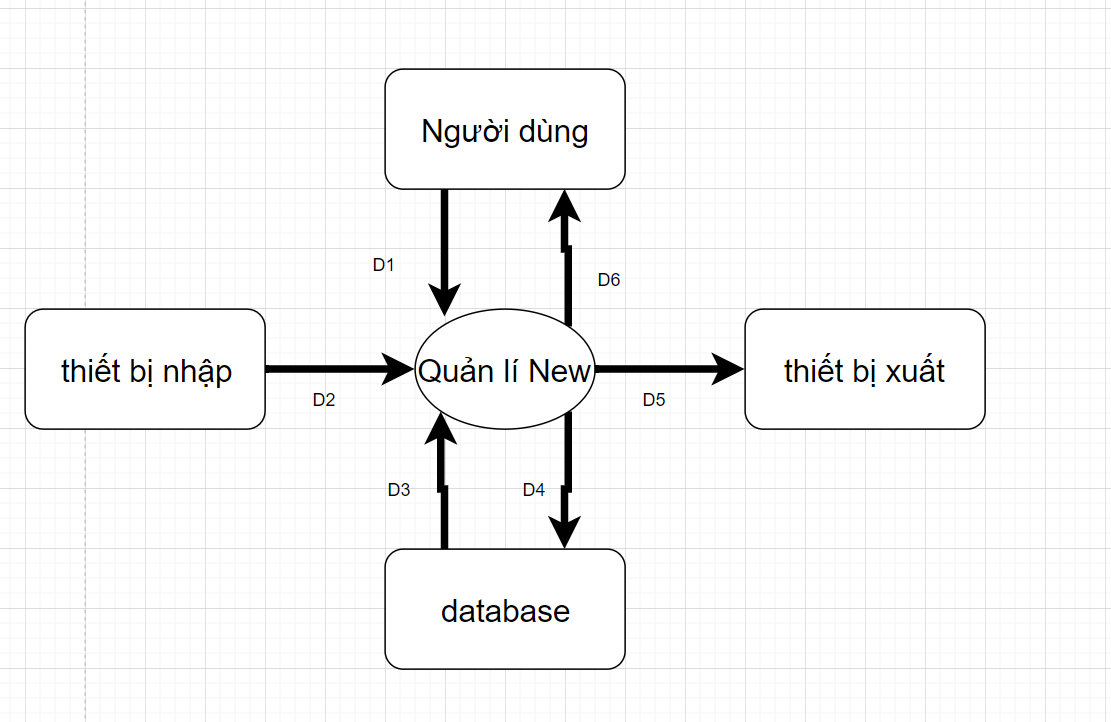
chế độ hiển thị phải ở dạng true hoặc false

D4: lưu D1 +D3 xuống database

D5: D4

D6: không có

* Quản lí tất cả các News



Thuật toán:

B1: Xác thực thông tin của admin

B2:kết nối với cơ sở dữ liệu

B3: truyền ra thông tin về tất cả các new đã được tạo.

B4: Đóng kết nối cơ sở dử liệu.

B5: Kết thúc

D1: nhận vào thông tin người dùng là tài khoản admin

D2: không có

D3: trả lại thông tin về các New đã được tạo

D4: không có

D5: D3

D6: D3

* Sửa News

Thuật toán:

B1: nhập những thông tin muốn thay đổi như là tiêu đề, nội dung, chế độ hiển thị

B2: kết nối với cơ sở dữ liệu

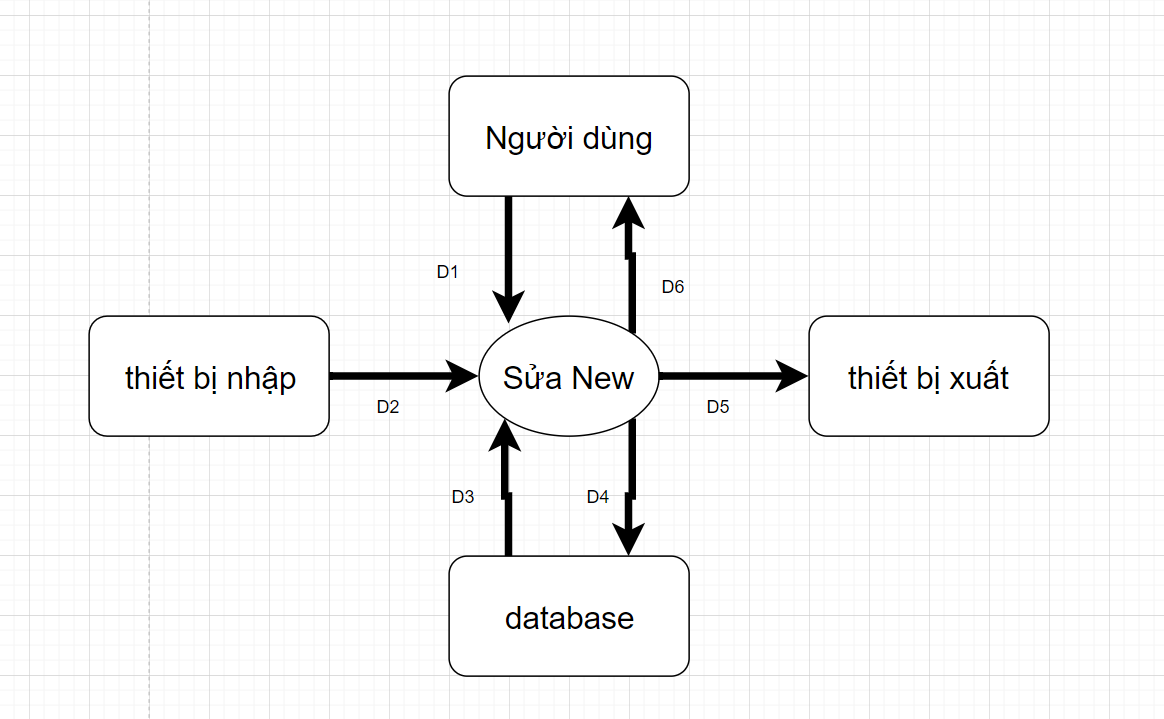
B3: thực hiện kiểm tra id của new có tồn tại không. Nếu không tồn tại id chuyển sang B6

B4: kiểm tra các thuốc tính truyền vào có rỗng không. Nếu có thuộc tính rỗng sẽ chuyển sang B6

B5: Thực hiện ghi đề các thông tin trước đó.

B6: đóng kết nối database

B7: kết thúc



D1: nhập lại các thông như tiêu đề, nội dung, chế độ hiển thị, id của new

D2: không có

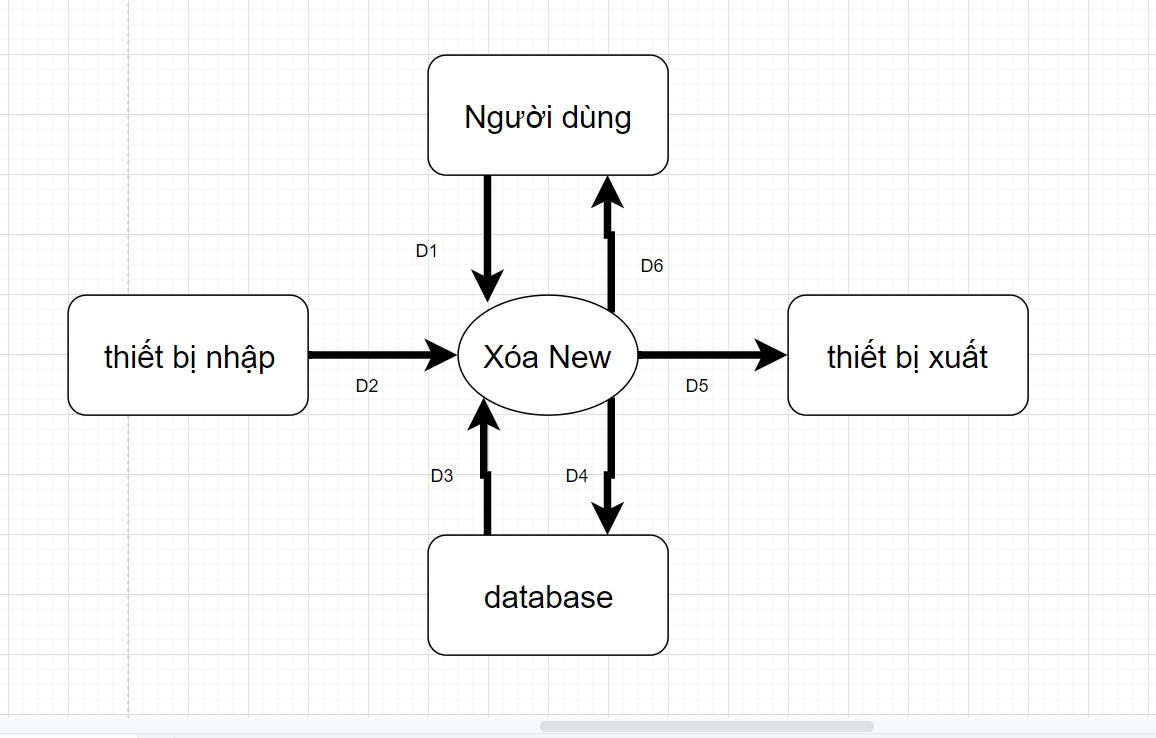
D3: kiểm tra id của new có tồn tại không sau đó kiểm tra tiêu đề, nội dung có giá trị nào bị null không. Các giá trị như tiêu đề, nội dung, chế độ hiển thị có thể giống với giá trị cũ

D4: thực hiện ghi đè tất cả nội dung được gửi xuống database

D5: không có

D6: trả về trạng thái success là true hoặc false

* Xóa News



Thuật toán:

B1: nhập id cần xóa của new

B2: kết nối với cơ sở dữ liệu

B3: thực hiện kiểm tra id của new có tồn tại không. Nếu không tồn tại id chuyển sang B5

B4: thực hiện xóa thông tin theo id truyền vào

B5: đóng kết nối database

B6: kết thúc

D1: nhập thông tin về id của New cần xóa

D2: không có

D3: kiểm tra id của new có tồn tại hay không

D4: truyền id của new cần xóa đến database và thực hiện xóa

D5: không có

D6: trả về trạng thái success là true hoặc false

* Load tất cả các new hiện tại trên trang font-end

Thuật toán:

B1: gửi request tới server yêu cầu load tất cả cá new hiện tại

B2: kết nối với cơ sở dữ liệu

B3: kiểm tra và truyền dữ liệu từ server

B4: đóng kết nối database

B5: cập nhật lại giao diện nguồi dùng

B6: kết thúc



D1: gửi request yêu cầu load tất cả các new hiện tại

D2: không có

D3: Trả về tất cả thông tin của các news

D4: không có

D5: D3

D6: Cập nhật lại giao diện người dùng

* Đọc thông tin chi tiết của news

Thuật toán:

B1: gửi request với id của new hiện tại lên server

B2: kết nối với database

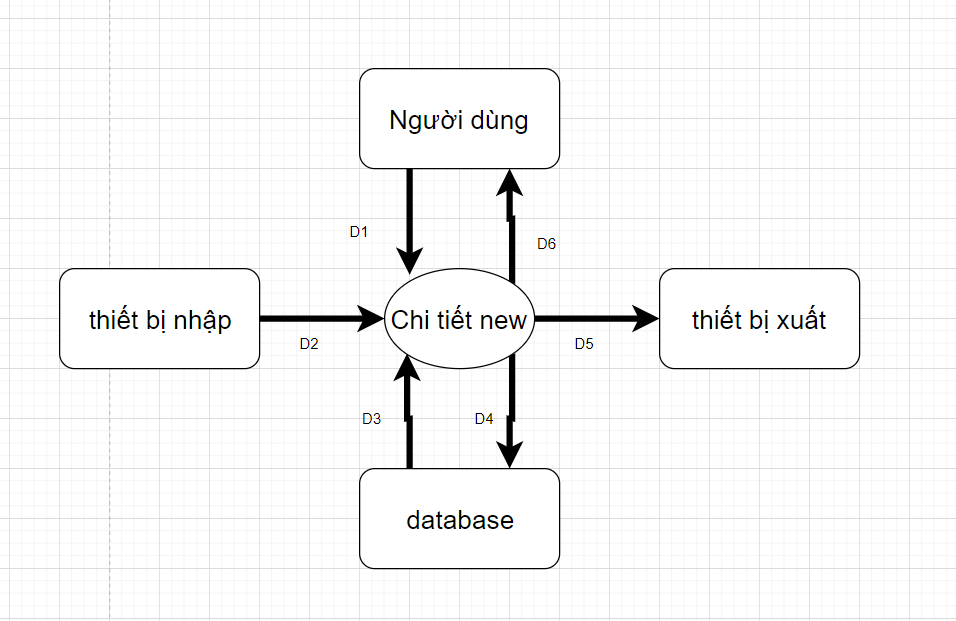
B3: server trả về toàn bộ thông tin của các new

B4: thực hiện kiểm trả id của new yêu cầu có trùng với các new đã được tra về hay không. Nếu không thì chuyển thẳng đến bước 6

B5: chuyển người dùng đến router chứa thông tin chi tiêt của new

B6: đóng kết nối database

B7: kết thúc



D1: gửi request với id của new cần xem chi tiết

D2: không có

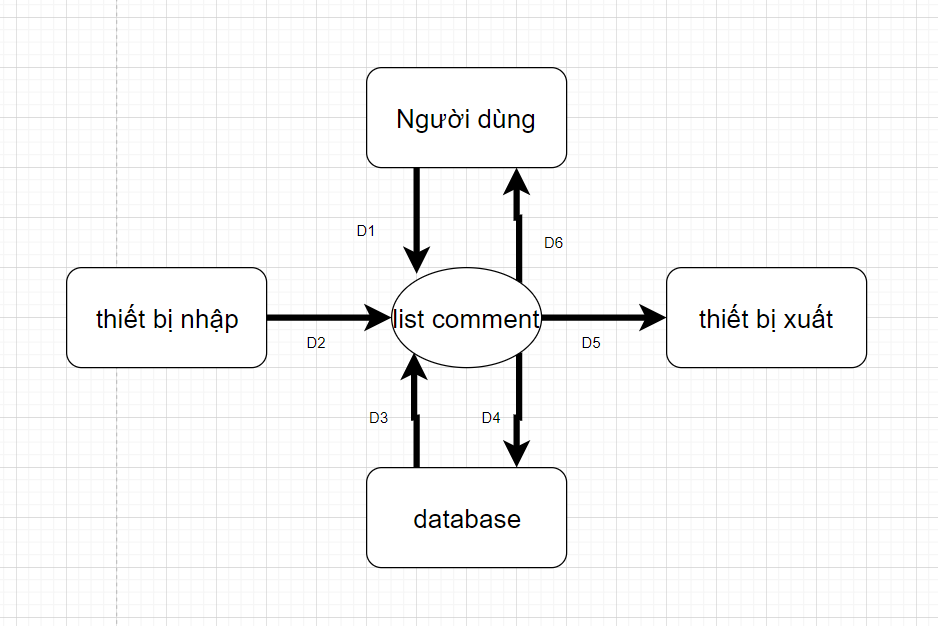
D3: trả về thông tin của tất cả các new và thực hiện kiểm tra new hiện tại có đúng hay không

D4: không có

D5: trả về thông tin của new trùng khớp với yêu cầu

D6: D5

* Load danh sách các comment



Thuật toán:

B1: gửi request yêu cầu load toàn bộ comment với id của new hiện tại lên server

B2: kết nối với database

B3: server trả về toàn bộ các comment với id là new hiện tại

B4: thực hiện load toàn bộ comment lên giao diện người dùng

B5: đóng kết nối database

B6: kết thúc

D1: gửi request yêu cầu load toàn bộ comment với id của new hiện tại

D2: không có

D3: trả về thông tin của tất cả các comment với id là new hiện tại

D4: không có

D5: thực hiện tải lên giao diện người dùng

D6: D5

* Thêm comment mới cho bài viết

Thuật toán:

B1: nhập thông tin từ người dùng

B2: kết nối với database

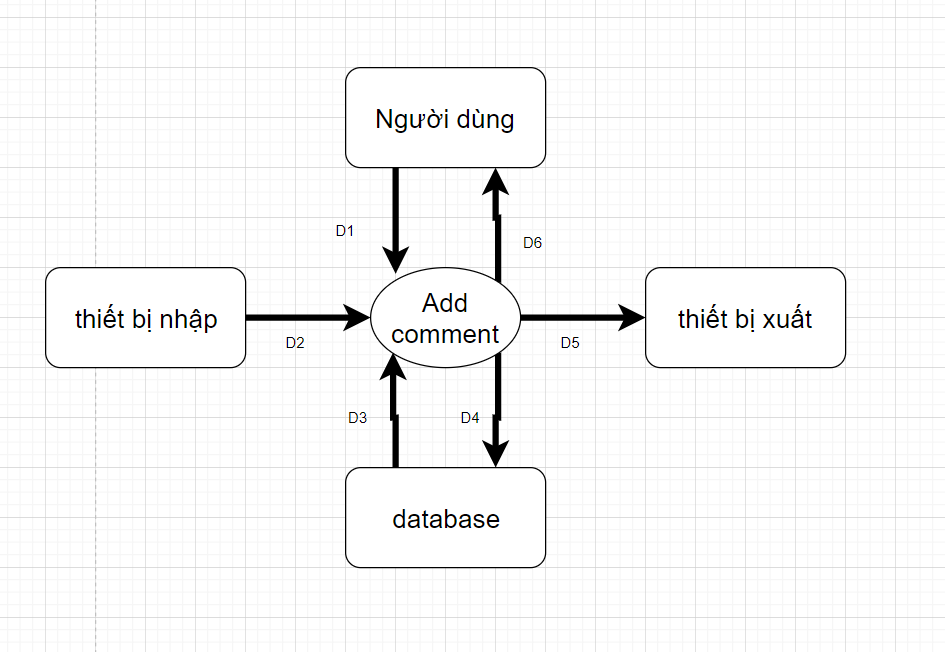
B3: kiểm tra thông tin được nhập từ người dùng có đúng hay không, có dữ liệu nào bị null không. Nếu không đúng chuyển sang bước 6

B4: thực hiện ghi dữ liệu vào database

B5: nếu ghi dữ liệu thành công trả về success là true, nếu khong thành công trả về false

B6: đóng kết nối với database

B7: kết thúc



D1: nhập thông tin từ người dùng như: your name, your email, your content từ người dùng và truyền theo D1

D2: không có

D3: kiểm tra thông tin nhập từ người dùng có thiếu hay không

D4: thực hiện ghi thông tin vào database

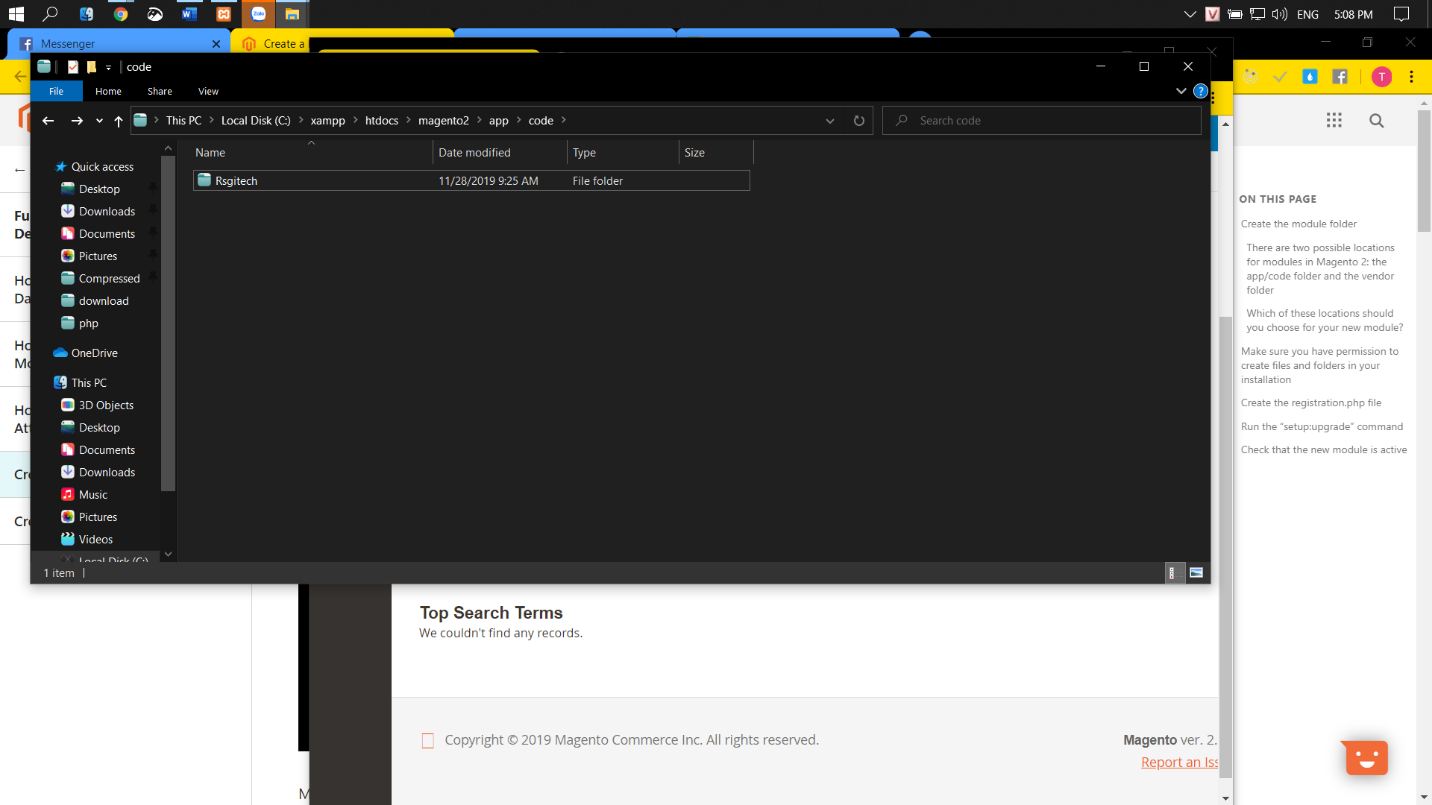
D5: thực hiện tải lên giao diện người dùng

D6: D5

1. **Tích hợp Module Tin tức**

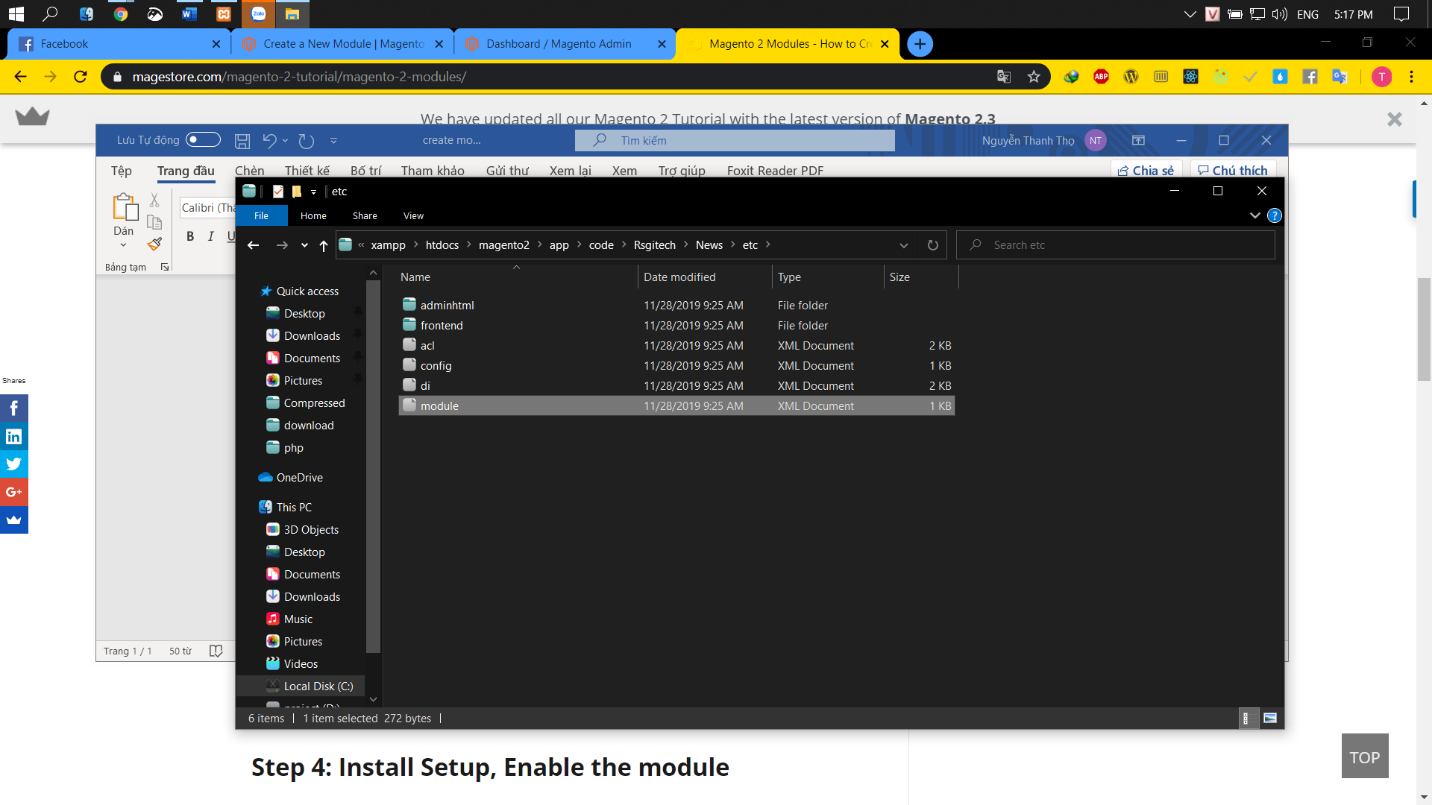
Bước 1: tạo thư mục module

* Chú ý ta có thể tạo thư mục chứa module mới trong 2 thư mục là app/code hoặc vendor/folders

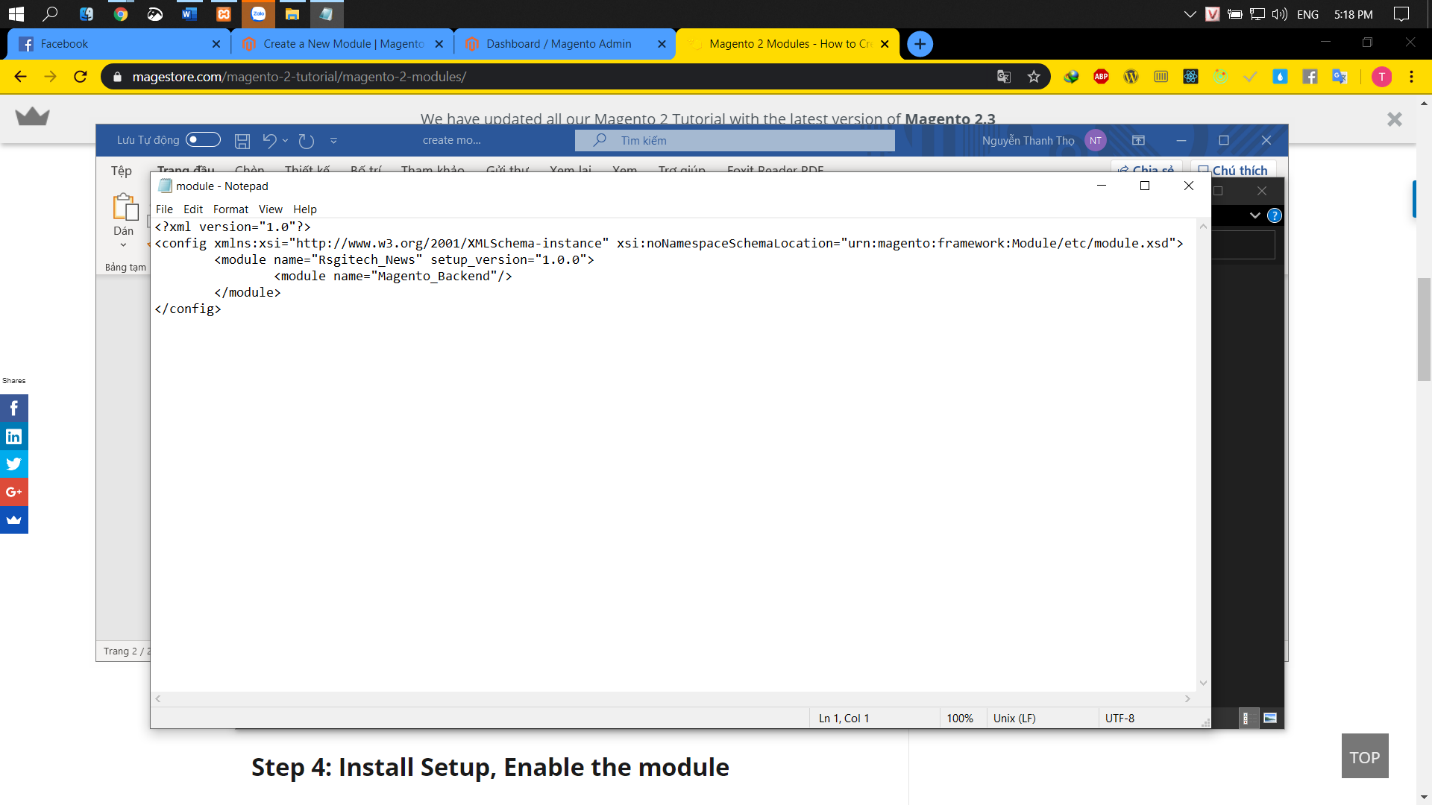


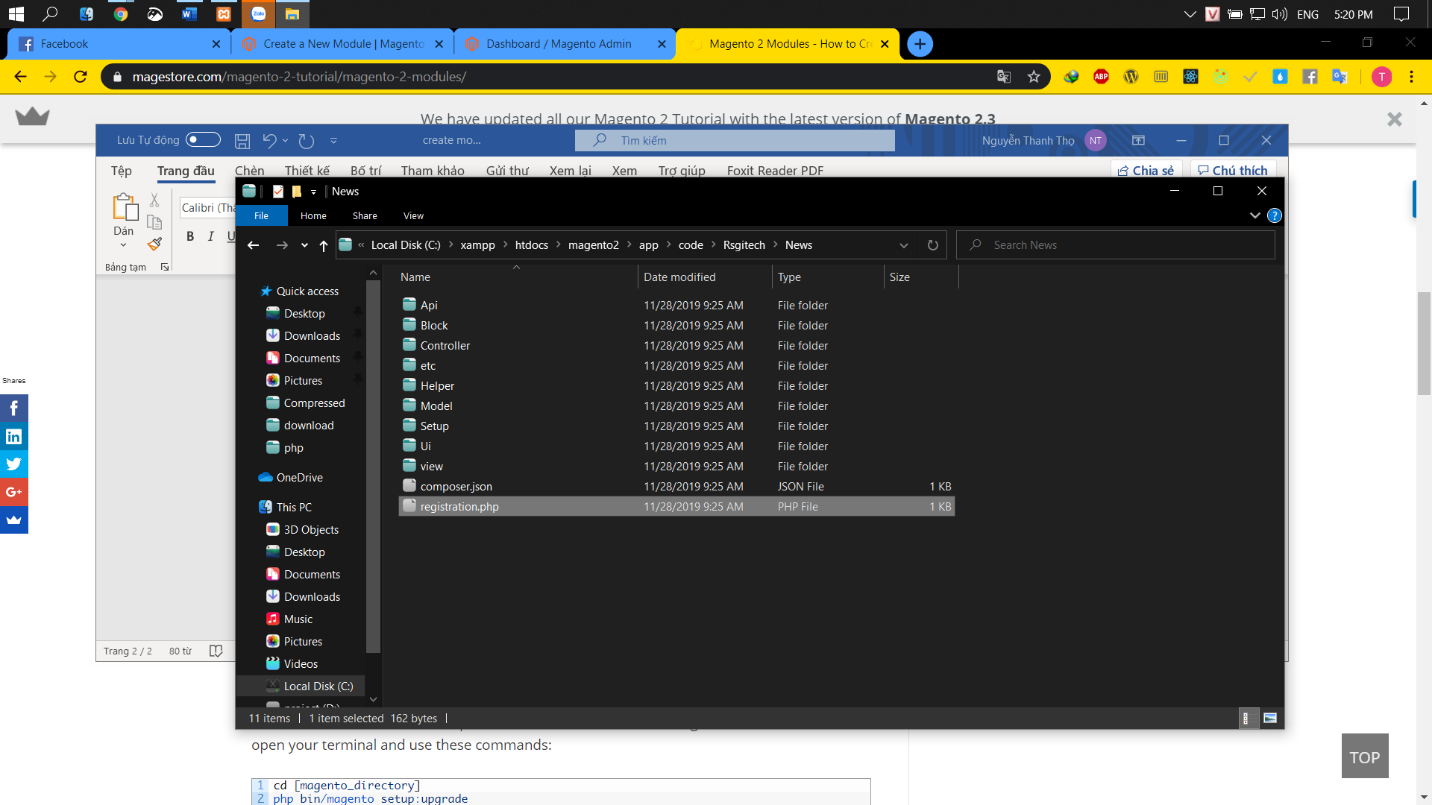
* Như hình thì mình tạo file Rsgitech theo magento2\app\code

Bước 2: khai báo module bằng file module.xml



* Chúng ta cần cấu hình một chút cho nó để magento 2 có thể nhận ra module mới

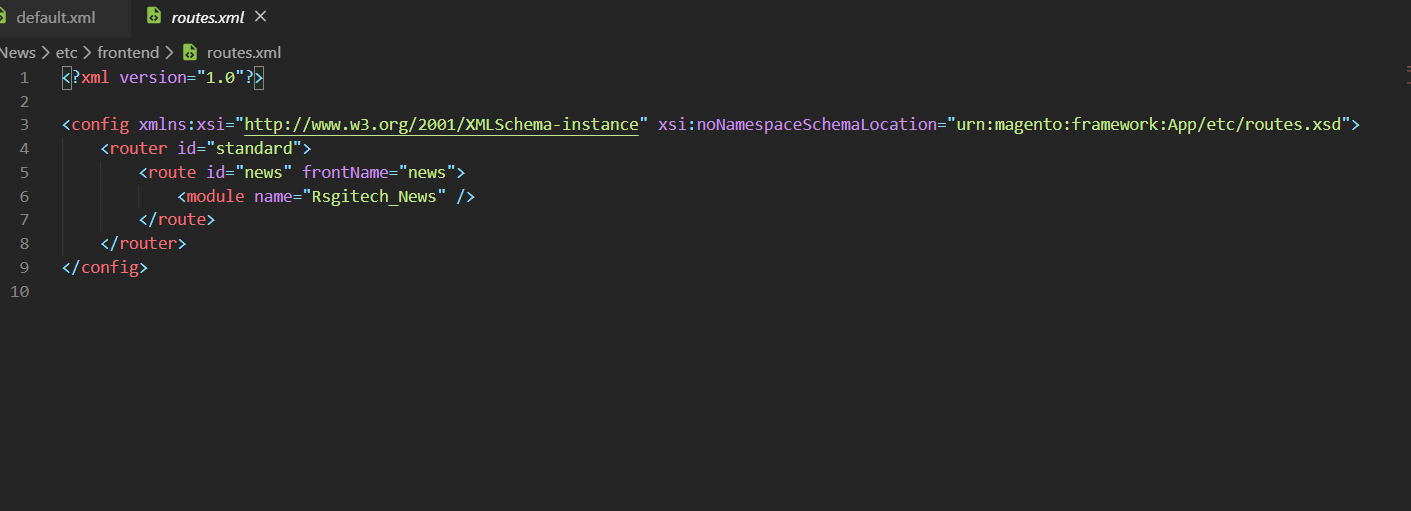
Bước 3: chúng ta cần đăng kí module bằng file registration.php

* Cấu hình cho file

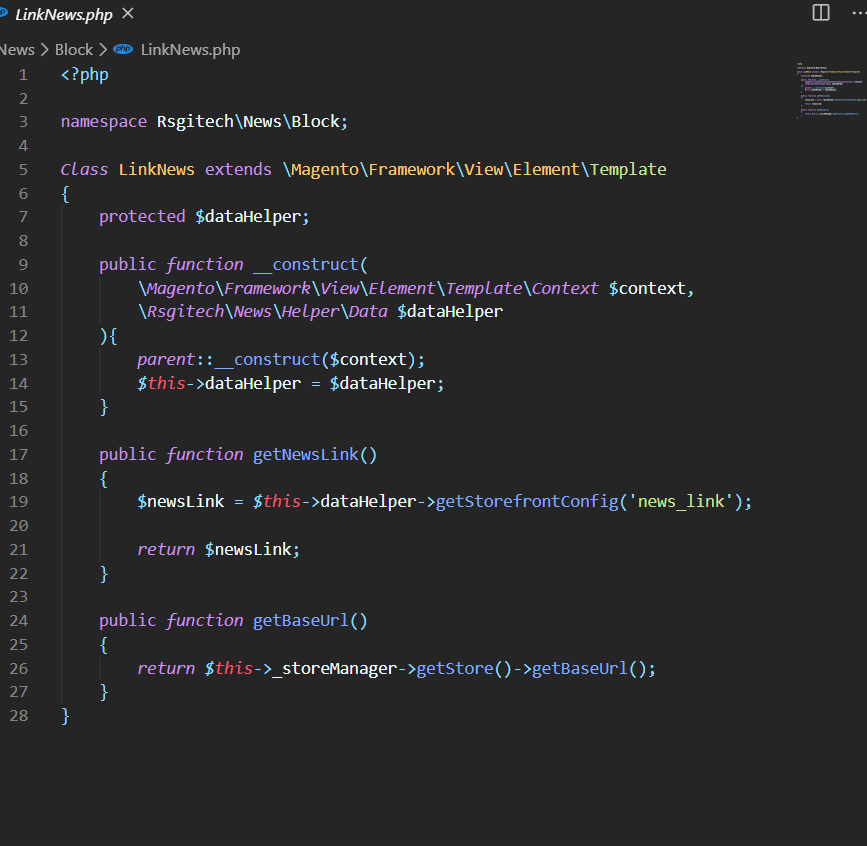


Bước 4: thực hiện cấu hình module mới cho magento 2 theo các file

1. Tạo file routes.xml

Rsgitech\News\etc\frontend\routes.xml

1. Tạo file Index.php Controller Action

Rsgitech\News\Controller\Index\Index.php

1. Tạo file LinkNews.php Block

Rsgitech\News\Block\LinkNews.php

1. Tạo file link.phtml Tmplate File

Rsgitech\News\view\frontend\templates\link.phtml

1. Tạo file defaul.xml Layout

Rsgitech\News\view\frontend\layout\default.xml

6.Tạo file ListNews.php Block

Rsgitech\News\Block\ListNews.php

<?php

namespace Rsgitech\News\Block;

*Class* ListNews extends \Magento\Framework\View\Element\Template

{

    protected $allNewsFactory;

    public *function* \_\_construct(

        \*Magento*\*Framework*\*View*\*Element*\*Template*\*Context* $context,

        \*Rsgitech*\*News*\*Model*\*AllnewsFactory* $allNewsFactory

    ){

*parent*::\_\_construct($context);

*$this*->allNewsFactory = $allNewsFactory;

    }

    public *function* getBaseUrl()

    {

*return* *$this*->\_storeManager->getStore()->getBaseUrl();

    }

    public *function* getListNews()

    {

        $page = (*$this*->getRequest()->getParam('p'))? *$this*->getRequest()->getParam('p') : 1;

        $limit = (*$this*->getRequest()->getParam('limit'))? *$this*->getRequest()->getParam('limit') : 2;

        $collection = *$this*->allNewsFactory->create()->getCollection();

        $collection->addFieldToFilter('status',1);

        $collection->setPageSize($limit);

        $collection->setCurPage($page);

*return* $collection;

    }

    protected *function* \_prepareLayout(){

*parent*::\_prepareLayout();

*$this*->pageConfig->getTitle()->set(\_\_('Latest News'));

*if* (*$this*->getListNews()){

            $pager = *$this*->getLayout()->createBlock('Magento\Theme\Block\Html\Pager', 'rsgitech.news.pager')

                                    ->setAvailableLimit(array(2=>2,10=>10,15=>15,20=>20))

                                    ->setShowPerPage(true)

                                    ->setCollection(*$this*->getListNews());

*$this*->setChild('pager', $pager);

*$this*->getListNews()->load();

        }

*return* *$this*;

    }

    public *function* getPagerHtml()

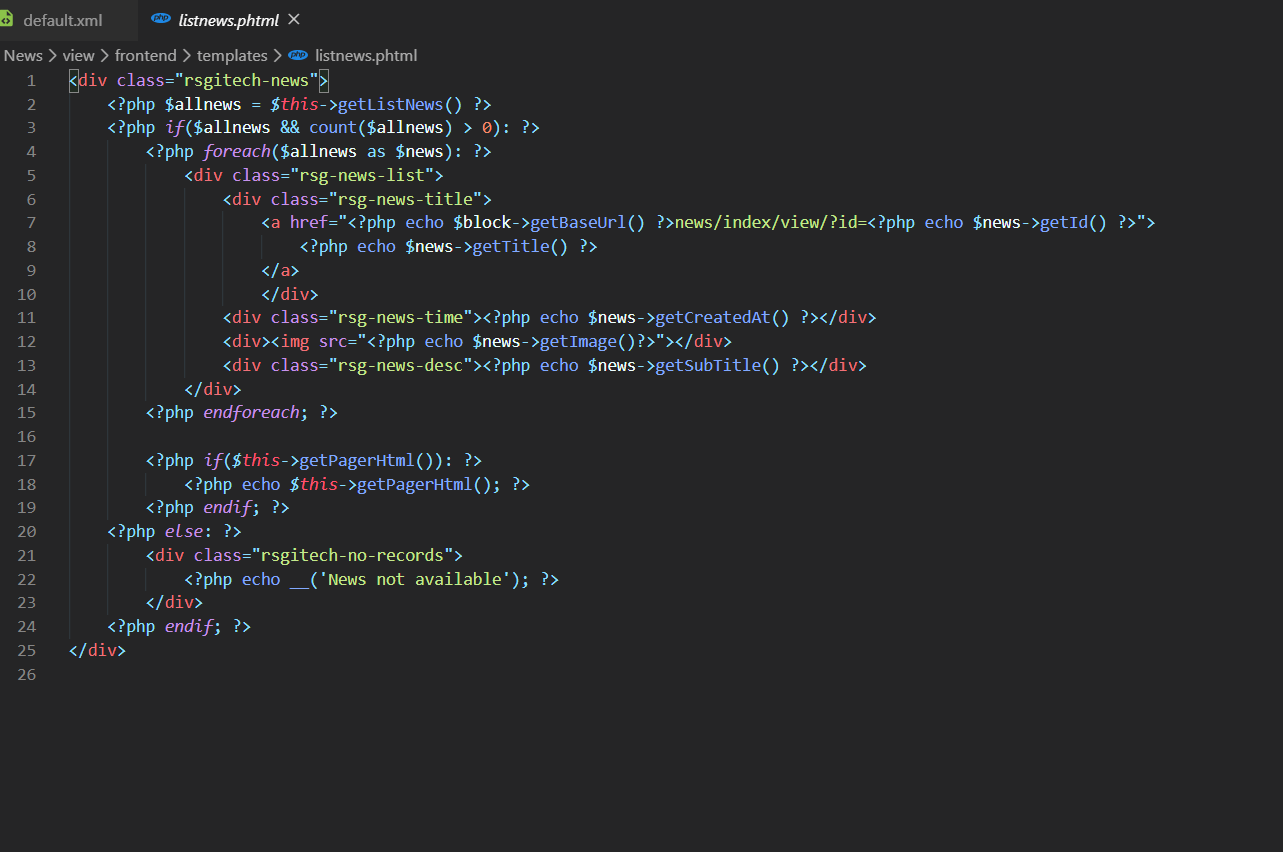
    {

*return* *$this*->getChildHtml('pager');

    }

}

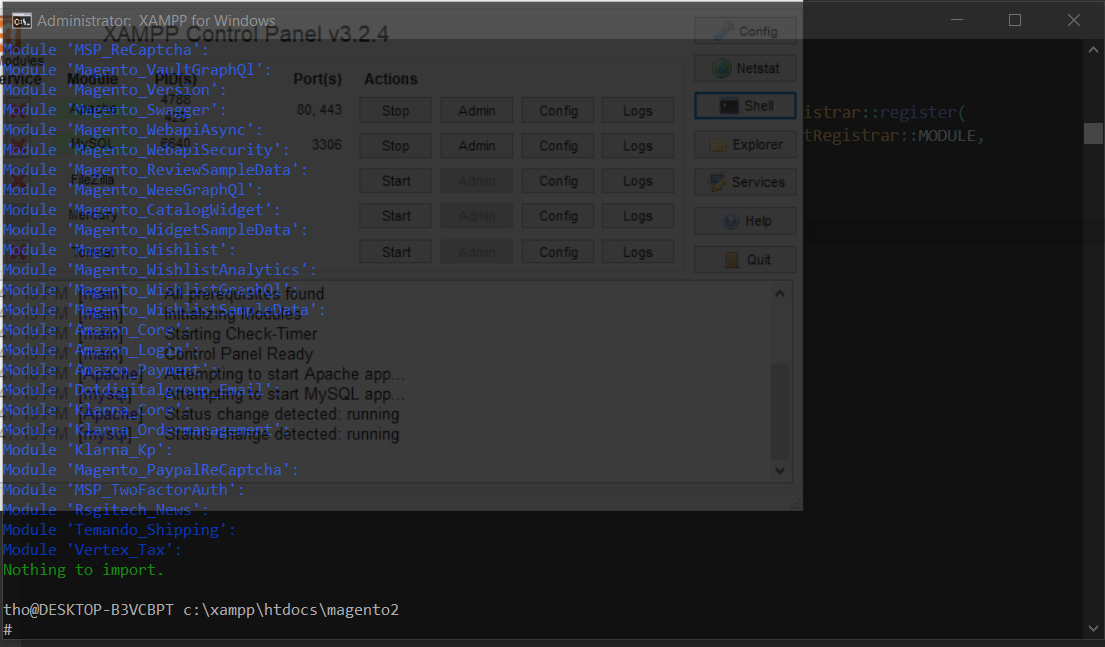
1. Tạo file listnews.phtml Template

Rsgitech\News\view\frontend\templates\listnews.phtml

1. Tạo file news\_index\_index.xml Layout

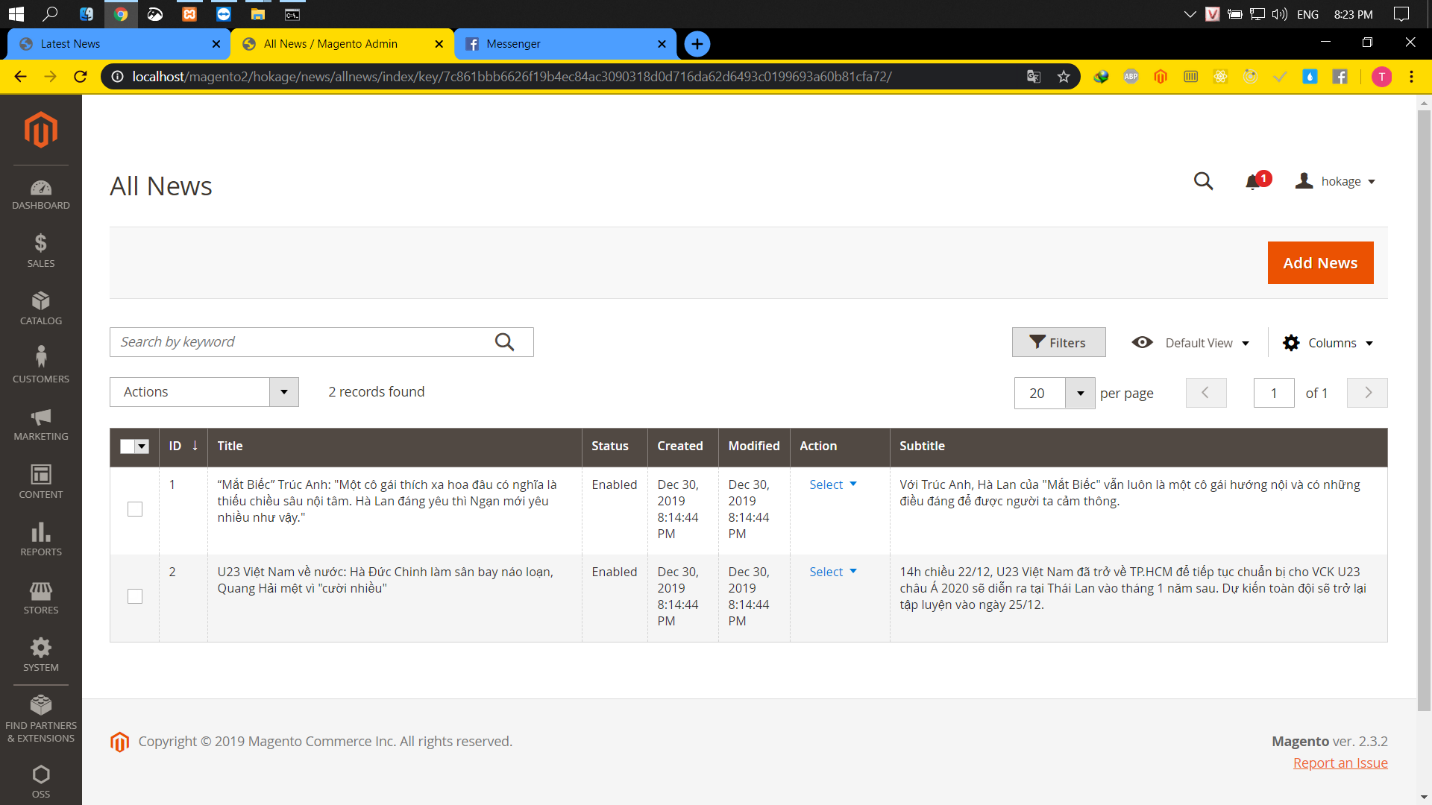
Rsgitech\News\view\frontend\layout\news\_index\_index.xml

Bước 5: cài đặt module bằng câu lệnh php bin/magento2 setup:upgrade

**Hoàn thành thêm module mới cho magento 2**

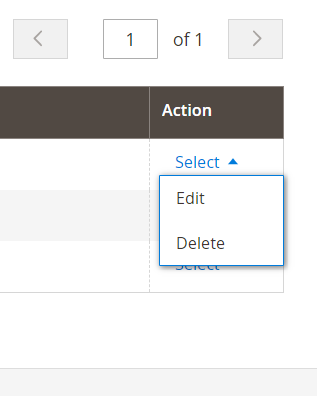


1. **Giao diện và tính năng**
2. **Danh sách xử lý**
3. Thiết kế giao diện

* Màn hình thiết kế
* Thông tin chức năng: Quản lí toàn bộ thông tin các trang Web như là: title, URL Key, Layout, Store View, Status, Created, Modified, Subtitle.
* Thiết kế xữ lý.

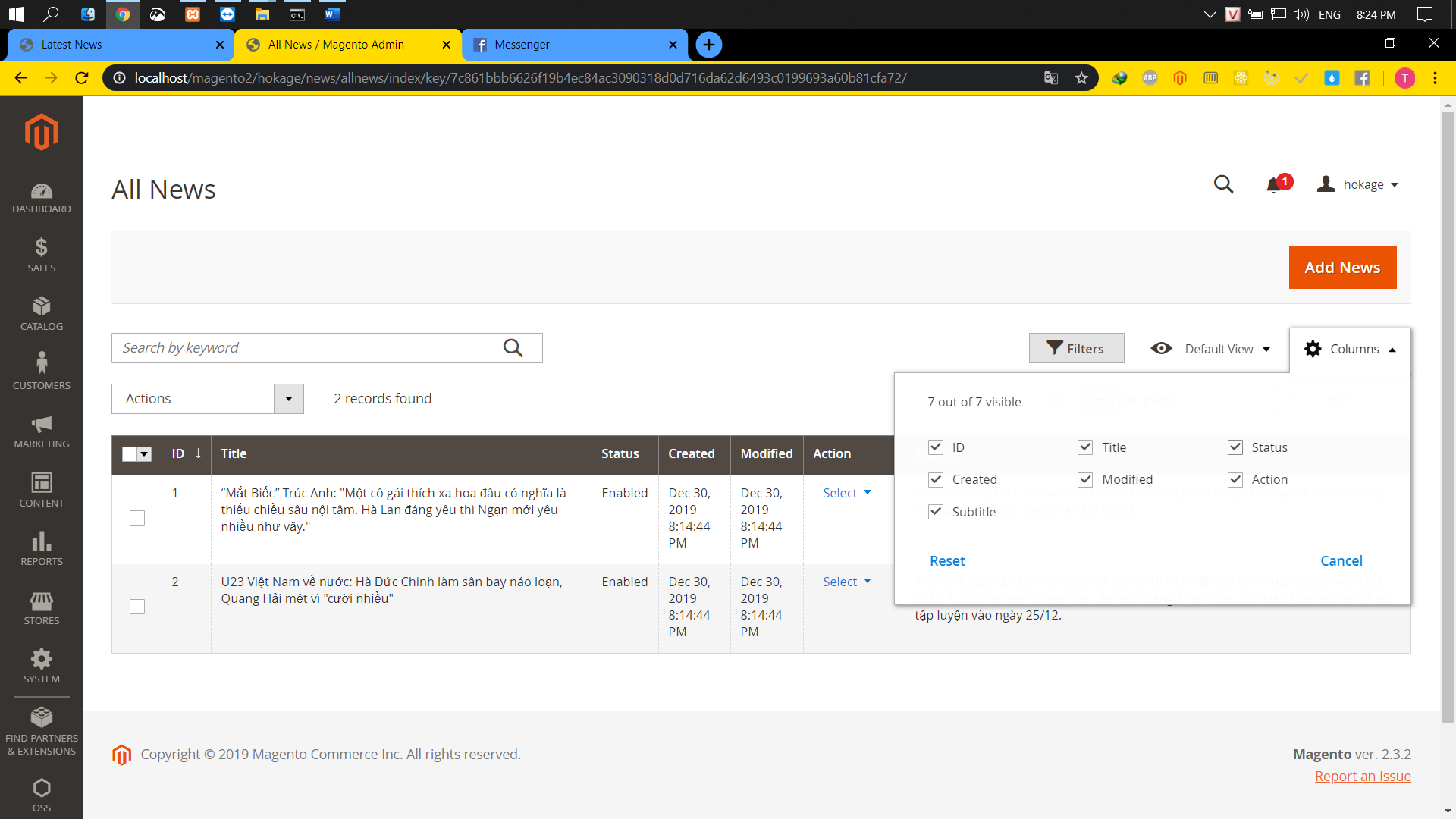
1. Danh sách thiết kế xử lý

* Load thông tin các trang



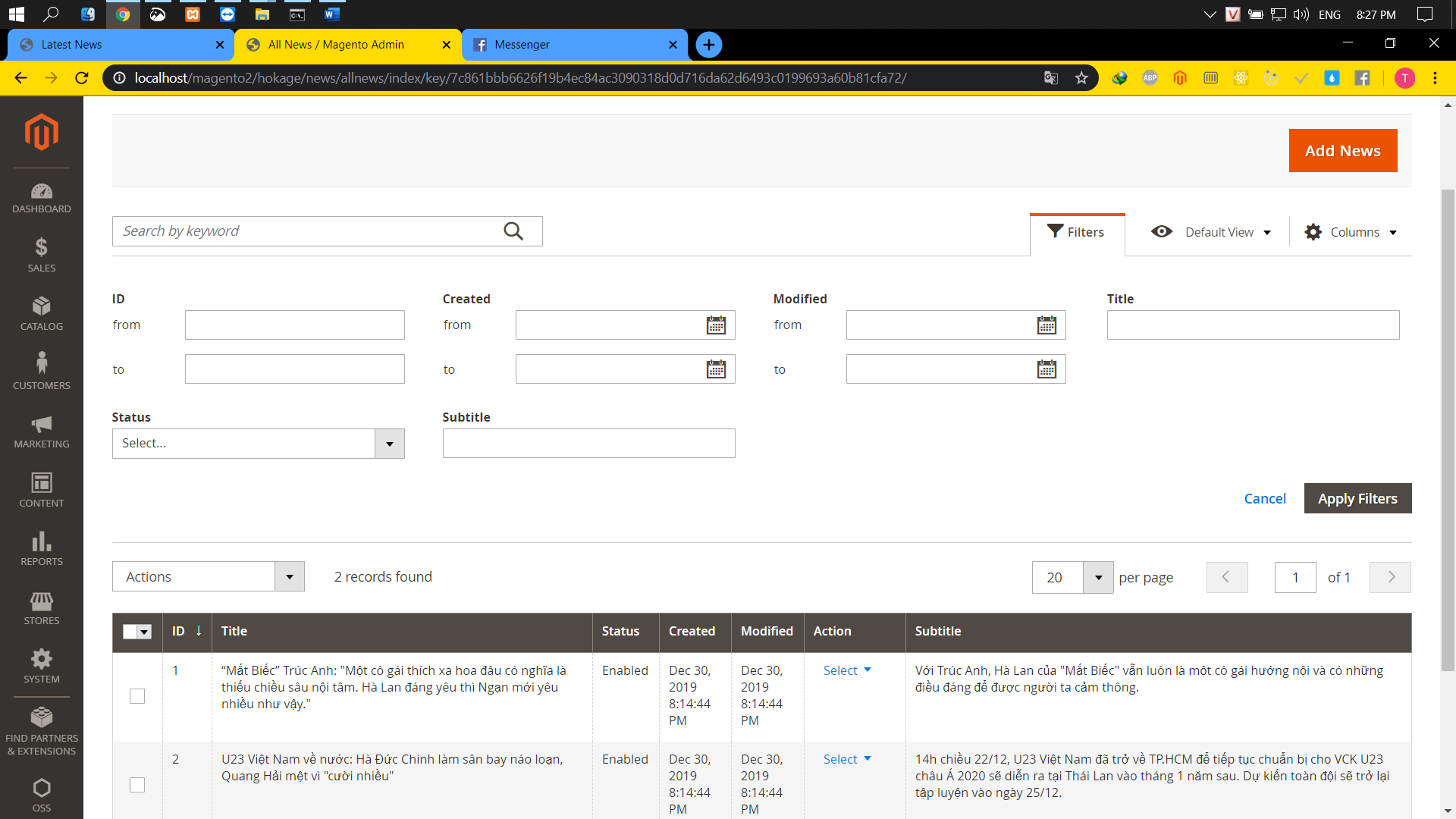
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Edit | Cho phép chỉnh sửa thông tin về các trang web. | Các trang web phải ở trạng thái disable | Không có |
| Delete | Cho phép admin có thể xóa trang web hoàn toàn ra khỏi danh sánh các trang Web | Không có | Không có |

* Tùy chỉnh các cột hiển thị



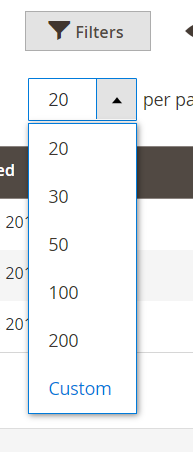
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Hiển thị | Load thông tin các cột muốn hiển thị lên table | Clip vào nút columns để chọn những thông tin cần hiển thị cho table | Có thể không chọn thuộc tính nào |
| Chọn | Chọn các cột muốn hiển thị thông tin chi tiết. | Không có | Các cột được chọn sẽ là các cột hiển thị trên table |
| Reset | Cho tất cả các cột trở về trạng thái mặc định | Không có | Không có |
| Cancel | Hủy bỏ những thay đổi và đóng việc filter | Không có | Không có |

* Lọc thông tin hiển thị lên table

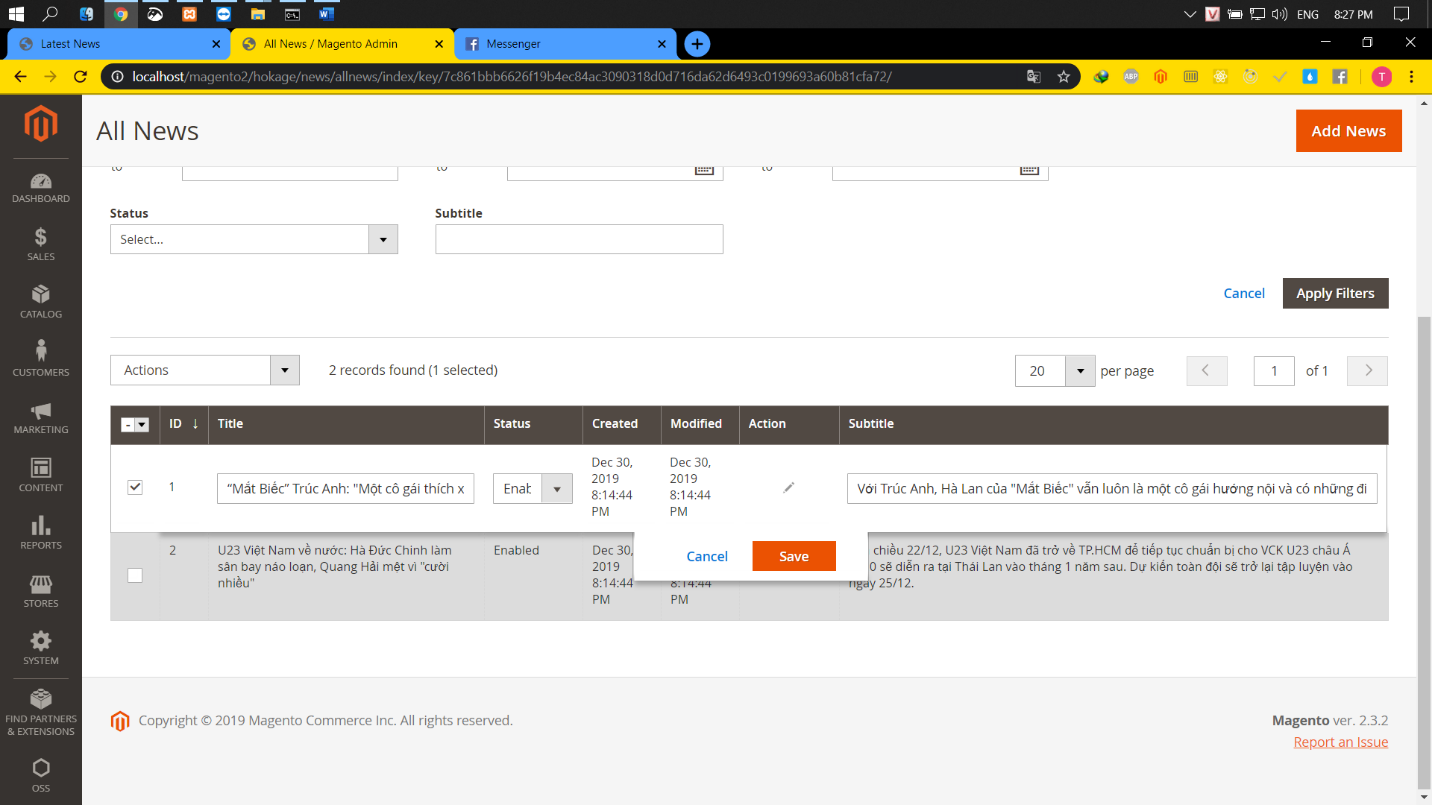


|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Lọc | Lọc thông tin theo các thuộc tính như:   * Id lọc theo thứ tự * Created, modified lọc theo thời gian * Title lọc theo tiêu đề, …Status lọc theo trạng thái của new * Subtitle lọc lọc theo tiêu đề subtile | Điền thôn tin cần lọc vào các ô | Có thể để null các thuộc tính không cần thiết |
| Apply Fileters | Cho phép người dùng áp dụng lọc table theo những thuộc tính đã được điền ở trên | Các thuộc tính được điền cần phải chính xác. Nếu không giá trị trên table sẽ là null | không có gì |
| Cancel | Hủy bỏ việc lọc theo thuộc tính | Click vào hủy bỏ | Không có gì |

* Thay đổi số dòng hiển thị trên bảng

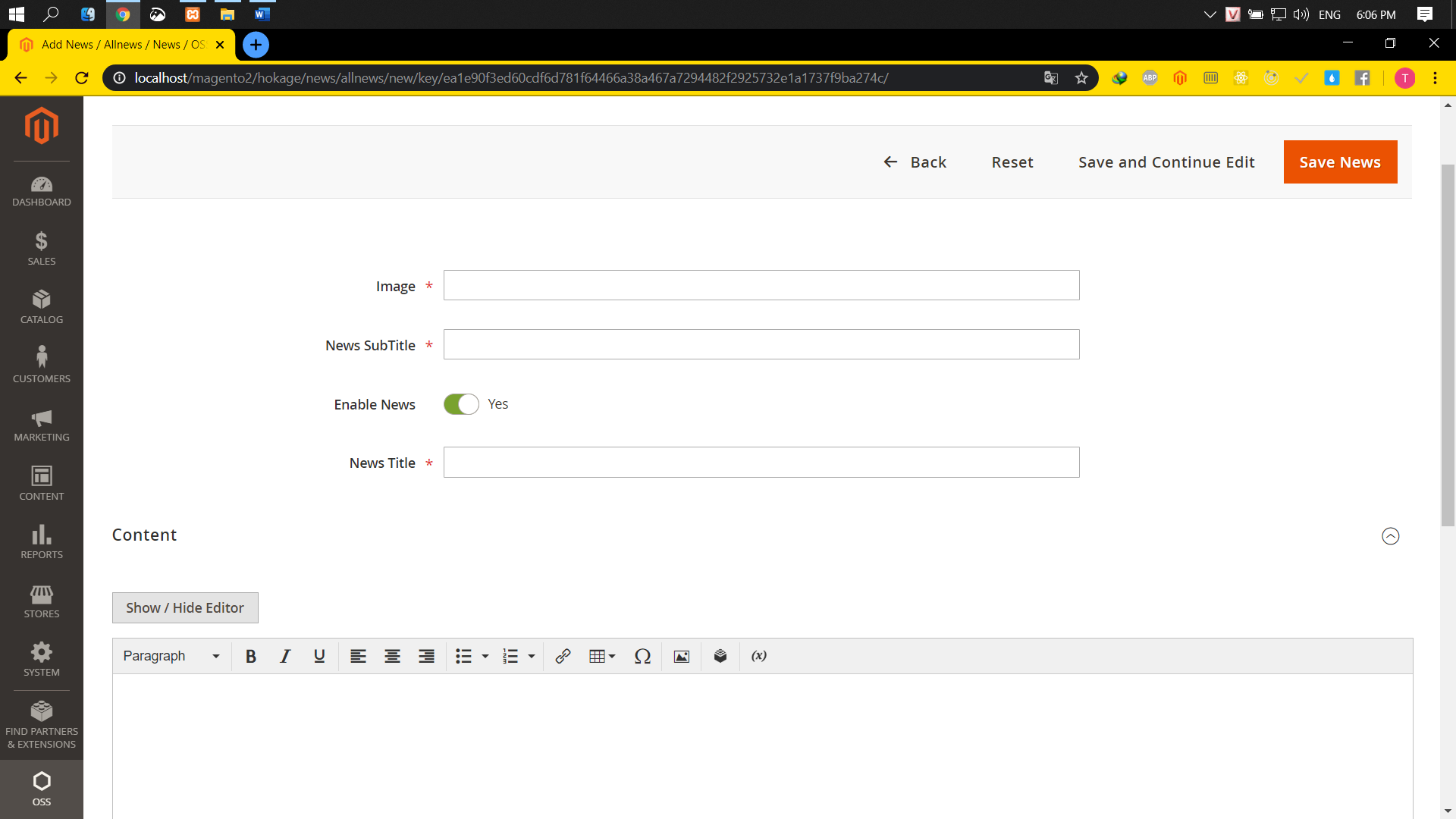


|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Click chọn combobox | Cho phép thay đổi số lượng hàng hiển thị tối đa trên 1 trang | Không có | Nhấn vào số lượng hàng muốn hiển thị trên bảng |
| Custom | Có thể thay đổi số lượng cột muốn hiển thị bằng cách nhập | Click chọn button custom | Không thể thay đổi id |

* Thay đổi thông tin ngay trên bảng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Click chọn cột | Cố định thông tin thay đổi bằng các đặt check vào thuộc tính id | Không có | Không có |
| Save | Cho phép thay đổi thông tin theo các thuộc tính đã chọn | Click vào save để lưu lại những thay đổi và chắc chắn rằng không có thuộc tính nào bị null | Không thể thay đổi id |
| Cancel | Hủy bỏ thay đổi và trả lại thông tin trước khi thay đổi | Click vào hủy bỏ và không có thuộc tính nào bị thay đổi | Không có |

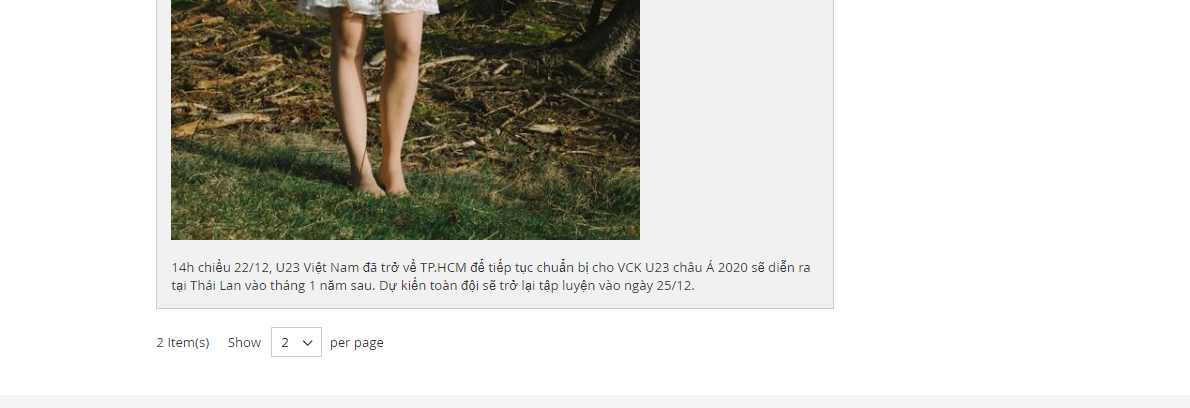
* Thêm trang



|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Add New | Thực hiện thêm 1 trang mới bằng cách chuyển tới link thay đổi trang | Không có | Không có |
| Enable News | Cho phép trang ở trạng thái Enable hoặc disable | Click để chọn thay đổi | Có là yes hoặc no |
| Image | Thực hiện điền tên cho image | Không có | Không được bỏ trống |
| News Subtitle | Thực hiện điền subtitle cho news | Không có | Không được bỏ trống |
| News title | Thực hiện điền title cho New | Không có | Không được bỏ trống |
| Content | Cho phép nhập những nội dung muốn hiển thị lên News | không có | Không được bỏ trống |
| Back | Trở về và không thực hiện them New | Không có | Không có |
| Reset | Hủy bỏ tất các các thay đổi nhưng không trở về trang admin | Không có | Có thể nhập lại thông tin |
| Save and Continue Edit | Cho phép post New lên và có thể tiếp tục thực hiện ghi thông tin | Không có | Save như có thể cho phép null title hoặc content |
| Insert Wedget | Cho phép người dùng có thể insert Wedget vào phần content | Không có | Có thể để trống |
| Insert Image | Cho phép người dùng có thể thêm các hình ảnh từ máy tính | Không có | Có thể để trống |
| Insert Variable | Cho phép insert các danh mục như: địa chỉ, thành phố, quốc gia, tên cửa hàng, … | Không có | Có thể để trống |
| Save News | Thực hiện lưu thông tin New lên trang admin và chuyển về trang admin | Không có | Không đc phép bỏ trống 1 trong 2 hoặc cả 2 nội dung là conten và title |

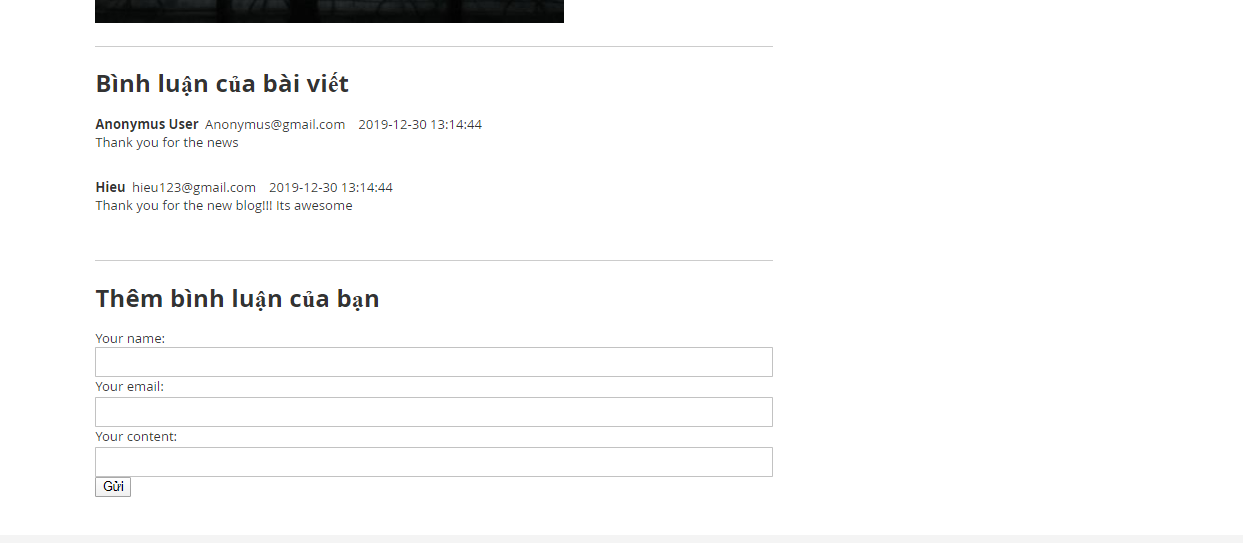
1. Thiết kế xử ý cho trang front-end

* Load thông tin News



|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Load danh sách các news | Thực hiện tải toàn bộ danh sach các new được đăng lên | Không có | Chỉ cho phép hiển thị các new ở dạng enable |
| Combobox show | Cho phép tải số lượng new lên bảng | Click để chọn thay đổi | Chọn số new tường ứng |
| Label item | Đếm số lượng new được đăng tải lên trang | Không có | Không có |
| Component | Hiển thị thông tin có trong new. Người dùng có thể click chọn để đọc thông tin chi tiết của new | Click để chọn new | Không có |

* Bình luận bài viết



|  |  |  |  |
| --- | --- | --- | --- |
| **Tên xữ lý** | **Ý nghĩa** | **Điều kiện** | **Ghi chú** |
| Bình luận của bài viết | Cho biết thông tin, ý nghĩ của những người dùng khác để lại | không có | Không thể xóa |
| Thêm bình luận | Cho phép bạn để lại bình luận, đánh giá về bài viết bằng cách để lại tên, thông tin email và nội dung bình luận | Không có | Không có |
| Your name | Nhập nickname người dùng muốn bình luận | Nhấn vào để điền | Không được bỏ trống |
| Your email | Nhập thông tin email của người dùng để xác thực tài khoản | Nhấn vào để điền | Không được bỏ trống |
| Your content | Nhập nội dung mà người dùng muốn đánh giá về bài viết | Nhấn vào để điền | Không được bỏ trống |

1. **Kết đoạn và hướng phát triển**
   1. **Những khó khăn khi phát triển module**

* Nhóm thực hiện không có thành viên có kinh nghiệm trong phát triển PHP
* Magento là một open source khá lớn, khiến việc phát triển gặp khó khăn cụ thể là developer mode của magento hoạt động chậm chạp khiến việc debug gặp nhiều khó khăn và tốn thời gian
* Lượng các design pattern áp dụng khá lớn và rộng rãi khiến việc nghiên cứu và áp dụng chưa được chuẩn chỉ
* Không có các thiết bị phần cứng cần thiết để đáp ứng requirements của magento
* Nhóm gồm các thành viên năm 3 là chính với lượng kiến thức chưa đủ để đáp ứng khả năng áp dụng framework
  1. **Hướng phát triển**
* Phát triển module thêm các chức năng như tối ưu hỗ trợ cho SEO, ...
* Hoàn thiện module theo hướng expansible

**LINK THAM KHẢO**

<http://www.magentocommerce.com/?ACT=25&fid=40&aid=9563_38jYypduC0XKiAfzF4Zi&board_id=1>

[*https://inchoo.net/magento/magentos-database-layout-and-its-eav-structure/*](https://inchoo.net/magento/magentos-database-layout-and-its-eav-structure/)

[*http://namluu.com/magento/boc-me-cach-magento-to-chuc-thiet-ke-san-pham-phan-ky-thuat/#EAV*](http://namluu.com/magento/boc-me-cach-magento-to-chuc-thiet-ke-san-pham-phan-ky-thuat/#EAV)

[*http://namluu.com/magento/lap-trinh-module-magento-2-tim-hieu-controller/*](http://namluu.com/magento/lap-trinh-module-magento-2-tim-hieu-controller/)

[*http://namluu.com/magento/gioi-thieu-ve-magento-2/*](http://namluu.com/magento/gioi-thieu-ve-magento-2/)

[*https://devdocs.magento.com/guides/v2.3/architecture/archi\_perspectives/arch\_diagrams.html*](https://devdocs.magento.com/guides/v2.3/architecture/archi_perspectives/arch_diagrams.html)

*https://www.javatpoint.com/architecture-of-magento-2*